

# **FlashRunner Series**

## **High-Performance, Standalone In-System Programmers**

### **Programmer's Manual**

Revision 1.21 — October 2019

***We want your feedback!***

*SMH Technologies is always on the lookout for new ways to improve its Products and Services. For this reason feedback, comments, suggestions or criticisms, however small, are always welcome.*

**SMH Technologies S.r.l.**

via Giovanni Agnelli, 1

33083 Villotta di Chions (PN) Italy

E-mail (general information): [info@smh-tech.com](mailto:info@smh-tech.com)

E-mail (technical support): [support@smh-tech.com](mailto:support@smh-tech.com)

Web: <http://www.smh-tech.com>

---

**Important**

SMH Technologies reserves the right to make improvements to FlashRunner, its documentation and software routines, without notice. Information in this manual is intended to be accurate and reliable. However, SMH Technologies assumes no responsibility for its use; nor for any infringements of rights of third parties which may result from its use.

SMH TECHNOLOGIES WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

**Trademarks**

SMH Technologies is the licensee of the SofTec Microsystems trademark.

All other product or service names are the property of their respective owners.

# Contents

## 0 Before Starting 27

- 0.1 Important Notice to Users 27
- 0.2 Getting Technical Support 27

## 1 System Setup/Upgrade 29

- 1.1 Software Setup 29
- 1.2 Programming Algorithms and Licenses 29
  - 1.2.1 *Installing New Licenses* 30
- 1.3 Upgrading the Firmware 31

## 2 FlashRunner Commands 32

- 2.1 Overview 32
  - 2.1.1 *Host Mode* 32
  - 2.1.2 *Standalone Mode* 33
- 2.2 Command Syntax 33
  - 2.2.1 *Sending a Command* 33
  - 2.2.2 *Receiving the Answer* 35
  - 2.2.3 *Numeric Parameters* 36
- 2.3 Command Summary 36
- 2.4 Command Reference 39
  - 2.4.1 *Command Documentation Conventions* 39
  - 2.4.2 *ASECURE* 40
  - 2.4.3 *AUNSECURE* 41
  - 2.4.4 *ACHGPWD* 42
  - 2.4.5 *ACHGUSER* 43
  - 2.4.6 *DLOCK* 44
  - 2.4.7 *DMGET* 45
  - 2.4.8 *DMSET* 46

## Contents

---

2.4.9	DUNLOCK	47
2.4.10	FSCD	48
2.4.11	FSCLRLOG	49
2.4.12	FSELDIR	50
2.4.13	FSEXIST51	
2.4.14	FSFORMAT	52
2.4.15	FSFREE	54
2.4.16	FSGETFILE	55
2.4.17	FSGETTIME	57
2.4.18	FSLS	58
2.4.19	FSMD	59
2.4.20	FSMKLOG	60
2.4.21	FSRD	61
2.4.22	FSRF	62
2.4.23	FSENDFILE	63
2.4.24	FSSETTIME	65
2.4.25	LVDSET CH	66
2.4.26	LVDSET DISABLE	68
2.4.27	LVDSET ENABLE	69
2.4.28	LVDSET LPT	71
2.4.29	LVDSET LPW72	
2.4.30	MCGETGW	73
2.4.31	MCGETIP	74
2.4.32	MCGETMAC	75
2.4.33	MCGETNM	76
2.4.34	MCGETTIME	77
2.4.35	MCGETVER	78
2.4.36	MCRESET	79
2.4.37	MCRESET ENGx	80
2.4.38	MCSETENG	81
2.4.39	MCSETGW	82
2.4.40	MCSETIP	83
2.4.41	MCSETNM	84
2.4.42	MCSETTIME	85
2.4.43	MCSPING	86

2.4.44	<i>MCUPDATE</i>	87
2.4.45	<i>MXCLOSE</i>	88
2.4.46	<i>MXGET</i>	89
2.4.47	<i>MXOPEN</i>	90
2.4.48	<i>OCDSET DISABLE</i>	91
2.4.49	<i>OCDSET ENABLE</i>	92
2.4.50	<i>RLYGET CYCLES</i>	93
2.4.51	<i>RLYGET DRVMODE</i>	95
2.4.52	<i>RLYSET DRVMODE</i>	96
2.4.53	<i>RLYSET CLOSE</i>	97
2.4.54	<i>RLYSET OPEN</i>	99
2.4.55	<i>RUN</i>	101
2.4.56	<i>SGETCRC</i>	102
2.4.57	<i>SGETCRC32</i>	103
2.4.58	<i>SGETERR</i>	104
2.4.59	<i>SGETSTS</i>	105
2.4.60	<i>SGETVER</i>	107
2.4.61	<i>SPING</i>	108
2.4.62	<i>SSETBR</i>	109
2.4.63	<i>TCSETDEV</i>	110
2.4.64	<i>TPSRCPATCH</i>	112
2.4.65	<i>TCSETPAR</i>	114
2.4.66	<i>TPCMD</i>	116
2.4.67	<i>TPEND</i>	117
2.4.68	<i>TPSETDST</i>	119
2.4.69	<i>TPSETSRC</i>	120
2.4.70	<i>TPSTART</i>	121
2.4.71	<i>TST</i>	122

### **3 Scripts 131**

3.1	Overview	131
3.2	Execution and Termination	133
3.2.1	<i>Standalone Script Execution</i>	133
3.2.2	<i>Remote Script Execution</i>	134

3.2.3	<i>Script Termination</i>	134
3.3	Script-Specific Directives	135
3.3.1	<i>Logging</i>	135
3.3.2	<i>Error Handling (ERR_ON and HALT_ON directives)</i>	138
3.3.3	<i>DELAY and WAIT Directives</i>	139
3.3.4	<i>PRINT Directive</i>	139
3.4	Comments	140
3.5	Control Flow on FlashRunner Series	140
3.5.1	<i>Introduction</i>	140
3.5.2	<i>Command syntax</i>	141
3.5.3	<i>Notes</i>	142
3.5.4	<i>Example</i>	142
<b>4</b>	<b>Dynamic Memory and Serial Numbering</b>	<b>143</b>
4.1	Introduction	143
4.2	Example	143
<b>5</b>	<b>SD Card File Structure</b>	<b>145</b>
5.1	Overview	145
5.2	Card Formatting	147
5.3	Using the Card on a PC	148
5.4	File Structure	149
<b>6</b>	<b>Data Protection System</b>	<b>151</b>
6.1	Overview	151
6.2	Installation	152
6.3	Data Protection System Operations	152
6.3.1	<i>Modes, Users and Passwords</i>	153
6.3.2	<i>SD Card Locking/Unlocking</i>	156

## 7 FlashRunner Control Panel 159

- 7.1 Overview 159
  - 7.1.1 *Starting the Control Panel and Connecting to FlashRunner* 159
  - 7.1.2 *Starting the FRPXIA3 Control Panel and Connecting to FlashRunner FRPXIA3* 161
  - 7.1.3 *Sending and Receiving FlashRunner Commands* 163
  - 7.1.4 *Creating FlashRunner Binary Files* 163
  - 7.1.5 *Transferring Scripts to the Instrument* 164
  - 7.1.6 *Retrieving Log Files* 165
  - 7.1.7 *Installing Programming Algorithms and Licenses* 165
  - 7.1.8 *Updating the Firmware* 166
- 7.2 Command-Line Utilities 166

## 8 FlashRunner Interface Library 169

- 8.1 Overview 169
- 8.2 The FlashRunner Interface Library 169
- 8.3 Installation 169
- 8.4 Interface Library Reference 170
  - 8.4.1 *Using the Interface Library Functions* 170
  - 8.4.2 *Return Values of the Interface Library Functions* 171
  - 8.4.3 *Unicode Functions* 171
  - 8.4.4 *Application Examples* 171
- 8.5 Function Reference for FlashRunner FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03, FR04 171
  - 8.5.1 *FR\_CloseCommunication* 172
  - 8.5.2 *FR\_FirmwareUpgrade* 173
  - 8.5.3 *FR\_GetAnswer* 175
  - 8.5.4 *FR\_GetFile* 177
  - 8.5.5 *FR\_GetLastErrorMessage* 180
  - 8.5.6 *FR\_OpenCommunication* 181
  - 8.5.7 *FR\_SendCommand* 182
  - 8.5.8 *FR\_SendFile* 183

- 8.6 Function Reference for FlashRunner FRPXIA3 185
  - 8.6.1 *FR\_CloseCommunication* 186
  - 8.6.2 *FR\_FirmwareUpgrade* 187
  - 8.6.3 *FR\_GetAnswer* 189
  - 8.6.4 *FR\_GetFile* 191
  - 8.6.5 *FR\_GetLastErrorMessage* 193
  - 8.6.6 *FR\_OpenCommunication* 194
  - 8.6.7 *FR\_ScanBus* 195
  - 8.6.8 *FR\_SendCommand* 196
  - 8.6.9 *FR\_SendFile* 197
- 8.7 FlashRunner FRPXIA3 LabVIEW Interface Library 199

## **9 LAN Settings 201**

- 9.1 Overview 201
- 9.2 Setting Up PC Connection 202
- 9.3 Testing Board Connection 207
- 9.4 Changing FlashRunner's IP Address 208

## **10 Target-Specific Commands and Connections 213**

- 10.1 Overview 213
- 10.2 ARM7214
  - 10.2.1 *Specific Commands* 214
  - 10.2.2 *Connections* 215
- 10.3 ARM9216
  - 10.3.1 *Specific Commands* 216
  - 10.3.2 *Connections* 217
- 10.4 Atmel AVR8 218
  - 10.4.1 *Specific Commands* 218
  - 10.4.2 *Connections* 218
- 10.5 Cypress PSoC 220
  - 10.5.1 *Specific Commands* 220



10.5.2	<i>Connections ISSP</i>	221
10.6	Cypress PSoC 3	222
10.6.1	<i>Specific Commands</i>	222
10.6.2	<i>Connections</i>	222
10.7	ELMOS EL16	224
10.7.1	<i>Specific Commands</i>	224
10.7.2	<i>Connections</i>	225
10.8	Infineon TriCore	226
10.8.1	<i>Specific Commands</i>	226
10.8.2	<i>Connections</i>	227
10.9	Freescale 56F800E	228
10.9.1	<i>Specific Commands</i>	228
10.9.2	<i>Connections</i>	230
10.10	Texas Instruments TMS320 Devices	231
10.10.1	<i>Specific Commands</i>	231
10.10.2	<i>Connections</i>	232
10.11	Freescale HC08	233
10.11.1	<i>Specific Commands</i>	233
10.11.2	<i>Connections</i>	234
10.12	Freescale MCF52	235
10.12.1	<i>Specific Commands</i>	235
10.12.2	<i>Connections (BDM Communication Mode)</i>	236
10.12.3	<i>Connections (EzPort Communication Mode)</i>	236
10.13	Freescale MPC500	237
10.13.1	<i>Specific Commands</i>	237
10.13.2	<i>Connections</i>	238
10.14	Freescale MPC5000	239
10.14.1	<i>Specific Commands</i>	239
10.14.2	<i>Connections</i>	240
10.15	Freescale RS08	241
10.15.1	<i>Specific Commands</i>	241
10.15.2	<i>Connections</i>	242

## Contents

---

10.16	Freescal S08	243
	10.16.1 <i>Specific Commands</i>	243
	10.16.2 <i>Connections</i>	244
10.17	Freescal S12	245
	10.17.1 <i>Specific Commands</i>	245
	10.17.2 <i>Connections</i>	246
10.18	Fujitsu MB91	247
	10.18.1 <i>Specific Commands</i>	247
	10.18.2 <i>Connections UART</i>	248
	10.18.3 <i>Connections MDI</i>	248
10.19	Freescal S12 (FSL_D Algorithm)	250
	10.19.1 <i>Specific Commands</i>	250
	10.19.2 <i>Connections</i>	251
10.20	Fujitsu MB90	252
	10.20.1 <i>Specific Commands</i>	252
	10.20.2 <i>Connections</i>	253
10.21	Infineon C166	254
	10.21.1 <i>Specific Commands</i>	254
	10.21.2 <i>Connections</i>	255
10.22	Infineon XC166	256
	10.22.1 <i>Specific Commands</i>	256
	10.22.2 <i>Connections (UART Communication Mode, CS Series)</i>	256
	10.22.3 <i>Connections (UART Communication Mode, CM Series)</i>	257
	10.22.4 <i>Connections (JTAG Communication Mode)</i>	257
10.23	Infineon XC2000	259
	10.23.1 <i>Specific Commands</i>	259
	10.23.2 <i>Connections JTAG</i>	260
10.24	Infineon XC2000 (DAP Protocol)	261
	10.24.1 <i>Specific Commands</i>	261
	10.24.2 <i>Connections</i>	262
10.25	Infineon XC800 (JTAG)	263

10.25.1	<i>Specific Commands</i>	263
10.25.2	<i>Connections JTAG</i>	264
10.25.3	<i>Connections UART</i>	264
10.26	Infineon XMC1xx	265
10.26.1	<i>Specific Commands</i>	265
10.26.2	<i>Connections SWD</i>	266
10.27	Toshiba TLCS-870/C/TMP8	267
10.27.1	<i>Specific Commands</i>	267
10.27.2	<i>Connections</i>	268
10.28	Renesas RX62	269
10.28.1	<i>Specific Commands</i>	269
10.28.2	<i>Connections</i>	271
10.29	NXP P89C51R	272
10.29.1	<i>Specific Commands</i>	272
10.29.2	<i>Connections</i>	273
10.30	XC800 (DAP and UART 1 wire protocols)	274
10.30.1	<i>Specific Commands</i>	274
10.30.2	<i>Connections</i>	276
10.31	NXP LPC1100L Algorithm	277
10.31.1	<i>Specific Commands</i>	277
10.31.2	<i>Connections</i>	278
10.32	Atmel ATXMega	279
10.32.1	<i>Specific Commands</i>	279
10.32.2	<i>Connections JTAG</i>	280
10.32.3	<i>Connections PDI</i>	280
10.33	Microchip PIC10/12/16	281
10.33.1	<i>Specific Commands</i>	281
10.33.2	<i>Connections</i>	282
10.34	Microchip PIC18	283
10.34.1	<i>Specific Commands</i>	283
10.34.2	<i>Connections</i>	284
10.35	Microchip PIC24, dsPIC30, dsPIC33	285
10.35.1	<i>Specific Commands</i>	285

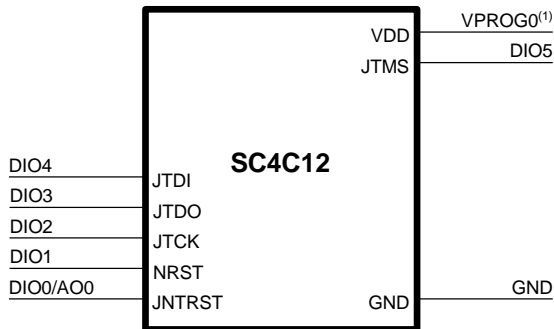
## Contents

---

10.35.2	<i>Connections</i>	286
10.36	Microchip PIC32	287
10.36.1	<i>Specific Commands</i>	287
10.36.2	<i>Connections</i>	288
10.37	Nec 78K	289
10.37.1	<i>Specific Commands</i>	289
10.37.2	<i>Connections (78K0R)</i>	290
10.37.3	<i>Connections (Kx1+ Only)</i>	290
10.37.4	<i>Connections (All Other Devices)</i>	291
10.38	Nec V850	292
10.38.1	<i>Specific Commands</i>	292
10.38.2	<i>Connections</i>	293
10.39	Renesas H8/300H	294
10.39.1	<i>Specific Commands</i>	294
10.39.2	<i>Connections</i>	295
10.40	Renesas H8S/H8SX/SuperH (Serial Boot Mode)	296
10.40.1	<i>Specific Commands</i>	296
10.40.2	<i>Connections</i>	296
10.41	Renesas M16C/M32C	298
10.41.1	<i>Specific Commands</i>	298
10.41.2	<i>Connections</i>	299
10.42	Renesas R8C/Tiny	300
10.42.1	<i>Specific Commands</i>	300
10.42.2	<i>Connections</i>	301
10.43	Renesas R32C	303
10.43.1	<i>Specific Commands</i>	303
10.43.2	<i>Connections</i>	304
10.44	Renesas SuperH (JTAG Interface)	305
10.44.1	<i>Specific Commands</i>	305
10.44.2	<i>Connections</i>	306
10.45	Samsung SAM8	307
10.45.1	<i>Specific Commands</i>	307
10.45.2	<i>Connections</i>	308

10.46	Serial Memories (SPI Interface)	309
	10.46.1 <i>Specific Commands</i>	309
	10.46.2 <i>Connections</i>	311
10.47	Serial Memories (Microwire Interface)	312
	10.47.1 <i>Specific Commands</i>	312
	10.47.2 <i>Connections</i>	313
10.48	Serial Memories (I <sup>2</sup> C Interface)	314
	10.48.1 <i>Specific Commands</i>	314
	10.48.2 <i>Connections</i>	315
10.49	Serial Memories (I2C Interface) SERMEM_B Algorithm	316
	10.49.1 <i>Specific Commands</i>	316
	10.49.2 <i>Connections</i>	317
10.50	Silicon Labs C8051F	318
	10.50.1 <i>Specific Commands</i>	318
	10.50.2 <i>Connections</i>	319
10.51	STMicroelectronics ST10	320
	10.51.1 <i>Specific Commands</i>	320
	10.51.2 <i>Connections</i>	321
10.52	STMicroelectronics ST72C	322
	10.52.1 <i>Specific Commands</i>	322
	10.52.2 <i>Connections</i>	323
10.53	STMicroelectronics ST7F	324
	10.53.1 <i>Specific Commands</i>	324
	10.53.2 <i>Connections</i>	325
10.54	STMicroelectronics STM32326	
	10.54.1 <i>Specific Commands</i>	326
	10.54.2 <i>Connections</i>	326
10.55	STMicroelectronics STM8	328
	10.55.1 <i>Specific Commands</i>	328
	10.55.2 <i>Connections</i>	329
10.56	STMicroelectronics STR9	330
	10.56.1 <i>Specific Commands</i>	330
	10.56.2 <i>Connections</i>	331

10.57 Texas Instruments MSP430/SC4C12	332
10.57.1 Specific Commands	332
10.57.2 Connections JTAG	333



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

10.57.3 Connections SWD	334
10.58 Texas Instruments TMS570	335
10.58.1 Specific Commands	335
10.58.2 Connections	336
10.59 TI BQ40/UCD30XX	337
10.59.1 Specific Commands	337
10.59.2 Connections	338
10.60 Renesas V850E2339	
10.60.1 Specific Commands	339
10.60.2 Connections	340
10.61 Microchip MCP2502X/5X	340
10.61.1 Specific Commands	340
10.61.2 Connections	341
10.62 STMicroelectronics STM32 (Serial Wire Debug)	343
10.62.1 Specific Commands	343
10.62.2 Connections	344
10.63 Texas Instruments CC2533	345
10.63.1 Specific Commands	345

- 10.63.2 *Connections* 346
- 10.64 NXP LPC122x (Serial Wire Debug) 347
  - 10.64.1 *Specific Commands* 347
  - 10.64.2 *Connections* 348
- 10.65 Renesas RL78 Devices 349
  - 10.65.1 *Specific Commands* 349
  - 10.65.2 *Connections for UART Single Wire (1Wire) e UART 2Wire* 350
- 10.66 Freescale Kinetis 351
  - 10.66.1 *Specific Commands* 351
  - 10.66.2 *Connections* 352
- 10.67 Texas Instruments Stellaris (Serial Wire Debug) 353
  - 10.67.1 *Specific Commands* 353
  - 10.67.2 *Connections* 354
- 10.68 Analog Devices ADP104w 355
  - 10.68.1 *Specific Commands* 355
  - 10.68.2 *Connections* 356
- 10.69 AMTEL ATSAM 357
  - 10.69.1 *Specific Commands* 357
  - 10.69.2 *Connections JTAG* 358
  - 10.69.3 *Connections SWD* 358
- 10.70 ATMEL AT89/AT90/ATMEGA/ATTINY 359
  - 10.70.1 *Specific Commands* 359
  - 10.70.2 *Connections UART* 360
  - 10.70.3 *Connections SPI* 360
  - 10.70.4 *Connections JTAG* 361
- 10.71 Silicon labs SIM3C-157 362
  - 10.71.1 *Specific Commands* 362
  - 10.71.2 *Connections JTAG* 363
- 10.72 NXP LPC43 364
  - 10.72.1 *Specific Commands* 364
  - 10.72.2 *Connections JTAG* 365
- 10.73 Freescale MX6D 366

## Contents

---

10.73.1	<i>Specific Commands</i>	366
10.73.2	<i>Connections JTAG</i>	367
10.74	Silicon Labs BGM111	368
10.74.1	<i>Specific Commands</i>	368
10.74.2	<i>Connections SWD</i>	368
10.75	Nordic NRF528	370
10.75.1	<i>Specific Commands</i>	370
10.75.2	<i>Connections SWD</i>	370
10.76	ALTERA EPM240	372
10.76.1	<i>Specific Commands</i>	372
10.76.2	<i>Connections JTAG</i>	373
10.77	Lattice	374
10.77.1	<i>Specific Commands</i>	374
10.77.2	<i>Connections JTAG</i>	375
10.78	Cast R8051XC2	376
10.78.1	<i>Specific Commands</i>	376
10.78.2	<i>Connections JTAG</i>	377
10.79	Cypress CY8C4	378
10.79.1	<i>Specific Commands</i>	378
10.79.2	<i>Connections SWD</i>	379
10.80	STMicroelectronics SPC500	380
10.80.1	<i>Specific Commands</i>	380
10.80.2	<i>Connections JTAG</i>	381
10.81	Freescale DSP56	382
10.81.1	<i>Specific Commands</i>	382
10.81.2	<i>Connections JTAG</i>	383
10.82	Freescale MKV30	384
10.82.1	<i>Specific Commands</i>	384
10.82.2	<i>Connections SWD</i>	385
10.83	Texas Instruments CC2640	386
10.83.1	<i>Specific Commands</i>	386
10.83.2	<i>Connections JTAG</i>	387



10.84	Freescale MC9S12/MC9RS0	388
	10.84.1 Specific Commands	388
	10.84.2 Connections BDM	389
10.85	Freescale MC68H	390
	10.85.1 Specific Commands	390
	10.85.2 Connections JTAG	391
10.86	Infineon TLE9000	392
	10.86.1 Specific Commands	392
	10.86.2 Connections SWD	393
10.87	NXP NCF29A	394
	10.87.1 Specific Commands	394
	10.87.2 Connections	395
10.88	Renesas RH850	396
	10.88.1 Specific Commands	396
	10.88.2 Connections CSI	397
10.89	Renesas HD64F/R4F/R5F	398
	10.89.1 Specific Commands	398
	10.89.2 Connections	399
10.90	Renesas M30	400
	10.90.1 Specific Commands	400
	10.90.2 Connections	401
10.91	Renesas R5F2	402
	10.91.1 Specific Commands	402
	10.91.2 Connections	403
10.92	Renesas R5F1	404
	10.92.1 Specific Commands	404
	10.92.2 Connections (R5F1)	405
10.93	Renesas R5F5	406
	10.93.1 Specific Commands	406
	10.93.2 Connections (R5F5)	407
10.94	Texas Instruments BQ40/UCD3	408
	10.94.1 Specific Commands	408
	10.94.2 Connections	409

10.95 Maxim Teridian 71M65xx TRDN	410
10.95.1 <i>Specific Commands</i>	410
10.95.2 <i>Connections</i>	411
10.96 UBX UBXM8030	412
10.96.1 <i>Specific Commands</i>	412
10.96.2 <i>Connections</i>	413

## **11 Errors 414**

11.1 Overview	414
11.2 Status LED	414
11.3 Command Errors	415

## **12 Approved SD Cards 562**

# Index of Figures

- Figure 1: Command Syntax 34
- Figure 2: Typical Answers 35
- Figure 3: Example of Decimal and Hexadecimal Numeric Parameters 36
- Figure 4: FlashRunner Registers 124
- Figure 5: Script Example 132
- Figure 6: Script Execution and Control Lines 134
- Figure 7: Basic Logging 136
- Figure 8: Advanced Logging 136
- Figure 9: SD Card File Structure 149
- Figure 10: Typical Data Protection System Workflow 152
- Figure 11: Data Protection System Modes and Users 154
- Figure 12: Data Protection System SD Card Lock 157
- Figure 13: FlashRunner Control Panel, Communication Settings 161
- Figure 14: FlashRunner FRPXIA3 Control Panel 162
- Figure 15: FlashRunner Control Panel, Binary File Conversion 163
- Figure 16: FlashRunner Control Panel, Binary File Conversion Diagram 164
- Figure 17: FlashRunner Control Panel, File Transfer 164
- Figure 18: Windows Control Panel 202
- Figure 19: Network Connections 203
- Figure 20: New Connection Wizard, Step 1 203
- Figure 21: New Connection Wizard, Step 2 204
- Figure 22: New Connection Wizard, Step 3 204
- Figure 23: New Connection Wizard, Step 4 205
- Figure 24: New Connection Wizard, Step 5 205
- Figure 25: Local Area Connection Status 206
- Figure 26: Local Area Connection Properties 206
- Figure 27: TCP/IP Properties 207
- Figure 28: Pinging FlashRunner 208
- Figure 29: Disabling the Proxy Server, Step 1 209
- Figure 30: Disabling the Proxy Server, Step 2 209
- Figure 31: LAN Enabled FlashRunner Models, Configuration Page 210
- Figure 32: LAN Enabled FlashRunner Models, Network Summary 211
- Figure 33: LAN Enabled FlashRunner Models, Network Configuration 212
- Figure 34: Example of Connection to an ARM7 Target Device 215

- Figure 35: Example of Connection to an AVR8 Target Device 219
- Figure 36: Example of Connection to a PSoC Target Device 221
- Figure 37: Example of Connection to a Cypress PSoC 3 Target Device through JTAG interface. 223
- Figure 38: Example of Connection to an EL16 Target Device 225
- Figure 39: Example of Connection to a TriCore Target Device 227
- Figure 40: Correct Linker Settings for 56F800E S-Rec File Generation 229
- Figure 41: Example of Connection to a 56F800E Target Device 230
- Figure 42: Example of Connection to TMS320 Target Device 232
- Figure 43: Example of Connection to an HC08 Target Device 234
- Figure 44: Example of Connection to a MCF52 Target Device (BDM Communication) 236
- Figure 45: Example of Connection to an MCF52 Target Device (EzPort Communication) 236
- Figure 46: Example of Connection to a MPC500 Target Device 238
- Figure 47: Example of Connection to a MPC5000 Target Device 240
- Figure 48: Example of Connection to an RS08 Target Device 242
- Figure 49: Example of Connection to a S08 Target Device 244
- Figure 50: Example of Connection to a S12 Target Device 246
- Figure 51: Example of Connection to a MB91 Target Device 248
- Figure 52: Example of Connection to a S12 Target Device 251
- Figure 53: Example of Connection to a MB90 Target Device 253
- Figure 54: Example of Connection to a C166 Target Device 255
- Figure 55: Example of Connection to a XC166 (UART Mode, CS Series) Target Device 257
- Figure 56: Example of Connection to a XC166 (UART Mode, CM Series) Target Device 257
- Figure 57: Example of Connection to a XC166 Target Device (JTAG Mode) 258
- Figure 58: Example of Connection to a XC2000 Target Device 260
- Figure 59: Example of Connection to a XC2000 (DAP protocol) Target Device 262
- Figure 60: Example of Connection to a XC800 Target Device 264
- Figure 61: Example of Connection to a Toshiba TLCS-870/C Target Device 268
- Figure 62: Example of Connection to RX62T Target Device 271
- Figure 63: Example of Connection to a NXP P89C51R Target Device 273
- Figure 64: Example of Connection to a XC800 Target Device (DAP) 276
- Figure 65: Example of Connection to a XC800 Target Device (UART 1 Wire) 276
- Figure 66: Example of Connection to a NXP LPC1100L Target Device 278
- Figure 67: Example of Connection to a ATXMEGA (JTAG) Target Device 280
- Figure 68: Example of Connection to a ATXMEGA (PDI) Target Device 280
- Figure 69: Example of Connection to a PIC10/12/16 Target Device 282
- Figure 70: Example of Connection to a PIC18 Target Device 284
- Figure 71: Example of Connection to a PIC24 Target Device 286
- Figure 72: Example of Connection to a Microchip PIC32 Target Device using EJTAG protocol. 288
- Figure 73: Example of Connection to a Microchip PIC32 Target Device using ICSP protocol. 288

- Figure 74: Example of Connection to a 78K0R Target Device 290
- Figure 75: Example of Connection to a 78K0S (Kx1+ Only) Target Device 290
- Figure 76: Example of Connection to a 78K0 Target Device 291
- Figure 77: Example of Connection to a V850 Target Device 293
- Figure 78: Example of Connection to a H8/300H Target Device 295
- Figure 79: Example of Connection to a H8S/H8SX/SuperH Device (Serial Boot Mode) 297
- Figure 80: Example of Connection to a M16C/M32C Target Device 299
- Figure 81: Example of Connection to a R8C/Tiny Target Device (UART Comm. Mode) 301
- Figure 82: Example of Connection to a R8C/Tiny Target Device (USART Comm. Mode) 301
- Figure 83: Example of Connection to a R8C/Tiny Target Device (Single-Wire Comm. Mode) 302
- Figure 84: Example of Connection to a SuperH Target Device (JTAG Interface) 306
- Figure 85: Example of Connection to a SAM8 Target Device 308
- Figure 86: Example of Connection to a Serial Memory (SPI Interface) 311
- Figure 87: Example of Connection to a Serial Memory (Microwire Interface) 313
- Figure 88: Example of Connection to a Serial Memory (I<sup>2</sup>C Interface) 315
- Figure 89: Example of Connection to a Serial Memory Sermem\_B Algorithm (I<sup>2</sup>C Interface) 317
- Figure 90: Example of Connection to a C8051F Target Device (JTAG Interface) 319
- Figure 91: Example of Connection to a C8051F Target Device (C2 Interface) 319
- Figure 92: Example of Connection to a ST10 Target Device 321
- Figure 93: Example of Connection to an ST7C Target Device 323
- Figure 94: Example of Connection to an ST7F Target Device 325
- Figure 95: Example of Connection to a STM32 Target Device 327
- Figure 96: Example of Connection to a STM8 Target Device 329
- Figure 97: Example of Connection to an STR9 Target Device 331
- Figure 98: Example of Connection to an MSP430 Target Device 333
- Figure 99: Example of Connection to a TMS570 Target Device 336
- Figure 100: Example of Connection to a UCD30XX Target Device 338
- Figure 101: Example of Connection to a V850E2 Target Device 340
- Figure 102: Example of Connection to a Microchip MCP2502X/5X Target Device 341
- Figure 103: Example of Connection to a STM32 Cortex Target Device MCU through SWD 344
- Figure 104: Example of Connection to a Texas Instruments CC2533 Target Device 346
- Figure 105: Example of Connection to a LPC 122x Target Device through SWD 348
- Figure 106: Example of Connection to RL78 Device UART Single-Wire 350
- Figure 107: Example of Connection to RL78 Device UART 2Wires 350
- Figure 108: Example of Connection to a Freescale Kinetis Target Device 352
- Figure 109: Example of Connection to a TI Stellaris Target Device MCU through SWD 354



# Index of Tables

Table 1: Command Summary	37
Table 2: SGETSTS Command: FlashRunner Status	105
Table 3: SGETSTS Command: Programming Status	106
Table 4: SGETSTS Command: Active User	106
Table 5: Script-Specific Directives	135
Table 6: ERR_ON and HALT_ON Directives Summary	139
Table 7: FlashRunner Commands Availability for Different Modes/Users	155
Table 8: ARM7 Specific Commands	214
Table 9: Atmel AVR8 Specific Commands	218
Table 10: Cypress PSoc Specific Commands	220
Table 11: Cypress PSoc 3 Specific Commands for JTAG interface.	222
Table 12: ELMOS EL16 Specific Commands	224
Table 13: Infineon TriCore Specific Commands	226
Table 14: Freescale 56F800E Specific Commands	228
Table 15: Texas Instruments TMS320 Specific Commands	231
Table 16: Freescale HC08 Specific Commands	233
Table 17: Freescale MCF52 Specific Commands	235
Table 18: Freescale MPC500 Specific Commands	237
Table 19: Freescale MPC5000 Specific Commands	239
Table 20: Freescale RS08 Specific Commands	241
Table 21: Freescale S08 Specific Commands	243
Table 22: Freescale S12 Specific Commands	245
Table 23: Fujitsu MB91 Specific Commands	247
Table 24: Freescale S12 Specific Commands	250
Table 25: Fujitsu MB90 Specific Commands	252
Table 26: Infineon C166 Specific Commands	254
Table 27: Infineon XC166 Specific Commands	256
Table 28: Infineon XC2000 Specific Commands	259
Table 29: Infineon XC2000 (DAP Protocol) Specific Commands	261
Table 30: Infineon XC800 Specific Commands	263
Table 31: Toshiba TLCS-870/C Specific Commands	267
Table 32: Renesas RX62 Specific Commands	269
Table 33: NXP P89C51R Specific Commands	272
Table 34: Infineon XC800 DAP Specific Commands	274

## Before Starting

---

# 1

Table 35: Infineon XC82x UART 1 Wire Specific Commands	275
Table 36: NXP LPC1100L Specific Commands	277
Table 37: ATMEL ATXMEGA Specific Commands	279
Table 38: Microchip PIC10/12/16 Specific Commands	281
Table 39: Microchip PIC18 Specific Commands	283
Table 40: Microchip PIC24, dsPIC30 and dsPIC33 Specific Commands	285
Table 41: Microchip PIC32 Specific Commands	287
Table 42: NEC 78K Specific Commands	289
Table 43: NEC V850 Specific Commands	292
Table 44: Renesas H8/300H Specific Commands	294
Table 45: Renesas H8S/H8SX/SuperH Specific Commands (Serial Boot Mode)	296
Table 46: Renesas M16C/M32C Specific Commands	298
Table 47: Renesas R8C/Tiny Specific Commands	300
Table 48: Renesas R32C Specific Commands	303
Table 49: Renesas SuperH Specific Commands (JTAG Interface)	305
Table 50: Samsung SAM8 Specific Commands	307
Table 51: Serial Memories (SPI Interface) Specific Commands	309
Table 52: Serial Memories (Microwire Interface) Specific Commands	312
Table 53: Serial Memories (I <sup>2</sup> C Interface) Specific Commands	314
Table 54: Serial Memories (I2C Interface) Specific Commands	316
Table 55: Silicon Labs C8051F Specific Commands	318
Table 56: STMicroelectronics ST10 Specific Commands	320
Table 57: STMicroelectronics ST7C Specific Commands	322
Table 58: STMicroelectronics ST7F Specific Commands	324
Table 59: STM32 Specific Commands	326
Table 60: STMicroelectronics STM8 Specific Commands	328
Table 61: STMicroelectronics STR9 Specific Commands	330
Table 62: Texas Instruments MSP430 Specific Commands	332
Table 63: Texas Instruments TMS570 Specific Commands	335
Table 64: TI UCD30XX Specific Commands	337
Table 65: Renesas V850E2 Specific Commands	339
Table 66: Microchip MCP2502X/5X Specific Commands	340
Table 67: STM32 SWD Specific Commands	343
Table 68: Texas Instruments CC2533 Specific Commands	345
Table 69: NXP LPC122x SWD Specific Commands	347
Table 70: Renesas RL78 Specific Commands	349
Table 71: Freescale Kinetis Specific Commands	351
Table 72: TI Stellaris SWD Specific Commands	353
Table 73: Status LED	414



Table 74: Error Summary 416

Table 75: Recommended SD Cards 562

Table 76: Recommended SD Cards 562



# 0 Before Starting



**Note:** *the FlashRunner System Software CD-ROM and/or SMH Technologies website ([www.smh-tech.com](http://www.smh-tech.com)) may contain an updated version of this user's manual. Please check before continuing reading this documentation.*

2

## 0.1 Important Notice to Users

While every effort has been made to ensure the accuracy of all information in this document, SMH Technologies assumes no liability to any party for any loss or damage caused by errors or omissions or by statements of any kind in this document, its updates, supplements, or special editions, whether such errors are omissions or statements resulting from negligence, accidents, or any other cause.

## 0.2 Getting Technical Support

SMH Technologies is continuously working to improve FlashRunner firmware and to release programming algorithms for new devices. SMH Technologies offers a fast and knowledgeable technical support to all of its customers and is always available to solve specific problems or meet specific needs.

To get in touch with SMH Technologies, please refer to the contact information below.

Phone: +39 0434 421111

Fax: +39 0434 639021

Technical Support: [support@smh-tech.com](mailto:support@smh-tech.com)



# 1 System Setup/Upgrade

## 1.1 Software Setup

The FlashRunner system software setup installs all of the required components to your hard drive. These components include:

- The FlashRunner Control Panel utility;
- Command-line utilities and Interface Library;
- Script examples;
- Documentation in PDF format.

To install the FlashRunner system software:

- Insert the “**System Software**” CD-ROM into your computer’s CD-ROM drive;
- A startup window will automatically appear. Choose “**Install Instrument Software**” from the main menu. Follow the on-screen instructions.



**Note:** *to install the FlashRunner system software on Windows 2000 or Windows XP, you must log in as Administrator.*

## 1.2 Programming Algorithms and Licenses

FlashRunner includes programming algorithms for several devices. In order to program a specific device, however, a specific license file for that device must be purchased.



**Note:** *FlashRunner comes already preinstalled with the license(s) you specified at the moment of purchase. You can purchase additional licenses at any future moment.*

Programming algorithms and license files are stored in the SD card (for more information on the SD card file system, see “SD Card File Structure” on page 145).

# 3

## 1.2.1 Installing New Licenses

When you buy an additional license for a specific device, you will get:

- An algorithm file (.ALG);
- A license file (.LIC);
- A device-specific script example (.FRS).

The .alg file contains the actual programming algorithm for the requested device (and several other devices of the same family).

The .lic file contains an unlocking code for a specific device on a specific FlashRunner instrument (licenses are serial number specific).

The script file contains an example of script to use as a starting point for your specific programming needs (for more information on scripts, see “Scripts” on page 131).

To install the new license, do the following:

1. Copy the **.ALG** file into the **\ALGOS** directory of the SD card (if an .alg file with the same name already exists, overwrite it);
2. Copy the **.LIC** file into the **\LICENSES** directory of the SD card.

To copy files on the SD card, use either a standard card reader connected to a PC or transfer the files using the FlashRunner **FSENDFILE** command.

Alternatively, you can use the FlashRunner Control Panel utility to install new licenses. For more information on the FlashRunner Control Panel, see “FlashRunner Control Panel” on page 159.

Moreover, you could also create subfolders inside `\LICENSES` folder with SN as folder name: in this case FlashRunner will search for a valid license also inside folder name which match current FlashRunner serial number. For example, if your FlashRunner SN is 80512, you could copy all your licenses inside `\LICENSES\80512` path.

### 1.3 Upgrading the Firmware

The FlashRunner firmware can be easily upgraded using the provided Control Panel utility. For more information on the FlashRunner Control Panel, see “FlashRunner Control Panel” on page 159.

## 2 FlashRunner Commands

### 2.1 Overview

FlashRunner is set up and controlled via ASCII-based commands. FlashRunner can receive and execute commands in two ways:

- Over a RS-232 or Ethernet connection (**Host mode**);
- Via “scripts” stored in its SD card (**Standalone mode**).

In the first case, FlashRunner is controlled by a host system; in the latter case, FlashRunner works in standalone mode and is fully autonomous.

6



FlashRunner **FRPXIA3** model is always connected to the PXI controller over the PXI chassis. The commands are sent by using any programming language that is able to send and receive data to/from the device (i.e. Microsoft Visual C++/Visual Basic, National Instrument LabView/LabWindows, etc.) An Interface Library is available upon which you can build your own application (see “FlashRunner Interface Library”). This Interface library works with the PCI driver provided with the installation package.

#### 2.1.1 Host Mode

In Host mode, commands are sent from the host system to FlashRunner:

- By using a terminal utility (like HyperTerminal on Windows);
- By using a TCP/IP command-line utility (like DevCon on Windows);
- By using any programming language that is able to send and receive data to/from a host system COM port or Ethernet port (i.e. Microsoft Visual C++/Visual Basic, National Instrument LabView/LabWindows,



etc.) An Interface Library is available upon which you can build your own application (see “FlashRunner Interface Library” on page 169).

Alternatively, you can use the FlashRunner Control Panel utility to send commands to the instruments. For more information, see “FlashRunner Control Panel” on page 159.



**Note (for terminal utilities):** *by default, FlashRunner communicates at 115200 bps, no parity, 8 data bits, 1 stop bit.*



**Note (for TCP/IP command-line utilities):** *FlashRunner's factory IP address is 192.168.1.100 and data is exchanged on port 2101. For learning how to change the FlashRunner IP address, see “LAN Settings” on page 201.*

## 2.1.2 Standalone Mode

In Standalone mode, FlashRunner does not need a connection to a host system. A group of control lines (SEL[4..0] in the “CONTROL” Connector) determines which of the 32 available “scripts” (stored in the SD card) is to be executed.

A script is simply a text file containing a sequence of FlashRunner interface commands, plus some script-specific directives. Scripts are explained in detail in the next chapter.

## 2.2 Command Syntax

### 2.2.1 Sending a Command

A typical FlashRunner command begins with the command name, followed by a space, followed by zero, one or more parameters (separated by a space), and a final Carriage Return character. All parts of a command are case insensitive. Below is an example of a FlashRunner command:

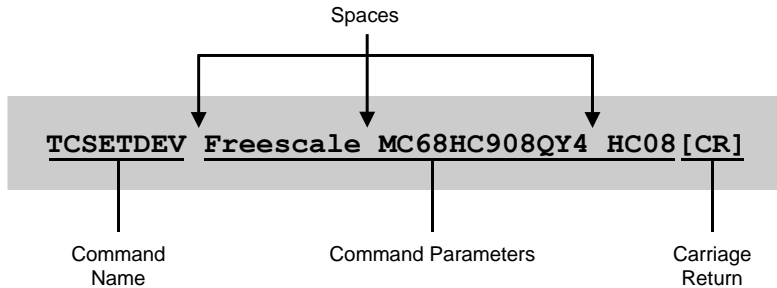


Figure 1: Command Syntax

This command instructs FlashRunner to select the appropriate device and programming algorithm.

The Carriage Return character has ASCII code 13 and corresponds to the `\r` escape character in the C programming language.

## 6



**Note:** when sending commands via a terminal utility, the Carriage Return character is automatically sent when you press the Enter key on your keyboard.

Likewise, inside a script, the Carriage Return character is automatically inserted at the end of a script line when you press the Enter key on your keyboard to start a new script line.

The Carriage Return character, however, must be programmatically added to the command string when sending the command from a programming language.



**Note:** your terminal utility may automatically add the Line Feed character (ASCII code 10, `\n` escape character in C) before the Carriage Return character. Likewise, depending on the editor used to create a script file, a script line may present a Line Feed character before the Carriage Return character. FlashRunner automatically discards this extra character and recognizes the command correctly—that is, FlashRunner accepts both `\n` and `\r\n` sequences.

## 2.2.2 Receiving the Answer

After receiving a command from the host system and executing it, FlashRunner responds with an answer string. The answer string is composed by zero or more response characters, followed by one result character, followed by a final Carriage Return. The result character is:

- > if the command has been executed successfully or
- ! if the command generated an error.

Below are two examples of answer (with and without error):

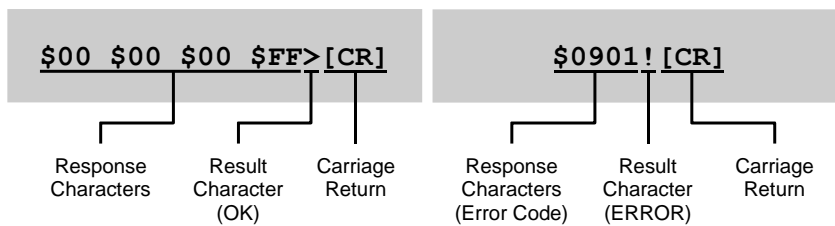


Figure 2: Typical Answers

When a FlashRunner command executes successfully, FlashRunner typically answers with just the > character (unless the command requires data to be returned back—see figure above).

When a FlashRunner command generates an error, FlashRunner answers with a four-digit hexadecimal error code followed by the ! character (see figure above).



**Note:** when FlashRunner commands are executed from within a script (see next sections), no answer is sent to the host system after each command is executed.

### 2.2.3 Numeric Parameters

Every numeric parameter of a command can be expressed either in decimal or hexadecimal format. Hexadecimal numbers must be preceded by the `$` symbol. The figure below shows three example of usage of the `DMSET` command to write four bytes in the FlashRunner's dynamic memory. All three examples shown below are equivalent.

```
DMSET $0000 $04 $CC $DD $EE $FF
DMSET 0 4 $CC $DD $EE $FF
DMSET 0 4 204 221 238 255
```

Figure 3: Example of Decimal and Hexadecimal Numeric Parameters

## 6

Numeric parameters returned by FlashRunner as command answer (CRC, memory data, error codes, etc.) are expressed in hexadecimal or decimal format, depending on the case.

## 2.3 Command Summary

The following table summarizes all of the FlashRunner commands. Each command is fully described in the “Command Reference” section. Target configuration commands and target programming commands can be executed inside scripts (see “Scripts” on page 131 for more information about scripts).

Table 1: Command Summary

Command Syntax	Description	Scriptable
<b>File System Commands</b>		
FSFORMAT	Initializes SD card contents	NO
FSRD <directory>	Removes directory	NO
FSMD <directory>	Makes directory	NO
FSDELDIR <directory>	Deletes all files inside a directory	NO
FSEXIST <filename>	Verify existence of file inside a directory	NO
<b>FSEXIST &lt;filename&gt;</b>	<b>Search for specific file name inside a directory</b>	<b>NO</b>
FSGETFILE <protocol> <src filename>	Gets file from SD card	NO
FSFREE	Returns SD free SD card in KB	NO
FSRF <filename>	Removes file	NO
FSL	Lists files	NO
FSMKLOG <filename>	Creates a readable log file	NO
FSCLRLOG	Clears internal log file	NO
FSGETTIME	Gets the instrument's date and time	NO
FSSETTIME <dd> <mm> <yy> <hh> <mm> <ss>	Sets the instrument's date and time	NO
<b>TPSRCPATCH &lt;patch_number&gt; &lt;src_file_patched_address&gt; &lt;src_address&gt; &lt;size&gt;</b>	<b>Command for signature programming process</b>	<b>YES</b>
<b>Status Commands</b>		
SGETVER	Gets version	NO
SGETERR	Returns detailed error information	NO
SSETBR <baud rate>	Sets baud rate	NO
SGETSTS	Gets instrument status	NO
SPING	Pings instrument	NO
SGETCRC FILE <filename> <start addr> <len>	Calculates a 16-bit CRC	NO
SGETCRC DYNAMIC <start addr> <len>		
SGETCRC32 FILE <filename>	Calculates a 32-bit CRC	NO
SGETCRC32 DYNAMIC		
<b>Dynamic Memory Commands (not available on FlashRunner FR03 models)</b>		
DMSET <start addr> <len> <data> <data> ... <data>	Writes to dynamic memory	YES
DMGET <start addr> <len>	Reads from dynamic memory	YES
<b>Target Configuration Commands</b>		
TCSETDEV <manufacturer> <device> <algo>	Sets target device	YES
TCSETPAR <par name> <par value>	Sets target device parameter	YES
<b>Target Programming Commands</b>		
TPSTART	Starts programming sequence	YES
TPSETSRC FILE <filename>	Sets data source	YES
TPSETSRC DYNAMIC		
TPSETDST FILE <filename>	Sets data destination	YES
TPSETDST DYNAMIC		
TPCMD <command> [par1] [par2] ... [parn]	Executes programming	YES

## FlashRunner Commands

Command Syntax	Description	Scriptable
<b>command</b>		
TFEND	Ends programming sequence	YES
<b>Multiplexing Commands (only available in FlashRunner FR01M01 model)</b>		
MXOPEN	Opens all relays	YES
MXCLOSE <isp site>	Closes relays	YES
MXGET RELAYCYCLES <isp site>	Reads relays cycles	YES
<b>ATE Commands (only available in FR01A0 model)</b>		
LVDSET LPW <pulse width>	Low-voltage detector pulse width	YES
LVDSET LPT <threshold>	Low-voltage detector threshold	YES
LVDSET CH INT AIN0 <isp line>	Low-voltage detector line	YES
LVDSET ENABLE	Low-voltage detector enable	YES
LVDSET DISABLE	Low-voltage detector disable	YES
OCDSET ENABLE	Over-current detector enable	YES
OCDSET DISABLE	Over-current detector disable	YES
RLYGET CYCLES <isp line>	Reads relays cycles	YES
RLYGET DRVMODE	Gets the current relay driving mode	YES
RLYSET DRVMODE SW HW	Sets the relay driving mode	YES
RLYSET CLOSE <isp line>	Closes the specified relay	YES
RLYSET OPEN <isp line>	Opens the specified relay	YES
<b>Script Execution Commands</b>		
RUN <script number> RUN <script file>	Executes the specified script	NO
<b>Test Commands</b>		
TST WRWORD <register> <mask> <value> TST RDWORD <register> <mask>	Writes or reads test registers	YES
<b>Security Commands (for more information, see “Data Protection System” on page 151)</b>		
ASECURE	Enters the “Secure” mode	NO
AUNSECURE	Exit the “Secure” mode	NO
ACHGPWD ADMIN <old pw> <new pw>	Changes Administrator password	NO
ACHGUSER OPERATOR ACHGUSER ADMIN <pwd>	Switches “Secure” mode user	NO
DLOCK <pwd>	Locks SD card data	NO
DUNLOCK <pwd>	Unlocks SD card data	NO
<b>Master Controller Commands (Specific for FRPXIA3, FR3070A and Quattro Series Models)</b>		
MCRESET MC	Resets the master controller	NO
MCSETENG <module>	Sets the active programming module	NO
MCUPDATE <module>	Prepares for firmware update	NO
MCSPING	Pings master controller	NO
MCRESET ENGx	Resets the engine x	NO
MCGETVER	Gets version	NO
<b>Master Controller Commands (Specific for FR3070A and Quattro Series Models)</b>		
MCGETTIME	Gets date and time	NO

Command Syntax	Description	Scriptable
MCSETTIME <dd> <mm> <yy> <hh> <mm> <ss>	Sets date and time	NO
MCSETIP <ip1> <ip2> <ip3> <ip4>	Sets IP address	NO
MCSETNM <nm1> <nm2> <nm3> <nm4>	Sets network mask	NO
MCSETGW <gw1> <gw2> <gw3> <gw4>	Sets gateway address	NO
MCGETIP	Gets IP address	NO
MCGETNM	Gets network mask	NO
MCGETGW	Gets gateway address	NO
MCGETMAC	Gets MAC address	NO

## 2.4 Command Reference

Each FlashRunner command is listed alphabetically and explained in the following pages.

### 2.4.1 Command Documentation Conventions

The following conventions are used in the documentation of FlashRunner commands:

- Uppercase text indicates a command name or a command option that must be entered as shown.  
E.g. **SGETVER**
- Lowercase text between <> indicates a command parameter name.  
E.g. **SSETBR** <baud rate>
- Lowercase text between [] indicates an optional command parameter.  
E.g. **TPCMD** <command> [par1] [par2] ... [parn]
- A vertical bar indicates a choice between two or more command options.  
E.g. **TPCMD** **MASSERASE** **F|E|C**

### 2.4.2 ASECURE

**Command syntax:**

**ASECURE**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Enters the Data Protection System's Secured Mode. This command requires that the SD card contains a valid Data Protection System license.

After executing the **ASECURE** command for the first time, the Administrator password is set by default to **ADMIN**.



**Note:** *the Data Protection System feature is optional and requires the purchasing of a specific license. For more information on the Data Protection System, see "Data Protection System" on page 151.*

**Example:**

```
ASECURE  
>
```



### 2.4.3 AUNSECURE

**Command syntax:**

**AUNSECURE**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Exits from the Data Protection System's Secured Mode. The Administrator password is reset to its default value, **ADMIN**.



**Note:** *the Data Protection System feature is optional and requires the purchasing of a specific license. For more information on the Data Protection System, see "Data Protection System" on page 151.*

**Example:**

```
AUNSECURE  
>
```

### 2.4.4 ACHGPWD

#### Command syntax:

```
ACHGPWD ADMIN <old pwd> <new pwd>
```

#### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

#### Parameters:

**old pwd:** current Administrator password.  
**new pwd:** new Administrator password.

#### Answer data:

**Success:** the number of bytes requested.  
**Error:** the error code.

#### Description:

Changes the Data Protection System's Administrator password. This command is only available when the current user is Administrator. After executing the **ASECURE** command for the first time, the Administrator password is set by default to **ADMIN**. The password is a case-insensitive alphanumeric string.



**Note:** *the Data Protection System feature is optional and requires the purchasing of a specific license. For more information on the Data Protection System, see "Data Protection System" on page 151.*

#### Example:

```
ACHGPWD ADMIN MY_OLD_PWD MY_NEW_PWD  
>
```

## 2.4.5 ACHGUSER

### Command syntax:

```
ACHGUSER OPERATOR
ACHGUSER ADMIN <pwd>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

### Parameters:

**pwd:** Administrator password.

### Answer data:

**Success:** the number of bytes requested.  
**Error:** the error code.

### Description:

Switches to the Operator or Administrator user. This command is only available when in the Data Protection System's Secured Mode. The default password for Administrator is **ADMIN** (can be changed with the **ACHGPWD** command).



**Note:** *the Data Protection System feature is optional and requires the purchasing of a specific license. For more information on the Data Protection System, see "Data Protection System" on page 151.*

### Examples:

```
ACHGUSER ADMIN MY_PWD
>
ACHGUSER OPERATOR
>
```

### 2.4.6 DLOCK

**Command syntax:**

```
DLOCK <pwd>
```

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

**Parameters:**

**pwd:** encryption password (can be different from Administrator password).

**Answer data:**

**Success:** the number of bytes requested.

**Error:** the error code.

**Description:**

Locks (encrypts) the data on the SD card. This will make the SD card contents viewable/editable only by the Administrator and not by unauthorized people.

To unlock the SD card contents, use the **DUNLOCK** command.

This command is only available when the current user is Administrator.



**Note:** *the Data Protection System feature is optional and requires the purchasing of a specific license. For more information on the Data Protection System, see “Data Protection System” on page 151.*

**Example:**

```
DLOCK MY_PWD  
>
```

## 2.4.7 DMGET

### Command syntax:

```
DMGET <start addr> <len>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** Yes

### Parameters:

**start addr:** address of the dynamic memory to start retrieving data from.

**len:** number of bytes to retrieve.

### Answer data:

**Success:** the number of bytes requested.

**Error:** the error code.

### Description:

Retrieves **len** bytes from the dynamic memory starting from address **addr**. This command is available only on FlashRunner models featuring dynamic memory.

Dynamic memory is a special memory area (embedded in the FlashRunner electronics) which is typically used for storing temporary, variable data (e.g. serial numbers) before programming it to the target device. Dynamic memory is rather small (512 bytes) and retains its contents only as long as FlashRunner is powered.

### Example:

```
DMGET $0000 4  
$00 $01 $02 $03>
```

## 2.4.8 DMSET

### Command syntax:

```
DMSET <start addr> <len> <data> ... <data>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** Yes

### Parameters:

**start addr:** address of the dynamic memory to start writing data to.  
**len:** number of bytes to write.  
**data:** bytes to write.

### Answer data:

**Success:** none.  
**Error:** the error code.

### Description:

Writes **len** bytes to the dynamic memory starting at address **addr**. This command is available only on FlashRunner models featuring dynamic memory.

Dynamic memory is a special memory area (embedded in the FlashRunner electronics) which is typically used for storing temporary, variable data (e.g. serial numbers) before programming it to the target device. Dynamic memory is rather small (512 bytes) and retains its contents only as long as FlashRunner is powered.

### Example:

```
DMSET $0000 4 $00 $01 $02 $03  
>
```

## 2.4.9 DUNLOCK

### Command syntax:

```
DUNLOCK <pwd>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

### Parameters:

**pwd:** encryption password (can be different from Administrator password).

### Answer data:

**Success:** the number of bytes requested.

**Error:** the error code.

### Description:

Unlocks the data on the SD card, making its contents viewable/editable by everyone.

To lock the SD card contents, use the **DLOCK** command.

This command is only available when the current user is Administrator.



**Note:** *the Data Protection System feature is optional and requires the purchasing of a specific license. For more information on the Data Protection System, see "Data Protection System" on page 151.*

### Example:

```
DUNLOCK MY_PWD  
>
```

### 2.4.10 FSCD

#### Command syntax:

```
FSCD [..] | [[\]<directory>]
```

#### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

#### Parameters:

**directory:** directory to change to.

#### Answer data:

Success: none.  
Error: the error code.

#### Description:

Changes the current directory in the SD card file system as follows:

<b>FSCD ..</b>	goes up one level.
<b>FSCD \</b>	goes to the root directory.
<b>FSCD &lt;directory&gt;</b>	goes to the <directory> directory, relative to the current directory.
<b>FSCD \&lt;directory&gt;</b>	goes to the <directory> directory, relative to the root directory.

#### Examples:

```
FSCD ..  
>  
FSCD LOGS  
>  
FSCD \BINARIES  
>
```



## 2.4.11 FSCLRLOG

**Command syntax:**

**FSCLRLOG**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Resets FlashRunner's internal log file.

For performance reasons, during script execution, FlashRunner updates a log file in a reserved, non-readable area of the SD card. To generate a readable log file you must send the **FSMKLOG** command.

The **FSCLRLOG** command deletes all information from the internal, reserved log file.

For more information on logs, see "Logging" on page 135.

**Example:**

```
FSCLRLOG  
>
```

## 2.4.12 FSDELDIR

**Command syntax:**

```
FSDELDIR <directory>
```

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR04

**Scriptable:** No

**Parameters:**

**directory:** name of the directory in which all files has to be deleted.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Deletes all files within specified directory.

**Example:**

```
FSDELDIR BINARIES  
>
```

## 2.4.13 FSEXIST

### Command syntax:

```
FSEXIST <filename>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR04

**Scriptable:** No

### Parameters:

**filename:** name of the file to be searched within current directory

### Answer data:

Success: none.  
Error: the error code.

### Description:

Search for specific file name inside a directory.

### Example:

```
FSEXIST 123456binaryexample.frb  
>
```

## 2.4.14 FSFORMAT

**Command syntax:**

**FSFORMAT**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Initializes the SD card. Due to the fact that FlashRunner uses some areas of the card as private areas, a card must be properly initialized before it can be used. The **FSFORMAT** command does the following:

- Erases all files on the card;
- Formats the card with a FAT16 or a proprietary file system (depending on the FlashRunner model);
- Allocates private areas;
- Creates the appropriate directory structure needed for proper FlashRunner operation.



**Note:** *the only way to properly format a card is to use FlashRunner's **FSFORMAT** command. Cards formatted directly by a host system will not work in FlashRunner.*



**Note:** *FlashRunner is only able to format blank or previously FAT16 formatted cards. A card that is already FAT32 formatted cannot be reformatted by FlashRunner.*



**Note:** *after formatting a card, the `DEFAULT.ALG` file must be copied in the `\ALGOS` directory. This file can be found in the FlashRunner System Software CD.*

**Example:**

```
FSFORMAT  
>
```

## 2.4.15 FSFREE

**Command syntax:**

**FSFREE**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03, FR04

**Scriptable:** Yes

**Parameters:**

none

**Answer data:**

Free SD card space in Kb.

**Description:**

It returns free SD card space in Kb..

**Example:**

**FSFREE**

381152>

## 2.4.16 FSGETFILE

### Command syntax:

```
FSGETFILE <protocol> <src filename>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**protocol:** transfer protocol to use. Currently, only **YMODEM** is supported.

**src filename:** file to retrieve.

### Answer data:

**Success:** none.

**Error:** the error code.

### Description:

Copies a file stored in the current directory of the SD card to the host system.

The transfer is made using the Ymodem protocol. After sending the **FSGETFILE** command, FlashRunner prepares for the file transfer.

If you use the FlashRunner Control Panel, after sending the **FSGETFILE** command, a file transfer dialog box will appear automatically. Specify the local path where you want the file to be copied and Ymodem as the transfer protocol. If you use the HyperTerminal utility, after sending the **FSGETFILE** command, select **Transfer > Receive File** from the HyperTerminal utility. Specify the local path where you want the file to be copied and Ymodem as the transfer protocol. If you use another terminal utility, follow the equivalent transfer procedure.



**Note:** *if the SD card is locked and the current user is Operator, the file to be transferred is encrypted by FlashRunner with the password used to lock the SD card (for more information on the Data Protection System, see “Data Protection System” on page 151). Once the file has been received by the host system, it can be decrypted with the `fr_filesec` utility (see “Command-Line Utilities” on page 166).*

*The extension of an encrypted file is different from that of the original file: the first letter of an encrypted file extension is always a “s”. E.g.: an encrypted .FRB file has the .SRB extension, and encrypted .LOG file has the .SOG extension, etc.*

### Example:

```
FSGETFILE YMODEM MYLOG.TXT  
>
```

6



## 2.4.17 FSGETTIME

**Command syntax:**

**FSGETTIME**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: the instrument's date and time in the format  
dd mm yy hh mm ss.

Error: the error code.

**Description:**

Gets FlashRunner's internal clock.

**Example:**

```
FSGETTIME  
17 07 06 16 02 00>
```

## 2.4.18 FSLs

**Command syntax:**

FSLs

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: the current directory contents.

Error: the error code.

**Description:**

Lists the contents of the current directory in the SD card file system.

**Example:**

```
FSLs
Directory of ROOT\
SWAP.001          256491520  10/23/2006  16:22:06
ALGOS             [DIR]          10/23/2006  16:22:08
LICENSES         [DIR]          10/23/2006  16:22:08
SCRIPTS          [DIR]          10/23/2006  16:22:08
BINARIES         [DIR]          10/23/2006  16:22:08
LOGS             [DIR]          10/23/2006  16:22:08>
```

## 2.4.19 FSMD

### Command syntax:

```
FSMD <directory>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**directory:** directory to create.

### Answer data:

Success: none.  
Error: the error code.

### Description:

Creates a new directory in the SD card file system, relative to the current directory.

### Example:

```
FSMD TEST  
>
```

## 2.4.20 FSMKLOG

### Command syntax:

```
FSMKLOG <filename>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

### Parameters:

**filename:** name of the file that will contain logging information.

### Answer data:

**Success:** none.  
**Error:** the error code.

### Description:

Creates a log file with the specified file name.

For performance reasons, during script execution, FlashRunner updates a log file in a reserved, non-readable area of the SD card. To generate a readable log file you must send the **FSMKLOG** command. The file will be created (in plain text format) in the **\LOGS** directory.

Use the **FSCLRLOG** command to delete all information from the internal, reserved log file.

For more information on logs, see “Logging” on page 135.

### Example:

```
FSMKLOG PRJ1.LOG  
>
```

## 2.4.21 FSRD

### Command syntax:

```
FSRD <directory>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**directory:** directory to remove.

### Answer data:

Success: the file contents (see description below).

Error: the error code.

### Description:

Removes a directory from the SD card file system, relative to the current directory. The directory to be removed must be empty (must not contain files or subdirectories).

### Example:

```
FSRD TEST  
>
```

## 2.4.22 FSRF

### Command syntax:

```
FSRF <filename>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**filename:** file to remove.

### Answer data:

**Success:** none.  
**Error:** the error code.

### Description:

Removes a file from the current directory.

### Example:

```
FSRF IMAGE.FRB  
>
```

### 2.4.23 FSENDFILE

**Command syntax:**

```
FSENDFILE <protocol> <dst filename>
```

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

**Parameters:**

**protocol:** transfer protocol to use. Currently, only **YMODEM** is supported.

**filename:** destination filename.

**Answer data:**

**Success:** none.

**Error:** the error code.

**Description:**

Copies a file from the host system to the current directory in the FlashRunner's SD card.

The transfer is made using the Ymodem protocol. After sending the **FSENDFILE** command, FlashRunner prepares for the file transfer.

If you use the FlashRunner Control Panel, after sending the **FSENDFILE** command, a file transfer dialog box will appear automatically. Specify the file you want to transfer and Ymodem as the transfer protocol. If you use the HyperTerminal utility, after sending the **FSENDFILE** command, select **Transfer > Send File** from the HyperTerminal utility. Specify the file you want to transfer and Ymodem as the transfer protocol. If you use another terminal utility, follow the equivalent transfer procedure.



**Note:** *if the SD card is locked and the current user is Operator, the file to be transferred must be encrypted with the password used to lock the SD card (for more information on the Data Protection System, see “Data Protection System” on page 151). To encrypt the file, use the `fr_filesec` utility (see “Command-Line Utilities” on page 166).*

*The extension of an encrypted file is different from that of the original file: the first letter of an encrypted file extension is always a “s”. E.g.: an encrypted .FRB file has the .SRB extension, and encrypted .LOG file has the .SOG extension, etc.*

### Example:

```
FSENDFILE YMODEM IMAGE.FRB  
>
```

6



## 2.4.24 FSSETTIME

### Command syntax:

```
FSSETTIME <dd> <mm> <yy> <hh> <mm> <ss>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** No

### Parameters:

**dd:** day of the month (1 to 31).  
**mm:** month (1 to 12).  
**yy:** year (last two digits, 0 to 99).  
**hh:** hour (0 to 23).  
**mm:** minutes (0 to 59).  
**ss:** seconds (0 to 59).

### Answer data:

Success: none.  
Error: the error code.

### Description:

Sets FlashRunner's internal clock.

### Example:

```
FSSETTIME 17 07 06 16 02 00  
>
```

## 2.4.25 LVDSET CH

### Command syntax:

```
LVDSET CH INT VPROG0|VPROG1
LVDSET CH AIN0 <isp line>
```

### Compatibility:

FR01AT0

**Scriptable:** Yes

### Parameters and options:

**VPROG0:** The line to be monitored is VPROG0.  
**VPROG1:** The line to be monitored is VPROG1.  
**<isp line>:** The line to be monitored is connected to AIN0 and is logically of the type specified. Possible values are:

DIO0	Digital input/output 0
DIO1	Digital input/output 1
DIO2	Digital input/output 2
DIO3	Digital input/output 3
DIO4	Digital input/output 4
DIO5	Digital input/output 5
DIO6	Digital input/output 6
VPROG0	Programmable voltage 0
VPROG1	Programmable voltage 1
GENERIC	Generic line, not an ISP line driven by FlashRunner

### Answer data:

Success: none.  
Error: the error code.

**Description:**

Sets the line to be monitored by the low-voltage detector circuitry. The VPROG0 and VPROG1 lines can be monitored directly; all other ISP lines can be monitored through the AIN0 line (which must be therefore wasted for this purpose and connected to the line to be monitored).

The **LVDSET LPT** and **LVDSET LPW** commands must be also sent to specify the low-pulse voltage threshold and the low-pulse duration, respectively.

Use the **LVDSET ENABLE** and **LVDSET DISABLE** commands to turn the low-voltage detecting on or off, respectively.

**Examples:**

```
LVDSET LPT 2500
>
LVDSET LPW 5000
>
LVDSET CH INT VPROG0
>
LVDSET ENABLE
INT VPROG0 2500 5000>
```

```
LVDSET LPT 1500
>
LVDSET LPW 5000
>
LVDSET CH AIN0 DIO0
>
LVDSET ENABLE
AIN0 DIO0 1500 100>
```

## 2.4.26 LVDSET DISABLE

**Command syntax:**

`LVDSET DISABLE`

**Compatibility:**

FR01AT0

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Disables the low-voltage detector circuitry.

**Example:**

```
LVDSET DISABLE  
>
```

## 2.4.27 LVDSET ENABLE

**Command syntax:**

**LVDSET ENABLE**

**Compatibility:**

FR01AT0

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: a summary of the low-voltage detector settings.

Error: the error code.

**Description:**

Enables the low-voltage detector circuitry.

Before enabling the low-voltage detector circuitry, the **LVDSET LPW**, **LVDSET LPT** and **LVDSET CH** commands must be sent.

In case of success, the command returns four values (see also the example below):

1. The source line (**INT** or **AIN0**);
2. The ISP line;
3. The low-voltage threshold, in millivolts;
4. The low-voltage pulse duration, in nanoseconds.

Use the **LVDSET DISABLE** commands to turn the low-voltage detecting off.

If a low-voltage event is detected between a **LVDSET ENABLE** and a **LVDSET DISABLE** command, a low-voltage error will be returned by the first command causing the low-voltage event. Low-voltage events

## FlashRunner Commands

---

can be returned by the following commands: **TPSTART**, **TPCMD**, **TPEND**.



**Note:** *due to the internal precision of the low-voltage detect circuitry, the low-pulse duration and voltage threshold specified are automatically adjusted to the nearest values allowed by the system. The answer to the **LVDSET ENABLE** command includes the actual values that will be used by FlashRunner.*

### Example:

```
LVDSET LPT 2500
>
LVDSET LPW 5000
>
LVDSET CH INT VPROG0
>
LVDSET ENABLE
INT VPROG0 2500 5000>
```

## 2.4.28 LVDSET LPT

### Command syntax:

```
LVDSET LPT <threshold>
```

### Compatibility:

```
FR01AT0
```

**Scriptable:** Yes

### Parameters:

**threshold:** voltage threshold below which the signal is considered low, expressed in millivolts.

### Answer data:

**Success:** none.  
**Error:** the error code.

### Description:

Sets the voltage threshold to be used by the low-voltage detector circuitry.

The **LVDSET LPW** and **LVDSET CH** commands must be also sent to specify the low-pulse duration and the ISP channel to be monitored, respectively.

Use the **LVDSET ENABLE** and **LVDSET DISABLE** commands to turn the low-voltage detecting on or off, respectively.

### Example:

```
LVDSET LPT 2500
>
LVDSET LPW 5000
>
LVDSET CH INT VPROG0
>
LVDSET ENABLE
INT VPROG0 2500 5000>
```

## 2.4.29 LVDSET LPW

**Command syntax:**

```
LVDSET LPW <pulse width>
```

**Compatibility:**

```
FR01AT0
```

**Scriptable:** Yes

**Parameters:**

**pulse width:** minimum duration of the low pulse to detect, in nanoseconds.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Sets the minimum low pulse voltage duration to be detected by the low-voltage detector circuitry.

The **LVDSET LPT** and **LVDSET CH** commands must be also sent to specify the low-pulse voltage threshold and the ISP channel to be monitored, respectively.

Use the **LVDSET ENABLE** and **LVDSET DISABLE** commands to turn the low-voltage detecting on or off, respectively.

**Example:**

```
LVDSET LPT 2500
>
LVDSET LPW 5000
>
LVDSET CH INT VPROG0
>
LVDSET ENABLE
INT VPROG0 2500 5000>
```



### 2.4.30 MCGETGW

**Command syntax:**

MCGETGW

**Compatibility:**

FR3070A, FR04

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: the gateway of the instrument's master controller.

Error: the error code.

**Description:**

Gets the gateway of the instrument's master controller.

**Example:**

MCGETGW

192 168 1 1>

### 2.4.31 MCGETIP

**Command syntax:**

`MCGETIP`

**Compatibility:**

FR3070A, FR04

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: the IP address of the instrument's master controller.

Error: the error code.

**Description:**

Gets the IP address of the instrument's master controller.

**Example:**

```
MCGETIP
192 168 1 100>
```

## 2.4.32 MCGETMAC

**Command syntax:**

MCGETMAC

**Compatibility:**

FR3070A, FR04

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: the MAC address of the instrument's master controller.  
Error: the error code.

**Description:**

Gets the MAC address of the instrument's master controller.

**Example:**

```
MCGETMAC
00 50 FC A0 67 2C>
```

### 2.4.33 MCGETNM

**Command syntax:**

**MCGETNM**

**Compatibility:**

FR3070A, FR04

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: the network mask of the instrument's master controller.

Error: the error code.

**Description:**

Gets the network mask of the instrument's master controller.

**Example:**

```
MCGETNM  
255 255 255 0>
```

## 2.4.34 MCGETTIME

**Command syntax:**

`MCGETTIME`

**Compatibility:**

FR3070A, FR04

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: the instrument's date and time in the format  
dd mm yy hh mm ss.

Error: the error code.

**Description:**

Gets the master controller's date and time.

**Example:**

```
MCGETTIME  
17 07 06 16 02 00>
```

## 2.4.35 MCGETVER

**Command syntax:**

`MCGETVER`

**Compatibility:**

FR3070A, FR04, FRPXIA3

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: master controller information.

Error: the error code.

**Description:**

Gets master controller's information. In particular, gets the master controller's serial number.

**Example:**

```
MCGETVER
SN=78263 OS_VER=01.00.03.07 HW_VER=01.00.01
FR_MODEL=FR04A16>
```

## 2.4.36 MCRESET

**Command syntax:**

`MCRESET MC`

**Compatibility:**

FR3070A, FR04, FRPXIA3

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Resets the instrument's master controller.

**Example:**

```
MCRESET MC  
>
```

## 2.4.37 MCRESET ENGx

**Command syntax:**

**MCRESET ENGx**

**Compatibility:**

FR3070A, FR04, FRPXIA3

**Scriptable:** No

**Parameters:**

Number of Engine (1 to 4).

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Resets selected programming engine.

**Example:**

```
MCRESET ENG1  
>
```



## 2.4.38 MCSETENG

**Command syntax:**

```
MCSETENG <module>
```

**Compatibility:**

FR3070A, FR04, FRPXIA3

**Scriptable:** No

**Parameters:**

**module:** programming module. Can be one of the following values: **ENG1**, **ENG2**, **ENG3**, **ENG4**.

**Answer data:**

**Success:** none.  
**Error:** the error code.

**Description:**

Instructs the instrument to send all subsequent commands to the specified programming module.

**Example:**

```
MCSETENG ENG1  
>
```

## 2.4.39 MCSETGW

### Command syntax:

```
MCSETGW <gw1> <gw2> <gw3> <gw4>
```

### Compatibility:

FR3070A, FR04

**Scriptable:** No

### Parameters:

<b>gw1:</b>	Gateway byte 1.
<b>gw2:</b>	Gateway byte 2.
<b>gw3:</b>	Gateway byte 3.
<b>gw4:</b>	Gateway byte 4.

### Answer data:

Success:	none.
Error:	the error code.

### Description:

Sets the gateway of the instrument's master controller.

### Example:

```
MCSETGW 192 168 1 1  
>
```

## 2.4.40 MCSETIP

**Command syntax:**

```
MCSETIP <ip1> <ip2> <ip3> <ip4>
```

**Compatibility:**

FR3070A, FR04

**Scriptable:** No

**Parameters:**

<b>ip1:</b>	IP address byte 1.
<b>ip2:</b>	IP address byte 2.
<b>ip3:</b>	IP address byte 3.
<b>ip4:</b>	IP address byte 4.

**Answer data:**

Success:	none.
Error:	the error code.

**Description:**

Sets the IP address of the instrument's master controller.

**Example:**

```
MCSETIP 192 168 1 100  
>
```

## 2.4.41 MCSETNM

### Command syntax:

```
MCSETNM <nm1> <nm2> <nm3> <nm4>
```

### Compatibility:

FR3070A, FR04

**Scriptable:** No

### Parameters:

<b>nm1:</b>	Network mask byte 1.
<b>nm2:</b>	Network mask byte 2.
<b>nm3:</b>	Network mask byte 3.
<b>nm4:</b>	Network mask byte 4.

### Answer data:

Success:	none.
Error:	the error code.

### Description:

Sets the network mask of the instrument's master controller.

### Example:

```
MCSETNM 255 255 255 0  
>
```

## 2.4.42 MCSETTIME

### Command syntax:

```
MCSETTIME <dd> <mm> <yy> <hh> <mm> <ss>
```

### Compatibility:

FR3070A, FR04

**Scriptable:** No

### Parameters:

**dd:** day of the month (1 to 31).  
**mm:** month (1 to 12).  
**yy:** year (last two digits, 0 to 99).  
**hh:** hour (0 to 23).  
**mm:** minutes (0 to 59).  
**ss:** seconds (0 to 59).

### Answer data:

Success: none.  
Error: the error code.

### Description:

Sets the master controller's date and time.

### Example:

```
MCSETTIME 17 07 06 16 02 00  
>
```

### 2.4.43 MCSPING

**Command syntax:**

`MCSPING`

**Compatibility:**

FR3070A, FR04, FRPXIA3

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: `MCPONG`.  
Error: the error code.

**Description:**

Pings the instrument's master controller.

**Example:**

```
MCSPING
MCPONG>
```

## 2.4.44 MCUPDATE

**Command syntax:**

```
MCUPDATE <module>
```

**Compatibility:**

FR3070A, FR04

**Scriptable:** No

**Parameters:**

**module:** programming module. Can be one of the following values: **ENG1**, **ENG2**, **ENG3**, **ENG4**.

**Answer data:**

**Success:** none.  
**Error:** the error code.

**Description:**

Prepares for the updating of the firmware of the selected programming module. After sending this command,

**Example:**

```
MCUPDATE ENG1  
>
```

## 2.4.45 MXCLOSE

**Command syntax:**

```
MXCLOSE <isp site>
```

**Compatibility:**

```
FR01M01
```

**Scriptable:** Yes

**Parameters:**

**isp site:** ISP site to activate (1 to the maximum number of available ISP sites).

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Closes all relays on the specified ISP site, connecting FlashRunner to the target board on the specified ISP site.

**Example:**

```
MXCLOSE 1  
>
```



## 2.4.46 MXGET

**Command syntax:**

```
MXGET RELAYCYCLES <isp site>
```

**Compatibility:**

FR01M01

**Scriptable:** Yes

**Parameters:**

**isp site:** ISP site to get relay information from (1 to the maximum number of available ISP sites).

**Answer data:**

Success: the number of relay cycles.  
Error: the error code.

**Description:**

Retrieves the number of relay open/close cycles on the specified ISP site.

**Example:**

```
MXGET RELAYCYCLES 1  
5248>
```

## 2.4.47 MXOPEN

**Command syntax:**

**MXOPEN**

**Compatibility:**

FR01M01

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Opens all relays on all ISP sites, disconnecting FlashRunner from the target system(s).

**Example:**

```
MXOPEN  
>
```

## 2.4.48 OCDSET DISABLE

**Command syntax:**

OCDSET DISABLE

**Compatibility:**

FR01AT0

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Disables the over-current detector circuitry.

**Example:**

```
OCDSET DISABLE  
>
```

## 2.4.49 OCDSET ENABLE

**Command syntax:**

`OCDSET ENABLE`

**Compatibility:**

FR01AT0

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Enables the over-current detector circuitry. This special circuitry detects any over-current event that may arise on the VPROG0 or VPROG1 lines during programming.

If an over-current event is detected between a `OCDSET ENABLE` and a `OCDSET DISABLE` command, an over-current error will be returned by the first command causing the over-current event. Over-current events can be returned by the following commands:

`TPSTART`, `TPCMD`, `TPEND`.

**Example:**

```
OCDSET ENABLE  
>
```

## 2.4.50 RLYGET CYCLES

### Command syntax:

```
RLYGET CYCLES <isp line>
```

### Compatibility:

FR01A0

**Scriptable:** Yes

### Parameters:

**isp line:** ISP line to get relay information from. Can be one of the following values:

<b>AIN0</b>	Analog input 0
<b>CLKOUT</b>	Clock output
<b>DIO0</b>	Digital input/output 0
<b>DIO1</b>	Digital input/output 1
<b>DIO2</b>	Digital input/output 2
<b>DIO3</b>	Digital input/output 3
<b>DIO4</b>	Digital input/output 4
<b>DIO5</b>	Digital input/output 5
<b>DIO6</b>	Digital input/output 6
<b>VPROG0</b>	Programmable voltage 0
<b>VPROG1</b>	Programmable voltage 1
<b>GND</b>	Ground

### Answer data:

Success: the number of relay cycles.  
Error: the error code.

### Description:

Retrieves the number of relay operations on the specified ISP line.

## FlashRunner Commands

---

### Example:

```
RLYGET CYCLES DIO0  
7563>
```

## 2.4.51 RLYGET DRVMODE

**Command syntax:**

```
RLYGET DRVMODE
```

**Compatibility:**

```
FR01AT0
```

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: the current relay driving mode (**HW** or **SW**).

Error: the error code.

**Description:**

Retrieves the current relay driving mode.

The relay driving mode can be hardware (**HW**) or software (**SW**). In hardware mode, all relays are open or closed via the RELAY control line in the "CONTROL" connector. In software mode, relays can be individually driven through the **RLYSET OPEN** and **RLYSET CLOSE** commands.

To set the relay driving mode, use the **RLYSET DRVMODE** command.

**Example:**

```
RLYGET DRVMODE  
SW>
```

## 2.4.52 RLYSET DRVMODE

**Command syntax:**

```
RLYSET DRVMODE HW|SW
```

**Compatibility:**

```
FR01AT0
```

**Scriptable:** Yes

**Options:**

**HW:** Hardware relay driving mode.  
**SW:** Software relay driving mode.

**Answer data:**

**Success:** none.  
**Error:** the error code.

**Description:**

Sets the relay driving mode.

The relay driving mode can be hardware (**HW**) or software (**SW**). In hardware mode, all relays are open or closed via the RELAY control line in the “CONTROL” connector. In software mode, relays can be individually driven through the **RLYSET OPEN** and **RLYSET CLOSE** commands.

To get the current relay driving mode, use the **RLYGET DRVMODE** command.

**Example:**

```
RLYSET DRVMODE SW  
>
```



## 2.4.53 RLYSET CLOSE

### Command syntax:

```
RLYSET CLOSE <isp line>
```

### Compatibility:

FR01A0

**Scriptable:** Yes

### Parameters:

**isp line:** ISP line associated to the relay to close. Can be one of the following values:

<b>AIN0</b>	Analog input 0
<b>CLKOUT</b>	Clock output
<b>DIO0</b>	Digital input/output 0
<b>DIO1</b>	Digital input/output 1
<b>DIO2</b>	Digital input/output 2
<b>DIO3</b>	Digital input/output 3
<b>DIO4</b>	Digital input/output 4
<b>DIO5</b>	Digital input/output 5
<b>DIO6</b>	Digital input/output 6
<b>VPROG0</b>	Programmable voltage 0
<b>VPROG1</b>	Programmable voltage 1
<b>GND</b>	Ground
<b>ALL</b>	Closes all relays

### Answer data:

Success: none.  
Error: the error code.

### Description:

Closes the relay associated to the specified ISP line, connecting the specified ISP line to the target board.

## FlashRunner Commands

---

### Example:

```
RLYSET CLOSE DIO0  
>
```

6

## 2.4.54 RLYSET OPEN

### Command syntax:

```
RLYSET OPEN <isp line>
```

### Compatibility:

FR01AT0

**Scriptable:** Yes

### Parameters:

**isp line:** ISP line associated to the relay to open. Can be one of the following values:

<b>AIN0</b>	Analog input 0
<b>CLKOUT</b>	Clock output
<b>DIO0</b>	Digital input/output 0
<b>DIO1</b>	Digital input/output 1
<b>DIO2</b>	Digital input/output 2
<b>DIO3</b>	Digital input/output 3
<b>DIO4</b>	Digital input/output 4
<b>DIO5</b>	Digital input/output 5
<b>DIO6</b>	Digital input/output 6
<b>VPROG0</b>	Programmable voltage 0
<b>VPROG1</b>	Programmable voltage 1
<b>GND</b>	Ground
<b>ALL</b>	Opens all relays

### Answer data:

Success: none.  
Error: the error code.

### Description:

Opens the relay associated to the specified ISP line, connecting the corresponding I/O line from the ATE system to the target board.

## FlashRunner Commands

---

### Example:

```
RLYSET OPEN DIO0  
>
```

6

## 2.4.55 RUN

### Command syntax:

```
RUN <script number>  
RUN <script file>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**script number:** number of the script to execute. Valid script numbers are 0 to 31.

**script file:** script file to execute. Must include the **.FRS** extension.

### Answer data:

Success: OK or **ERR** (see the “Scripts” chapter on page 131 for more information).

Error: the error code.

### Description:

Executes the specified script number or script file. For more information on scripts, see the “Scripts” chapter on page 131.

### Examples:

```
RUN 0  
>  
RUN MYSCRIPT.FRS  
>
```

## 2.4.56 SGETCRC

### Command syntax:

```
SGETCRC FILE <filename> <start addr> <len>  
SGETCRC DYNAMIC <start addr> <len>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**filename:** name of the file in the SD card, in the current path.  
**start addr:** address of the first byte to include in the CRC calculation (in case of calculation of CRC from a file, this parameter is the offset from the beginning of the file).  
**len:** number of bytes to include in the CRC calculation.

### Answer data:

**Success:** the 16-bit CRC.  
**Error:** the error code.

### Description:

Calculates the 16-bit CRC of a contiguous set of bytes. The bytes can be taken either from a file or from the dynamic memory. This command is typically used to verify that a file has been transferred correctly from the host system to the SD card.

### Examples:

```
SGETCRC FILE IMAGE.FRB $0000 4096  
$F416>  
SGETCRC DYNAMIC $0000 256  
$C0EB>
```

## 2.4.57 SGETCRC32

### Command syntax:

```
SGETCRC32 FILE <filename>  
SGETCRC32 DYNAMIC
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**filename:** name of the file in the SD card, in the current path.

### Answer data:

**Success:** the 32-bit CRC.

**Error:** the error code.

### Description:

Calculates the 32-bit CRC of the contents of a specified file or of the dynamic memory. This command is typically used to verify that a file has been transferred correctly from the host system to the SD card.



**Note:** *the CRC32 is calculated using the CRC-32-IEEE802.3 polynomial:  $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$ , with an initial checksum of 0x04C11DB7.*

### Examples:

```
SGETCRC32 FILE IMAGE.FRB  
$F2F41630>  
SGETCRC32 DYNAMIC  
$39B1C0EB>
```

## 2.4.58 SGETERR

**Command syntax:**

`SGETERR`

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: info about the last error.

Error: the error code.

**Description:**

Returns detailed error information. Returns the code of the last error occurred and other error codes that may have triggered that error.

**Example:**

```
SGETERR
$0307
$1000>
```



## 2.4.59 SGETSTS

### Command syntax:

```
SGETSTS
SGETSTS V
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

None.

### Answer data:

Success: instrument status information.  
Error: the error code.

### Description:

Returns five decimal numbers describing FlashRunner status information. If the verbose version of the command is used (**SGETSTS V**), the status numbers are shown together with text messages explaining their meaning.

The first number returned is the FlashRunner status and mirrors the “STATUS” LED activity, as summarized in the table below.

Table 2: SGETSTS Command: FlashRunner Status

Value	Description
0	FlashRunner is ready
1	FlashRunner is busy updating the firmware
2	SD card not present
3	Invalid card contents (the card has not been formatted correctly or doesn't contain the required directory structure). Use the <code>FSFORMAT</code> command to reformat the card.
4	Hardware self-test error
5	SD card locked, invalid password

## FlashRunner Commands

---

The second number returned is the value of the currently selected script (it's the value of the SEL[4..0] lines in the "CONTROL" connector).

The third number returned represents the programming status, as described in the table below.

Table 3: SGETSTS Command: Programming Status

Value	Description
0	No programming has been performed yet ("BUSY", "PASS" and "FAIL" LEDs are off)
1	FlashRunner is busy programming ("BUSY" LED is on)
2	The previous programming has ended with errors ("FAIL" LED is on)
3	The previous programming has been successful ("PASS" LED is on)

The fourth number represents the currently active user (only makes sense if the Data System Protection system has been installed).

Table 4: SGETSTS Command: Active User

Value	Description
0	No currently active user
2	Current user is Operator
3	Current user is Administrator

The fifth number represents whether the SD card is locked (1) or unlocked (2).

### Example:

```
SGETSTS  
00 00 03 00 00>
```

## 2.4.60 SGETVER

**Command syntax:**

```
SGETVER  
SGETVER ALGO
```

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: version information.

Error: the error code.

**Description:**

Returns version information

**SGETVER** returns instrument serial number, firmware (OS) version and hardware version.

**SGETVER ALGO** returns the current programming algorithm name, manufacturer supported, and version information.

**Examples:**

```
SGETVER  
SN=000000000 OS_VER=01.00.00.00 HW_VER=01.00.00>  
SGETVER ALGO  
NEC78K NEC 01.00.05.03 04/02/08 22 794000 2>
```

## 2.4.61 SPING

**Command syntax:**

`SPING`

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

**Parameters:**

None.

**Answer data:**

Success: `PONG.`  
Error: the error code.

**Description:**

Pings the instrument. Used to verify whether FlashRunner is connected to the host system and running correctly.

**Example:**

```
SPING
PONG>
```

## 2.4.62 SSETBR

### Command syntax:

```
SSETBR <baud rate>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** No

### Parameters:

**baud rate:** new communication baud rate, in bits per second. Must be between 9600 and 115200.

### Answer data:

**Success:** none.  
**Error:** the error code.

### Description:

Changes the communication speed between the host system and FlashRunner. After executing this command (and sending the answer to the host system), FlashRunner will change its baud rate. Subsequent commands must be sent with the new baud rate.

By default, FlashRunner communicates at 115200 bps. After receiving a **SSETBR** command, FlashRunner saves the new speed in its internal, non-volatile memory and retains this setting even when powered down.

### Example:

```
SSETBR 115200  
>
```



**Note:** when using a LAN enabled version, FlashRunner must always work at 115200 bps (default value). Do not change this value.

### 2.4.63 TCSETDEV

#### Command syntax:

```
TCSETDEV <manufacturer> <device> <algo>
```

#### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** Yes

#### Parameters:

**manufacturer:** manufacturer name.  
**device:** device code.  
**algo:** programming algorithm to be used.

#### Answer data:

**Success:** none.  
**Error:** the error code.

#### Description:

Specifies the target manufacturer, device code and programming algorithm to be used. This command must be sent before any **TCSETPAR** command and before a **TPSTART / TPEND** command block.

When you buy a programming algorithm, you will get the precise values to use for the **manufacturer**, **device** and **algo** parameters.



**Note:** *when first executing a **TCSETDEV** command, FlashRunner loads the specified programming algorithm from the SD card into its internal memory. Therefore, the first execution of the **TCSETDEV** command is significantly slower than successive **TCSETDEV** commands with the same parameters.*

**Example:**

```
TCSETDEV FREESCALE MC68HC908QY4 HC08  
>
```

## 2.4.64 TPSRCPATCH

### Command syntax:

```
TPSRCPATCH <patch_number> <src_file_patched_address>  
<src_address> <size>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03, FR04

**Scriptable:** Yes

### Parameters:

**patch\_number:** identify a patch number. For each programming session there could be at most 20 patches.

**src\_file\_patched\_address:** source file address which will be modifies by the patch.

**src\_address:** addresses from which the patch data starts

**size:** patch size in bytes.

### Answer data:

### Description:

this command is useful for the signature programming process, especially when the device granularity is greater than programming size. FlashRunner will program the source file (specified previously by **TPSETSRC FILE** command) until it will find the address specified in each **TPSRCPATCH** command, from where it will start programming data specified (previously by **DMSET** command).

### Example:

```
TPSETSRC FILE IMAGE.FRB  
$F2F41630>
```



```
DMSET 0 $10 $AA $55 $AA $55 $AA $55 $AA $55 $AA $55 $AA  
$55 $AA $55 $AA $55
```

```
>
```

```
TPSRCPATCH 0 $000020 0 16
```

```
>
```

```
TPCMD PROGRAM F $0000 $0000 512
```

```
>
```

## 2.4.65 TCSETPAR

### Command syntax:

**TCSETPAR** <par name> <par value>

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** Yes

### Parameters:

**par name:** parameter name.

**par value:** parameter value.

### Answer data:

**Success:** none.

**Error:** the error code.

### Description:

Sets device-specific and programming algorithm-specific parameters. This command must be sent after the **TCSETDEV** command and before a **TPSTART** / **TPEND** command block.

See “Target-Specific Commands and Connections” on page 213 for more information.



**Note:** each **TPSTART** / **TPEND** command block must be preceded by the **TCSETPAR** commands required for your specific target device. The **TPEND** command resets any previously set device-specific and programming algorithm-specific parameters.

**Example:**

```
TCSETDEV FREESCALE MC68HC908QY4 HC08
>
TCSETPAR FOSC 4000000
>
TCSETPAR VDD 3300
>
```

## 2.4.66 TPCMD

### Command syntax:

```
TPCMD <command> [par1] [par2] ... [parn]
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** Yes

### Parameters:

**command:** programming command.  
**par:** zero or more programming command parameters.

### Answer data:

**Success:** programming command specific.  
**Error:** the error code.

### Description:

Performs a programming operation (i.e. mass erase, program, verify, etc.) This command must be sent within a **TPSTART/TPEND** command block. Programming commands and their relative parameters are device-specific. See “Target-Specific Commands and Connections” on page 213 for more information.

### Example:

```
TPSTART  
>  
TPSETSRC FILE IMAGE.FRB  
$F2F41630>  
TPCMD PROGRAM F $0000 $0000 512  
>  
TPEND  
>
```

## 2.4.67 TPEND

**Command syntax:**

**TPEND**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Ends a programming block. This command must be preceded by a **TPSTART** command. **TPCMD** commands must be sent within a **TPSTART/TPEND** command block.

The **TPEND** command puts all of the FlashRunner programming lines in three-state or ground, depending on the target device.



**Note:** *if, after executing a **TPSTART** command, a script ends before reaching a **TPEND** command, the next **TPEND** command is executed all the same (skipping all commands in between). This in order to put all programming lines in three-state or ground. For more information about scripts, please refer to the "Scripts" chapter on page 131.*



**Note:** each *TPSTART* / *TPEND* command block must be preceded by the *TCSETPAR* commands required for your specific target device. The *TPEND* command resets any previously set device-specific and programming algorithm-specific parameters.

### Example:

```
TPSTART
>
TPSETSRC FILE IMAGE.FRB
$F2F41630>
TPCMD PROGRAM F $0000 $0000 512
>
TPEND
>
```

## 2.4.68 TPSETDST

### Command syntax:

```
TPSETSRC FILE <filename>
TPSETSRC DYNAMIC
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO

**Scriptable:** Yes

### Parameters:

**filename:** name of the file in the SD card (in the \BINARIES directory).

### Answer data:

**Success:** none.  
**Error:** the error code.

### Description:

Sets which destination will be used to save data retrieved in the in subsequent **TPCMD** commands. Typically used with DUMP command.

### Example:

```
TPSETDST DYNAMIC
>
TPCMD DUMP F $0000 $0000 4
>
```

## 2.4.69 TPSETSRC

### Command syntax:

```
TPSETSRC FILE <filename>
TPSETSRC DYNAMIC
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** Yes

### Parameters:

**filename:** name of the file in the SD card (in the `\BINARIES` directory).

### Answer data:

**Success:** the 32-bit CRC of the specified file or dynamic memory.  
**Error:** the error code.

### Description:

Sets the source of data to be used in subsequent `TPCMD` commands. Data can be taken either from a file (in the `\BINARIES` directory) or from the dynamic memory.

### Example:

```
TPSETSRC FILE IMAGE.FRB
$F2F41630>
TPCMD PROGRAM F $0000 $0000 512
>
DMSET $0000 4 $00 $01 $02 $03
>
TPSETSRC DYNAMIC
$39B1C0EB>
TPCMD PROGRAM F $0000 $0100 4
>
```



## 2.4.70 TPSTART

**Command syntax:**

**TPSTART**

**Compatibility:**

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** Yes

**Parameters:**

None.

**Answer data:**

Success: none.  
Error: the error code.

**Description:**

Starts a programming block. To end a programming block, send the **TPEND** command. **TPCMD** commands must be sent within a **TPSTART/TPEND** command block.

The **TPSTART** command performs some internal initializations and prepares FlashRunner to execute subsequent **TPCMD** commands.

**Example:**

```
TPSTART
>
TPSETSRC FILE IMAGE.FRB
$F2F41630>
TPCMD PROGRAM F $0000 $0000 512
>
TPEND
>
```

## 2.4.71 TST

### Command syntax:

```
TST WRWORD <register> <mask> <value>
TST RDWORD <register> <mask>
```

### Compatibility:

FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03

**Scriptable:** Yes

### Parameters:

**register:** name of the register associated with the hardware lines to test. Can be one of the following values:

<b>AIN0D</b>	AIN0 Data Register
<b>AO0D</b>	AO0 Data Register
<b>AO1D</b>	AO1 Data Register
<b>AORD</b>	AO Reduce Driving Register
<b>CLK</b>	Clock Register
<b>DIOD</b>	DIO Data Register
<b>DIODD</b>	DIO Data Direction Register
<b>DIOFR0</b>	DIO Function Register 0
<b>DIOFR1</b>	DIO Function Register 1
<b>DIOPU</b>	DIO Pull-Up Register
<b>VPROG0D</b>	VPROG0 Data Register
<b>VPROG0OM</b>	VPROG0 Output Mode Register
<b>VPROG1D</b>	VPROG1 Data Register
<b>VPROG1OM</b>	VPROG1 Output Mode Register

**mask:** 16-bit value which acts as a “mask”. This parameter is combined with a bitwise AND operation with the value to be written to or read from the specified **register**. Specify **\$FFFF** to write/read all of the bits to/from the specified **register**.

**value:** 16-bit value to be written to the specified **register**.

**Answer data:**

Success: none for the **TST WRWORD** command. The value read for the **TST RDWORD** command.

Error: the error code.

**Description:**

Reads or writes a 16-bit value on the specified FlashRunner register.



**Note:** *before using the **TST WRWORD** command, the **TCSETDEV SOFTEC DEFAULT DEFAULT** command must have been sent and the SD card must contain the **DEFAULT.ALG** programming algorithm.*

*The **TST RDWORD** command is always available.*

The following figure explains how the various registers affect the FlashRunner ISP lines.

## FlashRunner Commands

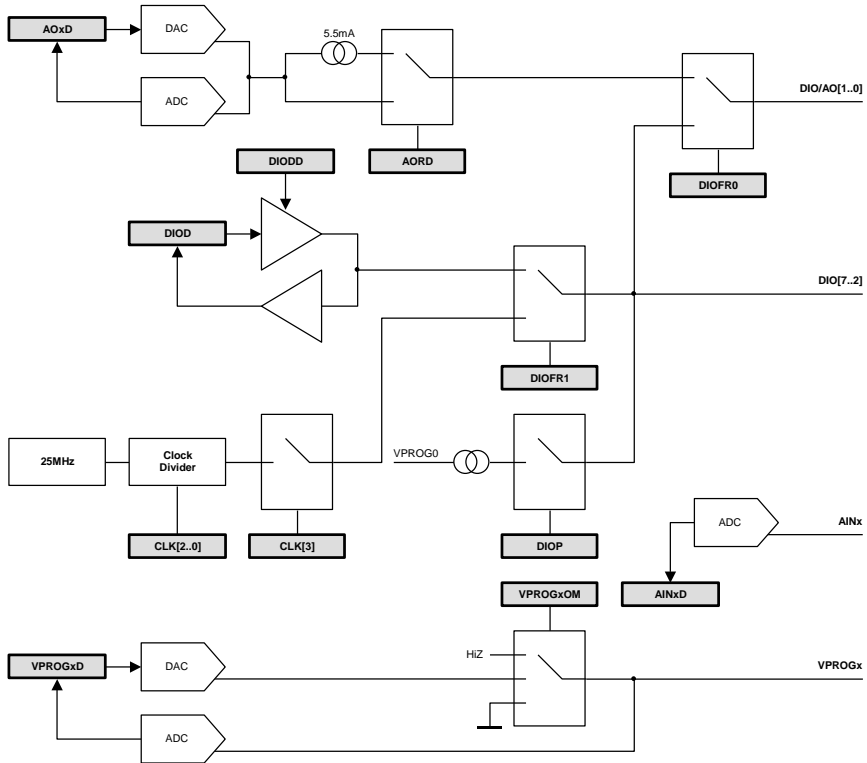


Figure 4: FlashRunner Registers

The following pages explain each register in detail.

### Legend:

X = Don't care

U = Undefined

= Not permitted

**AIN0D (AIN0 Data Register)**

Contains the voltage of the AIN0 analog line, expressed in mV. Read-only register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Write:																
Reset:	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0

**AOxD (AOx Data Register)**

Sets or reads the voltage of the AOx analog line, expressed in mV. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Write:	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Reset:	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U

**AORD (AO Reduce Driving Register)**

Enables or returns the presence of a limiting current circuitry (5.5mA) on the AO0/AO1 analog output lines. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	AO1RD	AO0RD
Write:	X	X	X	X	X	X	X	X	X	X	X	X	X	X	AO1RD	AO0RD
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	U	U

AOxRD – AOx Reduce Driving Enable Bit

0 = Disabled

1 = Enabled

## FlashRunner Commands

### CLK (Clock Register)

Enables an internal clock signal generator (INT\_CLK) with a specified frequency. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	0	0	0	0	CLKENA	CLK2	CLK1	CLK0
Write:	X	X	X	X	X	X	X	X	X	X	X	X	CLKENA	CLK2	CLK1	CLK0
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CLKENA – Clock Enable Bit

0 = Disabled

1 = Enabled

CLK[2:0] – Clock Frequency

CLK2	CLK1	CLK0	Frequency
0	0	0	25 MHz
0	0	1	12.5 MHz
0	1	0	6.25 MHz
0	1	1	3.125 MHz
1	0	0	1.5625 MHz
1	0	1	781.25 KHz
1	1	0	390.625 KHz
1	1	1	97.656 KHz

6

### DIOD (DIO Data Register)

Sets or reads the value of the DIO[7..0] lines. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	0	DIO6D	DIO5D	DIO4D	DIO3D	DIO2D	DIO1D	DIO0D
Write:	X	X	X	X	X	X	X	X	DIO7D	DIO6D	DIO5D	DIO4D	DIO3D	DIO2D	DIO1D	DIO0D
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DIOxD – DIOx Data

0 = DIOx line driven low

1 = DIOx line driven high

### DIODD (DIO Data Direction Register)

Sets or reads the data direction of the DIO[7..0] lines. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	DIO7DD	DIO6DD	DIO5DD	DIO4DD	DIO3DD	DIO2DD	DIO1DD	DIO0DD
Write:	X	X	X	X	X	X	X	X	DIO7DD	DIO6DD	DIO5DD	DIO4DD	DIO3DD	DIO2DD	DIO1DD	DIO0DD
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

DIOxDD – DIOx Data Direction

0 = DIOx line set as input

1 = DIOx line set as output

### DIOFR0 (DIO Function Register 0)

Sets the DIO0 and/or DIO1 lines as digital or analog lines. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	AO1F	AO0F
Write:	X	X	X	X	X	X	X	X	X	X	X	X	X	X	AO1F	AO0F
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	U	U

#### AOxF – AOx Function

0 = Digital

1 = Analog

### DIOFR1 (DIO Function Register 1)

Sets the DIO[7..0] output mode as digital or clock output. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	DIO7F	DIO6F	DIO5F	DIO4F	DIO3F	DIO2F	DIO1F	DIO0F
Write:	X	X	X	X	X	X	X	X	DIO7F	DIO6F	DIO5F	DIO4F	DIO3F	DIO2F	DIO1F	DIO0F
Reset:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

#### DIOxF – DIOx Function

0 = DIOx line set as digital line

1 = DIOx line set as clock output signal

### DIOPU (DIO Pull-Up Register)

Enables or disables an active pull-up on the DIO[6..0] lines. Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	0	DIO6P	DIO5P	DIO4P	DIO3P	DIO2P	DIO1P	DIO0P
Write:	X	X	X	X	X	X	X	X	X	DIO6P	DIO5P	DIO4P	DIO3P	DIO2P	DIO1P	DIO0P
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

#### DIOxP – DIOx Pull-Up

0 = DIOx pull-up disabled

1 = DIOx pull-up enabled

### VPROGxD (VPROGx Data Register)

Sets or reads the voltage of the VPROGx line, expressed in mV. Read/write register.

## FlashRunner Commands

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Write:	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Reset:	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U

### VPROGxOM (VPROGx Output Mode Register)

Sets or reads the output mode of the VPROGx line (high impedance, ground, or the value specified in the VPROGxD register). Read/write register.

	Bit15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Bit0
Read:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	VOM1	VOM0
Write:	X	X	X	X	X	X	X	X	X	X	X	X	X	X	VOM1	VOM0
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	U	U

VOM[1..0] – VPROG Output Mode

VOM1	VOM0	Output Mode
0	0	HIZ
0	1	GND
1	0	VPROGxD Register
1	1	Invalid

6

**Script Example 1 (drives the DIO0 line to 1 and then to 0):**

```

; Loads the Test programming algorithm
TCSETDEV SOFTEC DEFAULT DEFAULT
; Sets the DIO0 line as digital
TST WRWORD DIOFRO $0001 0
; Sets the DIO0 line as output
TST WRWORD DIODD $0001 1
; Outputs a high logic value on DIO0
TST WRWORD DIOD $0001 1
; Outputs a low logic value on DIO0
TST WRWORD DIOD $0001 0

```

**Script Example 2 (outputs a 97.656 KHz clock signal to the DIO0 line):**

```

; Loads the Test programming algorithm
TCSETDEV SOFTEC DEFAULT DEFAULT
; Sets and enables a 97.656 KHz clock frequency
TST WRWORD CLK $000F $000F
; Sets the DIO0 line as clock output signal
; and sets the DIO7 line as digital line
; (by default, the DIO7 line is always set

```



```
; as clock output signal)
TST WRWORD DIOFR1 $0081 $0001
```

**Script Example 3 (outputs 5V to the DIO0 line and reads it back):**

```
; Logs command answers
#LOG_ON 2
; Loads the Test programming algorithm
TCSETDEV SOFTEC DEFAULT DEFAULT
; Writes a 5V value to the A00 Data Register
TST WRWORD A00D $FFFF 5000
; Sets the DIO0 line as analog output
TST WRWORD DIOFR0 $0001 1
; Reads back the value
TST RDWORD A00D $FFFF
; The log file will contain the read value
```

**Script Example 4 (outputs 4V to the VPROG0 line and reads it back):**

```
; Logs command answers
#LOG_ON 2
; Loads the Test programming algorithm
TCSETDEV SOFTEC DEFAULT DEFAULT
; Write a 4V value on the VPROG0 Data Register
TST WRWORD VPROG0D $FFFF 4000
; Drives the VPROG0 line
TST WRWORD VPROG0OM $0003 2
; Reads back the VPROG0 voltage
TST RDWORD VPROG0D $FFFF
; The log file will contain the read value
```

**Script Example 5 (reads the AIN0 voltage):**

```
; Logs command answers
#LOG_ON 2
; (Only a TST RDWORD command will be executed,
; the TCSETDEV SOFTEC DEFAULT DEFAULT command
; is not necessary)
; Reads the AIN0 voltage
TST RDWORD AIN0D $FFFF
; The log file will contain the read value
```

6

## 3 Scripts

### 3.1 Overview

Scripts are text files, stored in the SD card, which contain a sequence of FlashRunner commands. Scripts are used when working in Standalone mode. The figure below shows an example of script.

```
;
; FLASHRUNNER SCRIPT EXAMPLE FOR FREESCALE MC68HC908QY4
;
; Use this example as a starting point for your specific programming needs
;
; Turns off logging
#LOG_OFF

; Halt on errors
#HALT_ON FAIL

; Sets device
TCSETDEV FREESCALE MC68HC908QY4 HC08

; Oscillator frequency, Hz (change as needed)
TCSETPAR FOSC 16000000

; Oscillator divisor (change as needed)
TCSETPAR FDIV 4

; Target voltage, mV (change as needed)
TCSETPAR VDD 5000

; Power-down time, ms (change as needed)
TCSETPAR PWDOWN 10

; Power-up time, ms (change as needed)
TCSETPAR PWUP 10

; Image file to be programmed (must be placed in the \BINARIES directory)
TPSETSRC FILE TEST.FRB

; Starts programming block
TPSTART

; Security bytes needed to perform subsequent operations (change as needed)
TPCMD SETPWD CONST $FF $FF $FF $FF $FF $FF $FF $FF

; Mass erases Flash memory
TPCMD MASSERASE F

; Blank checks Flash memory (change address and length as needed)
TPCMD BLANKCHECK F $EE00 4608

; Trims internal oscillator (change frequency and trimming location as needed)
TPCMD TRIM 3200000 $FFC0

; Programs Flash memory (change addresses and length as needed)
TPCMD PROGRAM F $EE00 $EE00 4608

; Verifies Flash memory, read-out method (change addresses and length as needed)
TPCMD VERIFY F R $EE00 $EE00 4608

; Ends programming block
TPEND
```

Figure 5: Script Example

Each script line may contain a FlashRunner command, a script directive, a comment, or nothing.



**Note:** *not all FlashRunner commands are scriptable. For a list of scriptable commands, see Command Summary on page 37.*

## 3.2 Execution and Termination

### 3.2.1 Standalone Script Execution

On FlashRunner models with script selection lines, a group of control lines (SEL[4..0] in the “CONTROL” Connector) determines which of a set of 32 scripts (named SCRIPT0.FRS, SCRIPT1.FRS, ..., SCRIPT31.FRS) will be executed. The status of the SEL control lines is indicated by the “SEL” LEDs.

On FlashRunner models with no script selection lines, only one script (which must be named SCRIPT0.FRS) is available.

Script files must be stored in the SD card in the “SCRIPTS” directory.

The event that triggers script execution is the START control line becoming active (while the BUSY line is not active). This line can be driven either manually (for example, by the operator through a push-button) or automatically by the ATE control logic.

When FlashRunner begins executing a script, the “BUSY” LED turns on. During script execution FlashRunner will not answer host commands.

The following diagram illustrates typical temporal relations between the various FlashRunner control lines. Please note that all control lines are active low.

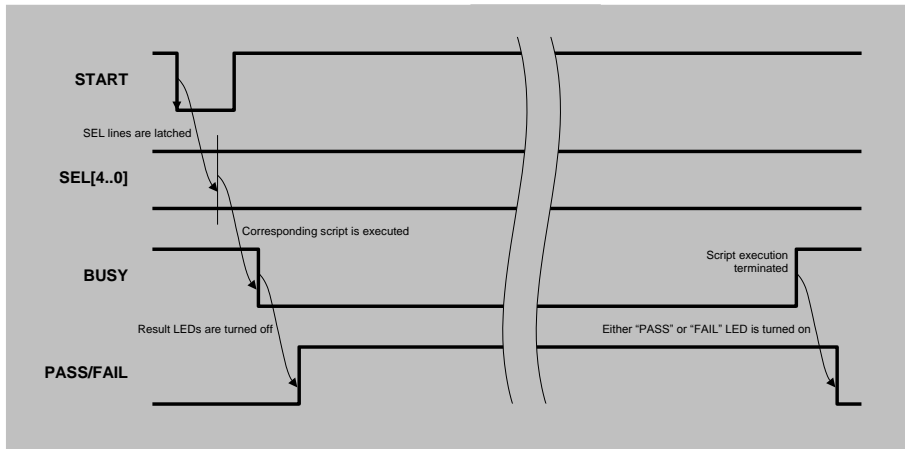


Figure 6: Script Execution and Control Lines

# 6

## 3.2.2 Remote Script Execution

Additionally, scripts can be manually executed in host mode. The `RUN` command (see page 66) executes a specified script file.

Script files must be stored in the SD card in the "SCRIPTS" directory.

## 3.2.3 Script Termination

Script execution ends either after FlashRunner has executed the last command in the script, or immediately after the first command in the script that is not successful.

When script execution ends, the "BUSY" LED turns off and either the "PASS" or "FAIL" LED turns on, depending on whether the last command executed was successful or not.



**Note:** the `#HALT_ON` directive (see later in this chapter) can be used inside scripts to instruct FlashRunner to ignore errors.

## 3.3 Script-Specific Directives

FlashRunner commands contained in a script file are executed sequentially, exactly as they would be executed in Host mode. However, script files contain additional directives (not available in Host mode) which control how scripts are executed. The following table lists these directives.

Table 5: Script-Specific Directives

Directive Syntax	Description
#DELAY <time ms>	Interrupts script execution for the specified amount of time
#ERR_ON FAIL, #HALT_ON FAIL	Stops script execution when a FlashRunner command causes an error (default behaviour)
#ERR_ON PASS, #HALT_ON PASS	Stops script execution when a FlashRunner command is executed successfully
#ERR_ON NEVER, #HALT_ON NEVER	Never stops script execution, even if errors occur
#LOG_ON <level>	Turns on logging and sets logging level
#LOG_OFF	Turns off logging
#PRINT	Outputs one or more strings or bytes (either prints the data to the log file or sends it to the host system)
#WAIT START STOP	Waits for the START or STOP control line to be activated

Each directive is valid from its script line forward. A script file may contain zero or more directives.

### 3.3.1 Logging

On some FlashRunner models, script logging is available.

The **#LOG\_ON** directive turns on script logging. Logging ends either when a **#LOG\_OFF** directive is encountered or (if no **#LOG\_OFF** directive is encountered) when the script ends.

Two levels of logging are available.

- The **#LOG\_ON 1** directive turns on basic logging. Two lines are written to the log file: one when the **#LOG\_ON** directive is encountered (containing the **START LOGGING** field) and one when the next **#LOG\_OFF** directive is encountered (or when the script ends, if no **#LOG\_OFF** directive is encountered), containing the **END LOGGING** field. The figure below shows an example of basic logging.

```
SCRIPT1.FRS:020:22.01.07.12.20.53.93:START LOGGING
SCRIPT1.FRS:057:22.01.07.12.20.54.60:END LOGGING:OK
```

Figure 7: Basic Logging

- The **#LOG\_ON 2** directive turns on advanced logging. In addition to the basic logging, execution of each line between the **#LOG\_ON** directive and the next **#LOG\_OFF** directive (or the end of the script) is also logged. The figure below shows an example of advanced logging.

```
SCRIPT1.FRS:020:22.01.07.12.17.19.92:START LOGGING
SCRIPT1.FRS:025:22.01.07.12.17.19.92:TCSETDEV FREESCALE MC68HC908QY4 HC08:>
SCRIPT1.FRS:028:22.01.07.12.17.19.93:TCSETPAR FOSC 16000000:>
SCRIPT1.FRS:031:22.01.07.12.17.19.93:TCSETPAR FDIV 4:>
SCRIPT1.FRS:034:22.01.07.12.17.19.93:TCSETPAR VDD 5000:>
SCRIPT1.FRS:037:22.01.07.12.17.19.94:TCSETPAR PWDOWN 10:>
SCRIPT1.FRS:040:22.01.07.12.17.19.94:TCSETPAR PWUP 10:>
SCRIPT1.FRS:043:22.01.07.12.17.19.94:TPSETSRC FILE TEST.FRB:$F2F41630>
SCRIPT1.FRS:046:22.01.07.12.17.20.28:TPSTART:>
SCRIPT1.FRS:049:22.01.07.12.17.20.29:TPCMD SETPWD CONST $FF $FF $FF $FF $FF $FF $FF $FF:>
SCRIPT1.FRS:052:22.01.07.12.17.20.29:TPCMD MASSERASE F:>
SCRIPT1.FRS:055:22.01.07.12.17.20.59:TPCMD BLANKCHECK F $EE00 4608:>
SCRIPT1.FRS:058:22.01.07.12.17.20.91:TPCMD TRIM 3200000 $FFC0:$69>
SCRIPT1.FRS:061:22.01.07.12.17.21.01:TPCMD PROGRAM F $EE00 $EE00 4608:>
SCRIPT1.FRS:064:22.01.07.12.17.21.83:TPCMD VERIFY F R $EE00 $EE00 4608:>
SCRIPT1.FRS:067:22.01.07.12.17.22.06:TPEND:>
SCRIPT1.FRS:069:22.01.07.12.17.22.07:END LOGGING:OK
```

Figure 8: Advanced Logging

The **#LOG\_OFF** directive turns off script logging. When encountering this directive (or when reaching the end of the script) FlashRunner writes a line in the log file containing the **END LOGGING** field, followed by a result field (**OK** or **ERR**) which reflects the result of the last command executed. Please note that the **#LOG\_OFF** directive is ignored if no **#LOG\_ON** directives were encountered before.



A line of a log file contains several fields separated by the `:` character. These fields are, from left to right:

1. The script file;
2. The script line (a three-digit decimal number with leading zeros);
3. A timestamp (in the format `dd.mm.yy.hh.mm.ss.ss`);
4. Command executed, or:
  - **START LOGGING** (in correspondence to the **#LOG\_ON** directive);
  - **END LOGGING** (in correspondence to the **#LOG\_OFF** directive or end of script);
  - **PRINT** (in correspondence to the **#PRINT LOG** directive).
5. Command answer, or:
  - Nothing (in correspondence to the **#LOG\_ON** directive);
  - Result of previous command (**OK** or **ERR**) (in correspondence to the **#LOG\_OFF** directive or end of script);
  - Data output by the **#PRINT** directive.

Scripts may contain zero, one or several **#LOG\_ON** / **#LOG\_OFF** directive pairs, in order to log nothing, the entire script or only portions of a script.

Comments, empty lines and script directives other than **#LOG\_ON**, **#LOG\_OFF** and **#PRINT LOG** are not logged.



**Note:** for performance reasons, FlashRunner creates the log file in a reserved, non-readable area of the SD card. To generate a readable log file, you must send the **FSMKLOG** command (see page 60). Use the **FSCLRLOG** command (see page 49) to delete all information from the internal, reserved log file.

### 3.3.2 Error Handling (ERR\_ON and HALT\_ON directives)

When FlashRunner commands are executed from within a script, no answer is sent to the host system, regardless of whether they are successful or not. If logging is enabled, errors are logged into the log file.

The **#ERR\_ON** and **#HALT\_ON** script directives instruct FlashRunner how to handle errors during script execution.

The **#ERR\_ON** directive stops script execution and signals an error as follows:

- The **#ERR\_ON FAIL** directive causes script execution to stop immediately after a command generates an error. In this case, the “FAIL” LED turns on. This is the default setting.
- The **#ERR\_ON PASS** directive causes script execution to stop immediately after a command is successful. In this case, the “FAIL” LED turns on.
- Finally, the **#ERR\_ON NEVER** directive instructs FlashRunner to never stop script execution (errors are ignored). In this case, the “PASS” LED will turn on at the end of script execution.

The **#HALT\_ON** directive behaves slightly differently and stops script execution and signals an error as follows:

- The **#HALT\_ON FAIL** directive causes script execution to stop immediately after a command generates an error. In this case, the “FAIL” LED turns on. This is the default setting.
- The **#HALT\_ON PASS** directive causes script execution to stop immediately after a command is successful. In this case, the “PASS” LED turns on.
- Finally, the **#HALT\_ON NEVER** directive instructs FlashRunner to never stop script execution (errors are ignored). Depending on whether errors were encountered during script execution, the “PASS” or “FAIL” LED will turn on at the end of script execution.

The **#ERR\_ON** directive setting has priority over the **#HALT\_ON** directive setting. The following table summarizes the behaviour of the two directives.

Table 6: ERR\_ON and HALT\_ON Directives Summary

Directive	Description	Script Execution Result (LED turned on at end of script execution)
#ERR_ON FAIL	Stops script execution when a FlashRunner command causes an error (default behaviour)	FAIL
#ERR_ON PASS	Stops script execution when a FlashRunner command is executed successfully	FAIL
#ERR_ON NEVER	Never stops script execution, even if errors occur	PASS
#HALT_ON FAIL	Stops script execution when a FlashRunner command causes an error (default behaviour)	FAIL
#HALT_ON PASS	Stops script execution when a FlashRunner command is executed successfully	PASS
#HALT_ON NEVER	Never stops script execution, even if errors occur	PASS if no error occurred, FAIL otherwise

### 3.3.3 DELAY and WAIT Directives

The **#DELAY** and **#WAIT** directives temporarily interrupt the execution of a script.

- The **#DELAY** directive interrupts script execution for a specified amount of time (expressed in milliseconds).
- The **#WAIT START** and **#WAIT STOP** directives interrupt script execution until the START or STOP control line (respectively) is activated (either manually by the user or automatically by the ATE control logic).

### 3.3.4 PRINT Directive

The **#PRINT** directive outputs one or more strings or bytes either to the log file or to the host system. The **#PRINT** directive syntax is the following:

```
#PRINT LOG|COM|LAN STR|CHR <data1> [data2] ... [datan]
```

Where:

- **LOG**, **COM** or **LAN** is the output channel (log file, RS-232 or Ethernet, respectively);
- **STR** or **CHR** is the data type (string or byte);

## Scripts

---

- `data1`, `data2`, `dataN` is the actual data to be output. One or more strings, or one or more bytes, can be output. Strings must be enclosed by quotation marks.

The following example will print a string to the log file:

```
#PRINT LOG STR "This will be written to the log file."
```

The following example will send three bytes (\$40, \$41 and \$42) to the host system through the RS-232 channel:

```
#PRINT COM CHR $40 $41 $42
```

Please note that, unlike other FlashRunner commands, neither a response character (> or !) nor the final carriage return character is appended to the output data.

# 6

## 3.4 Comments

A script line may contain a comment. A comment is a line of text preceded by the `;` character. Comments are ignored by FlashRunner.

## 3.5 Control Flow on FlashRunner Series

### 3.5.1 Introduction

With the aim of raising the flexibility and the customization degree of FlashRunner series, script syntax has been improved starting from **01.04.08.00** OS version, which implements low level commands able to control the flow of the script's command.

Syntax that is used gets back to the classical programming languages and shall be immediately clear to all the users familiars with them, because it reproduces the *if*, *then*, *else* statement.

In fact, in "C" programming language the control flow syntax is as follows:

```
if (expression)
    statement1
else
    statement2
```

where the else part is optional. The expression is evaluated; if it is true (that is, if *expression* has a non-zero value), statement<sub>1</sub> is executed. If it is false (expression is zero) and if there is an else part, statement<sub>2</sub> is executed instead.<sup>1</sup>

### 3.5.2 Command syntax

In FlashRunner this same goal can be achieved using the syntax below in the script file:

```
#IF expression
#THEN statement1
#ELSE statement2
```

in which *expression* is **TRUE** when the command returns ">" character (it means that command has been executed successfully), or it is **FALSE** if the command returns an error (with correspondant error code).

Syntax can be used also for the opposite:

```
#IFN expression
#THEN statement1
#ELSE statement2
```

in which statement<sub>1</sub> is executed when *expression* is false.

---

<sup>1</sup> "The ANSI C Programming Language" 2nd ed., Brian W. Kernighan and Dennis M. Ritchie, Prentice Hall

### 3.5.3 Notes

- Please note that syntax above can be used only inside a script file and it's not recognized on command line console.
- Control flows can't be nested
- Only one *expression* can be evaluated and only one *statement* can be executed for each case
- If *expression* evaluation return false, error stack will be traced in the log file. Anyway, if all the subsequent commands will return ">", script will not return with a execution error, and log file will not end with END LOGGING:ERR.

### 3.5.4 Example

The following example is an extract from a script where the MASSERASE operation is carried out only if blank check operation returns an error, that is the device to be programmed is not blank.

```
#IFN TPCMD BLANKCHECK P $A0000000 $180000  
#THEN TPCMD MASSERASE P
```

With this approach it is possible to get a more effective of the script, reducing in some cases the processing script execution.

Please refer to your algorithm specific commands before implementing conditional scripting it in your script file.

## 4 Dynamic Memory and Serial Numbering

### 4.1 Introduction

Thanks to its built-in dynamic memory, FlashRunner provides you with the possibility of serial numbering during programming operations. During each programming cycle, a host system generates a serial number and transfers it to FlashRunner's dynamic memory. The content of the dynamic memory is then programmed into the target device.

### 4.2 Example

The following example illustrates how serial numbering can be performed.

Let's assume that the serial number is composed by 4 bytes, must be programmed into the target device starting from address \$400, and the next serial number to be programmed is \$55 \$AA \$22 \$FE.

The host system transfers this serial number to FlashRunner's dynamic memory with the following command:

```
DMSET 0 4 $55 $AA $22 $FE
```

It then sends the following commands to program (and verify) the serial number to the target device:

```
TPSTART  
TPSETSRC DYNAMIC  
TPCMD PROGRAM F $0 $400 4  
TPCMD VERIFY F $0 $400 4  
TPEND
```

Another technique is to include the following commands:

```
TPSETSRC DYNAMIC
TPCMD PROGRAM F $0 $400 4
TPCMD VERIFY F $0 $400 4
```

inside a script, within a **TPSTART/TPEND** block (and after the rest of the target device's memory has been programmed, so that serial number programming takes place only if the rest of the memory has been successfully programmed).

As before, the host system first transfers the next serial number to be programmed to FlashRunner through the **DMSET** command, and then simply runs the script through the **RUN** command.

For a detailed description of the above-mentioned commands, see the "FlashRunner Commands" chapter.



## 5 SD Card File Structure

### 5.1 Overview

The SD card is an integral part of FlashRunner. A properly configured SD card must always be present in order for FlashRunner to perform programming operations.

The SD card contains:

- Programming algorithms;
- Scripts;
- Data to be programmed to the target device;
- Licensing information;
- Log files.

The card has a fixed directory structure where the various files needed for proper FlashRunner operation are stored. FlashRunner uses a FAT16 or a proprietary file system, depending on the instrument model. FAT16 cards are directly readable/writable by a host system using any commercial SD card reader.

However, FlashRunner uses certain areas of the card as private swap areas in order to speed up programming operations.



**Note:** With OS version 1.4 (OS1.4) FlashRunner provides new features:

- Supports also FAT32 on SD cards
- User binary file (.frb FR files), script files (.frs files) and log files (.log) are supported with long name format of 42 characters
- Added specific commands for file management in SD card subdirectories

*Forbidden characters: following characters are not allowed for naming purposes of .frs, .frb and log files:*

`" * + , . / : ; < = > ? [ \ ] | å`

*Space character is not allowed too.*

*The FAT16 file system on OS1.3 does not allow use of long file names. Only 8.3 file names are allowed.*

*Additionally, files with the same name, even if in different directories, are not allowed.*

7



**Note:** Particular attention shall be taken to use SD cards, already formatted with FR OS1.4, on older FR OS (rev. 1.3 and older). In fact, SWAP file generated by OS1.4 is not managed from older OS.

*Using SD cards formatted with newer OS releases (from FROS1.4) on older OS releases (OS1.3 and previous one) **will destroy the content of the SD card.***

*SD card must be formatted again to work on older OS releases.*

## 5.2 Card Formatting

Due to the fact that FlashRunner uses some areas of the card as private swap areas, a card must be properly initialized before it can be used. To initialize a card, you must insert it into FlashRunner and send the **FSFORMAT** command. This command does the following:

- Erases all files on the card;
- Formats the card with a FAT16, FAT32 (OS 1.4) or a proprietary file system (depending on the FlashRunner model);
- Allocates private areas;
- Creates the appropriate directory structure needed for proper FlashRunner operation.



**Note:** *the only way to properly format a card is to use FlashRunner's **FSFORMAT** command. Cards formatted directly by a host system will not work in FlashRunner, since they will not have the reserved areas needed by FlashRunner.*



**Note:** *FlashRunner is only able to format blank or previously FAT16 or FAT 32 (OS 1.4) formatted cards. A card that is already FAT32 formatted cannot be reformatted by FlashRunner.*



**Note:** *after formatting a card, the **DEFAULT.ALG** file must be copied in the **\ALGOS** directory. This file can be found in the FlashRunner System Software CD.*

## **5.3 Using the Card on a PC**

If you move the card to a PC to read/write its contents, it is suggested that, before to remove it from the PC or card reader, you safely “unplug it” from your system. On Windows XP, follow the “Safely Remove Hardware” procedure.

Failing to do so may result in FlashRunner not being able to read the card correctly.

## 5.4 File Structure

<code>\ALGOS</code> <code>  DEFAULT.ALG</code> <code>  XXX.ALG</code>	<p>This directory contains programming algorithms. Each file (.ALG extension) includes programming algorithms for several devices. In order to program a specific device, however, a specific license file (see below) must be purchased. The <code>DEFAULT.ALG</code> file must always be present.</p>
<code>\BINARIES</code> <code>  AAA.FRB</code>	<p>This directory contains the Flash or EEPROM images used to program the target device. These files have a .FRB extensions and must be created through the FlashRunner Control Panel utility (see page 159).</p>
<code>\LICENSES</code>	<p>This directory contains all of the purchased licenses. Licenses are files with .LIC extension which enable FlashRunner to program specific devices or add new features. When you purchase a new license, you must copy it here. You could also create subfolders with SN as folder name: in this case FlashRunner will search for a valid license also inside folder name which match current FlashRunner serial number</p>
<code>\LOGS</code>	<p>This directory contains log files. Log files are text files created by FlashRunner through the <code>FSMKLOG</code> command. Log files have names and extension chosen by the user.</p>
<code>\SCRIPTS</code> <code>  SCRIPT0.FRS</code> <code>  ...</code>	<p>This directory contains script files. Script files for standalone execution must have a predefined name (SCRIPT0 to SCRIPT31). Script files for host-controlled execution may have any name. All scripts must have a .FRS extension.</p>

Figure 9: SD Card File Structure



## 6 Data Protection System

### 6.1 Overview

FlashRunner implements an optional data protection system that makes the contents of the binary file to be programmed to the target device not readable (and not duplicable) by non-authorized people. Additionally, the protection system extends to the programming cycle, preventing production personnel from tampering with the programming flow.

With the FlashRunner Data Protection System:

- Binary image files cannot be read/modified by unauthorized people;
- The production operator cannot alter the programming cycle as set up by the production engineer.

The Data Protection System works by encrypting the FlashRunner SD card with an administrator password; without the administrator password, FlashRunner usage is limited, allowing the production personnel to only execute a pre-set programming flow, without being able to read, copy or modify the binary code.

In a typical scenario, the production engineer sets up a FlashRunner SD card, by copying there the binary code to be programmed together with the desired programming flow. The SD card is then encrypted and handed to the production facility. The figure below illustrates this typical scenario.

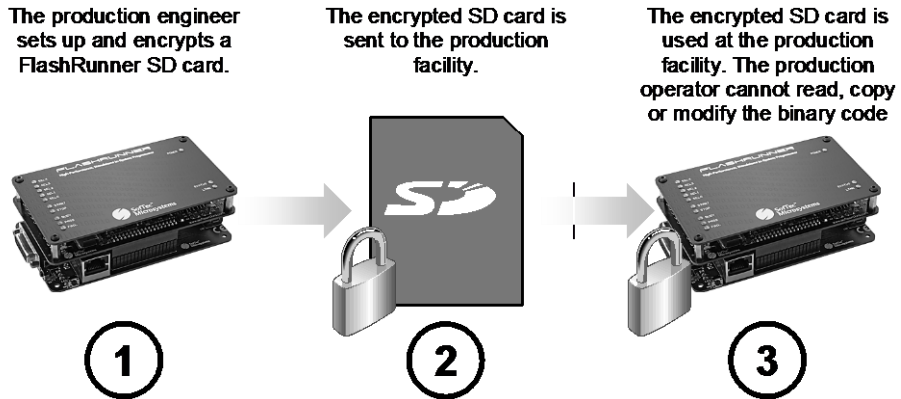


Figure 10: Typical Data Protection System Workflow

## 6.2 Installation

The Data Protection System is an optional feature of FlashRunner. When you purchase it, you will get a serial-number specific license file (.lic) that must be copied into the `\LICENSES` directory of the SD card.

To copy files on the SD card, use either a standard card reader connected to a PC or transfer the files using the FlashRunner `FSENDFILE` command.

Alternatively, you can use the FlashRunner Control Panel utility to install new licenses. For more information on the FlashRunner Control Panel, see “FlashRunner Control Panel” on page 159.

## 6.3 Data Protection System Operations

The following pages illustrate the concepts behind the Data Protection System, and the corresponding dedicated FlashRunner commands. For a detailed description of these commands please refer to “Command Reference” on page 39.



### 6.3.1 Modes, Users and Passwords

When the Data Protection System is installed, two modes are available:

- An “Unsecured” mode (the default state), with normal FlashRunner system behavior;
- A “Secured” mode, with special system behavior.

In Secured mode, two users are available:

- Operator (can only perform certain programming operations);
- Administrator (can perform every FlashRunner operation).

If FlashRunner is set to work Secure mode, it will begin as user Operator at every startup.

The **ASECURE** command switches from Unsecured to Secured mode (Administrator user); the **AUNSECURE** command switches back to Unsecured mode. The **ACHGUSER** command switches between an user and the other.

The first time the **ASECURE** command is executed, the Administrator password is set to **ADMIN** by default: the **ACHGPWD** command should be used to modify the default password.

The figure below illustrates the relations between Data Protection System modes and users, and the commands to change mode/user.

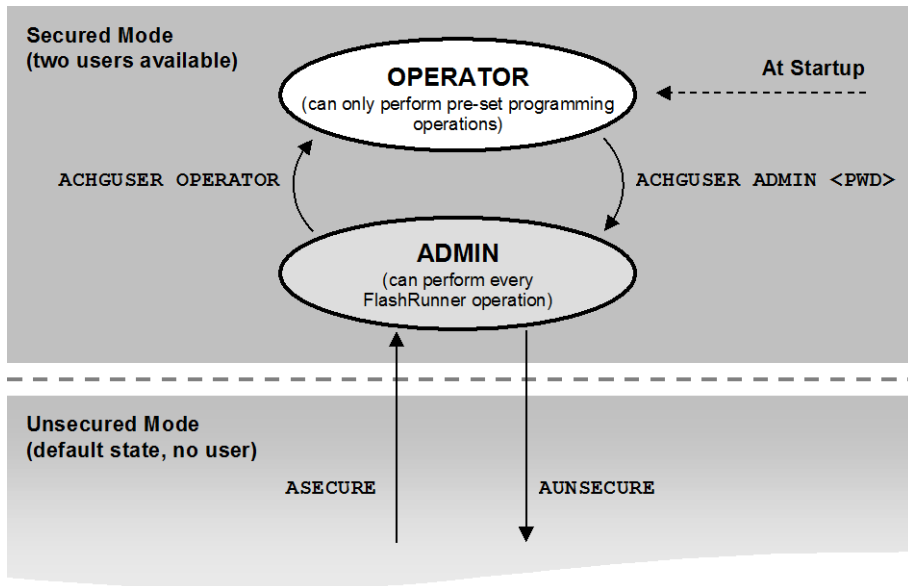


Figure 11: Data Protection System Modes and Users

The table below lists the FlashRunner commands available for each user.

Table 7: FlashRunner Commands Availability for Different Modes/Users

Command	Unsecured Mode	Secured Mode (Operator)	Secured Mode (Administrator)
ACHGPWD	-	-	●
ACHGUSER	-	□	●
ASECURE	●	-	-
AUNSECURE	-	-	●
DLOCK	-	-	●
DMGET	●	□	●
DMSET	●	□	●
DUNLOCK	-	-	●
FSCD	●	□	●
FSCLRLOG	●	-	●
FSFORMAT	●	-	●
FSGETFILE	●	□	●
FSGETTIME	●	□	●
FSLS	●	□	●
FSMD	●	-	●
FSMKLOG	●	□	●
FSRD	●	-	●
FSRF	●	-	●
FSENDFILE	●	□	●
FSSETTIME	●	-	●
LVDSET	●	□	●
MCRESET MC	●	□	●
MCSETENG	●	□	●
MCUPDATE	●	□	●
MCSPING	●	□	●
MCGETTIME	●	□	●
MCSETTIME	●	□	●
MCSETIP	●	□	●
MCSETNM	●	□	●
MCSETGW	●	□	●
MCGETIP	●	□	●
MCGETNM	●	□	●
MCGETGW	●	□	●
MCGETMAC	●	□	●
MCGETVER	●	□	●
MXCLOSE	●	□	●
MXGET	●	□	●
MXOPEN	●	□	●
OCDSET	●	□	●

**Errore. Nel documento non esiste testo dello stile specificato.**

---

Command	Unsecured Mode	Secured Mode (Operator)	Secured Mode (Administrator)
RLYGET	●	□	●
RLYSET	●	□	●
RUN	●	□	●
SGETCRC	●	□	●
SGETERR	●	□	●
SGETSTS	●	□	●
SGETVER	●	□	●
SPING	●	□	●
SSETBR	●	-	●
TCSETDEV	●	-	●
TCSETPAR	●	-	●
TPCMD	●	-	●
TPEND	●	-	●
TPSETDST	●	□	●
TPSETSRC	●	□	●
TPSTART	●	-	●
TST	●	□	●

Please note that the Operator user is prevented from performing both data editing commands and reading reserved information (such as a binary file to be programmed). Files transferred to/from the instrument by the Operator user via the **FSGETFILE** and **FSENDFILE** commands are encrypted (for more information, see the **FSGETFILE** and **FSENDFILE** command reference).

### 6.3.2 SD Card Locking/Unlocking

The Administrator user has the possibility of locking (encrypting) the SD card data, so that only a person with the Administrator password is able to view/modify the SD card contents.

The commands that lock and unlock the SD cards are **DLOCK** and **DUNLOCK**, respectively (see figure below).

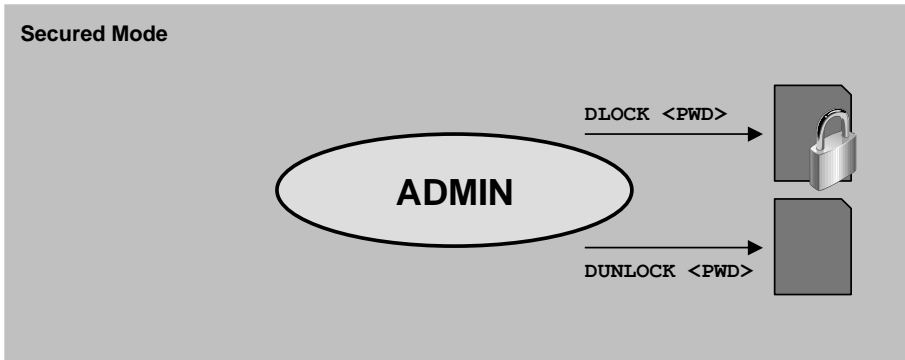


Figure 12: Data Protection System SD Card Lock



# 7 FlashRunner Control Panel

## 7.1 Overview

The FlashRunner Control Panel is a Windows utility that allows you to communicate with FlashRunner and perform the following operations:

- Send and receive FlashRunner commands;
- Create FlashRunner binary files from Motorola S19, Intel-Hex and binary formats and transfer them to the instrument;
- Transfer scripts to the instrument;
- Retrieve log files;
- Install new programming algorithms and licenses;
- Update the instrument firmware.



FlashRunner **FRPXIA3** Model use “FRPXIA3 Control Panel” Windows utilities. Refer directly to chapter 7.1.2.

### 7.1.1 Starting the Control Panel and Connecting to FlashRunner

To launch the FlashRunner Control Panel utility, select **Start > Programs > SMH Technologies > FlashRunner > Control Panel**. The Control Panel utility will open.

To establish a connection with FlashRunner, on the “**Communication Settings**” section, select:

- “**FlashRunner serial version**” (if you are connected to FlashRunner through a serial port), or

**Errore. Nel documento non esiste testo dello stile specificato.**

---

- **“FlashRunner LAN version”** (if you are connected to FlashRunner through an Ethernet port).

Next, specify:

- The COM port you are using and the baud rate (for the serial connection—by default, FlashRunner communicates at 115200 bps), or
- The instrument IP address (for the Ethernet connection). For learning how to set up the FlashRunner IP address, see “LAN Settings” on page 201.

Click the **“Connect”** button.



**Note:** the **“Port”** parameter lists COM1 to COM4 ports, but you can type in a different COM port.



**Note:** if connection to FlashRunner through an Ethernet port fails (but you are still able to ping the instrument, see *Testing Board Connection* on page 207), make sure that the port 2101 is not blocked by your Firewall.



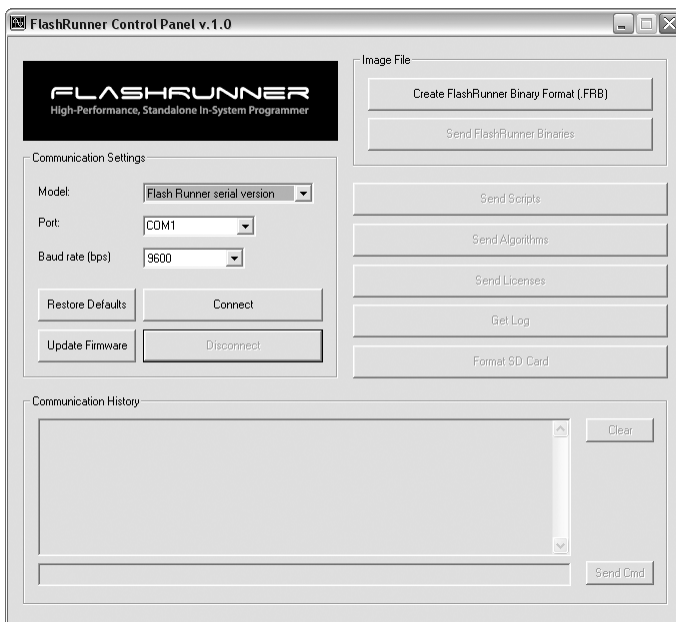


Figure 13: FlashRunner Control Panel, Communication Settings

### 7.1.2 Starting the FRPXIA3 Control Panel and Connecting to FlashRunner FRPXIA3

To launch the FRPXIA3 Control Panel utility, select **Start > Programs > SMH Technologies > FRPXIA3 > FRPXIA3 Control Panel**. The Control Panel utility will open.

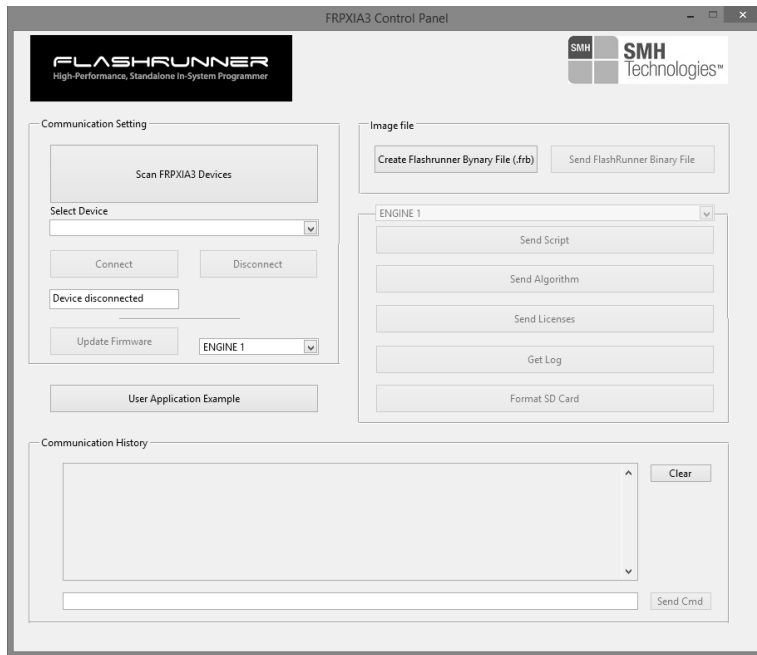


Figure 14: FlashRunner FRPXIA3 Control Panel

To establish a connection with FlashRunner, on the “**Communication Settings**” click the “**Scan FRPXIA3 devices**” button. Follow the message box instruction and select the device in the scroll-down menu, then click the “**Connect**” button.



**Note:** if you have more than one FRPXIA3 installed in your chassis, all devices will be showed on the “**Select Device**” scroll-down menu.

### 7.1.3 Sending and Receiving FlashRunner Commands

You can send commands to FlashRunner by typing them in the edit box below the communication history and then pressing the Return key (or clicking the **“Send Cmd”** button).

By pressing the arrow up and down keys, you can navigate through the most recently typed commands.

On the **“Communication History”** section you can see all the commands that have been sent and received.

### 7.1.4 Creating FlashRunner Binary Files

FlashRunner accepts only image files in a .FRB (FlashRunner Binary) format. To convert your binary, Intel-Hex or S19 image file to the FlashRunner format, click the **“Create FlashRunner Binary Format”** button. The following dialog box will appear.

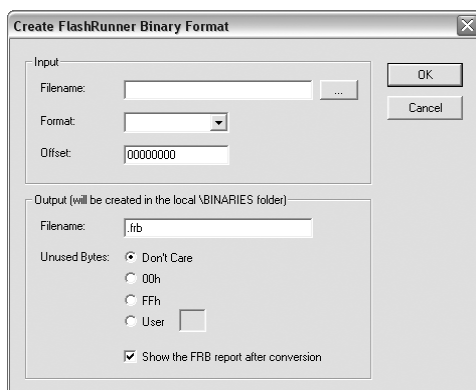


Figure 15: FlashRunner Control Panel, Binary File Conversion

In the **“Input”** section, specify the source file to be converted, its format, and the address from which the file conversion will start (offset). In the **“Output”** section, specify the output filename and the value used to fill unused locations. Please note that the **“Don't Care”** option fills unused locations with unpredictable values.

The diagram below illustrates how file conversion is performed.

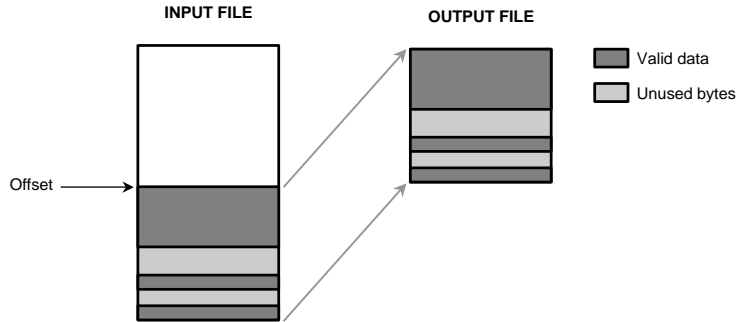


Figure 16: FlashRunner Control Panel, Binary File Conversion Diagram

Click the “**OK**” button. The FlashRunner Binary file will be created in the local `\BINARIES` folder.

To transfer the created image to FlashRunner, click the “**Send FlashRunner Binary**” button. The following dialog box will appear.

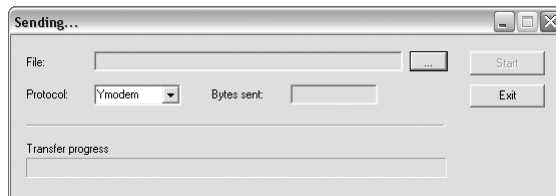


Figure 17: FlashRunner Control Panel, File Transfer

Click the “...” to browse for the image file to be sent, then click “**Start**” to begin the transfer. The file will be saved to the FlashRunner SD card, in the `\BINARIES` folder.

### 7.1.5 Transferring Scripts to the Instrument

To transfer script files to FlashRunner, click the “**Send Script**” button. A file transfer dialog box will appear.

Browse for the script file to be sent, then click **“Start”** to begin the transfer. The file will be saved to the FlashRunner SD card, in the `\SCRIPTS` folder.

### 7.1.6 Retrieving Log Files

To retrieve the current log from FlashRunner, click the **“Get Log”** button. A file transfer dialog box will appear.

Specify the filename with which you want the log to be saved, then click **“Start”** to begin the transfer. By default, the log file will be saved to the local `\LOGS` folder.



***Note:** the **“Get Log”** function of the FlashRunner Control Panel automatically creates a temporary log file, called `TMP_CP.LOG`, in the FlashRunner SD card, under the `\LOGS` directory. It is this file that is then transferred to the host.*

### 7.1.7 Installing Programming Algorithms and Licenses

New programming algorithms and new programming licenses can be installed by clicking the **“Send Algorithm”** and **“Send License”** buttons, respectively. A file transfer dialog box will appear.

Browse for the programming algorithm or license file to be sent, then click **“Start”** to begin the transfer. The algorithm or license file will be saved to the FlashRunner SD card, in the `\ALGOS` or `\LICENSES` folder, respectively.



***Note:** the FRPXIA3 Control Panel utility begins the file transfer when you select the file on the file dialog box.*

### 7.1.8 Updating the Firmware

To update the firmware, make sure the instrument is disconnected from the target device. Then, make sure you have selected the correct FlashRunner model and COM port/baud rate or IP address settings, then click the **“Update Firmware”** button (this button is available only when no connection is established with the instrument).

A dialog box will appear, instructing you to turn off FlashRunner and turn it back on while the **“START”** signal in the **“CONTROL”** connector is being kept active (on FlashRunner models featuring a **“START”** push-button, simply keep the **“START”** button pressed while powering on the instrument). After doing so, the **“STATUS”** LED will blink, indicating that FlashRunner is ready to accept a firmware update. Click **“OK”** to continue.

A file transfer dialog box will appear. Browse for the firmware file to be sent to the instrument, then click **“Start”** to begin the transfer. After the firmware transfer has been completed, FlashRunner will automatically reboot and will be ready for operation.



***Note:** the **FRPXIA3** Control Panel utility allow to update the engines firmware simply selecting the engine on the scroll-down menu and clicking the **“Update Firmware”** button.*

## 7.2 Command-Line Utilities

In addition to the FlashRunner Control Panel, some Windows command-line utilities are available, in the local **“Utilities\Command Line”** folder:

- **fr\_transfer**: to transfer a file to/from FlashRunner;
- **fr\_bin2frb**: to convert a binary, S19 or Intel-Hex file to a FlashRunner binary file;
- **fr\_command**: to send a FlashRunner command over a RS-232 or Ethernet connection;
- **fr\_chk**: to calculate the CRC of a file;
- **fr\_filesec** to encrypt/decrypt a file.

Type the command name followed by `?` to show help about command usage (e.g. `fr_transfer ?`).

These command-line utilities are useful when you need to perform a file conversion or a file transfer and you can't use the FlashRunner Control Panel—for example, when integrating FlashRunner with a programming language such as National Instruments LabView.

Alternatively, these command-line utilities can be used within a production system to automate file conversion and/or file transfer by creating batch files.





# 8 FlashRunner Interface Library

## 8.1 Overview

This chapter deals with interfacing user written PC applications with FlashRunner. This chapter assumes you have already read the previous sections of this manual and got acquainted with the instrument.

## 8.2 The FlashRunner Interface Library

The FlashRunner Interface Library is a DLL which includes all of the functions that allow you set up a communication channel with the instrument and send commands to FlashRunner.

Dynamic-link libraries (DLL) are modules that contain functions and data. A DLL is loaded at run time by its calling modules (.exe or .dll). When a DLL is loaded, it is mapped into the address space of the calling process.

The FlashRunner Interface Library contains Visual C++ written routines, and can be used to interface the instrument from within, for example, a Microsoft Visual C++ or Visual Basic application, as well as any other programming language that supports the DLL mechanism. For details on how to call DLL functions from within your application, please refer to the your programming language's documentation.

## 8.3 Installation

Before to start working with the FlashRunner Interface Library, you must set up your system with all the required files and drivers. The files to be installed, into your application's directory, are:

- The “**FR\_COMM.dll**” (this file must also be redistributed with your application);

- For Visual C++ only: the “**FR\_COMM.lib**” and “**FR\_COMM.h**” files (you must include these files in your project);
- For Visual Basic only: the “**FR\_COMM.bas**” file (you must include this file in your project).

These files are automatically installed by the System Software setup (in your local “**Utilities\Interface Library\Bin**” directory, relative to the installation path), as described in “Software Setup” on page 29.

## 8.4 Interface Library Reference

### 8.4.1 Using the Interface Library Functions

When you control FlashRunner within your own application, you will typically follow the steps indicated below:

1. **Open a communication channel with the instrument.**  
The `FR_OpenCommunication()` function must be called prior to any other Interface Library function.
2. **Send commands to the instrument and read answers back.**  
Use the `FR_SendCommand()` and `FR_GetAnswer()` functions to send a command and receive the answer sent back by the instrument, respectively.
3. **Transfer files to/from the SD Card.**  
Two dedicated functions, `FR_SendFile()` and `FR_GetFile()`, allow you to transfer a file from the PC to the instrument’s SD card and vice-versa, respectively.  
The `FR_SendFile()` function is typically used to upload a binary file to the instrument, while the `FR_GetFile()` function is typically used to download a log file to the PC.
4. **Close the communication channel with the instrument.**  
This is done by the `FR_CloseCommunication()` function.

## 8.4.2 Return Values of the Interface Library Functions

Most of the FlashRunner Interface Library functions return an `unsigned long` value which indicates whether the function was successfully executed (return value = 0) or not (return value other than 0). In the latter case it is possible to get extended error information by calling the function `FR_GetLastErrorMessage()` function.

## 8.4.3 Unicode Functions

Every Interface Library function comes in two versions, an ASCII version and an Unicode version. ASCII function names end with A, while Unicode function names end with W. For example, the `FR_SendCommand()` function is available as an ASCII version as:

```
FR_COMM_ERR WINAPI FR_SendCommandA (FR_COMM_HANDLE  
handle, const char *command);
```

and as an Unicode version as:

```
FR_COMM_ERR WINAPI FR_SendCommandW (FR_COMM_HANDLE  
handle, const wchar_t *command);
```

## 8.4.4 Application Examples

Application examples for Visual C and Visual Basic are provided in the local “\Utilities\Interface Library\Examples” directory, relative to the installation path.

## 8.5 Function Reference for FlashRunner FR01AT0, FR01ENG, FR01LAN, FR01M01, FR01PRO, FR03, FR04

Each Interface Library function is listed alphabetically and explained in the following pages.

### 8.5.1 FR\_CloseCommunication

**Include file:**

```
#include "FR_COMM.h"
```

**Function prototypes:**

```
FR_COMM_ERR WINAPI FR_CloseCommunicationA  
    (FR_COMM_HANDLE handle);  
FR_COMM_ERR WINAPI FR_CloseCommunicationW  
    (FR_COMM_HANDLE handle);
```

**Parameters:**

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**Return value:**

0: the function was successful.  
Other than 0: an error occurred. Call the `FR_GetLastErrorMessage()` function to get extended error information.

**Description:**

Closes the communication link with the instrument.

## 8.5.2 FR\_FirmwareUpgrade

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_ERR WINAPI FR_FirmwareUpgradeA
    (FR_COMM_HANDLE handle,
     const char *filename,
     FR_FileTransferProgressProc
     progress);

FR_COMM_ERR WINAPI FR_FirmwareUpgradeW
    (FR_COMM_HANDLE handle,
     const wchar_t *filename,
     FR_FileTransferProgressProc
     progress);
```

### Parameters:

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**filename:** name of the firmware file.

**progress:** address of a callback function which will receive the progress status of the upgrade operation. If not used, set this parameter to 0.

### Return value:

0: the function was successful.

Other than 0: an error occurred. Call the `FR_GetLastErrorMessage()` function to get extended error information.

### Description:

Updates the instrument firmware. Before calling this function, turn off FlashRunner and turn it back on while the "START" signal in the

“CONTROL” connector is being kept active (on FlashRunner models featuring a “START” push-button, simply keep the “START” button pressed while powering on the instrument). After doing so, the “STATUS” LED will blink, signalling that the instrument is ready to receive a firmware update. The `FR_FirmwareUpgrade()` function must be called at this point. After the firmware transfer has been completed, FlashRunner will automatically reboot and will be ready for operation.

### 8.5.3 FR\_GetAnswer

**Include file:**

```
#include "FR_COMM.h"
```

**Function prototypes:**

```
FR_COMM_ERR WINAPI FR_GetAnswerA
    (FR_COMM_HANDLE handle,
     char *answer,
     unsigned long maxlen,
     unsigned long timeout_ms);

FR_COMM_ERR WINAPI FR_GetAnswerW
    (FR_COMM_HANDLE handle,
     wchar_t *answer,
     unsigned long maxlen,
     unsigned long timeout_ms);
```

**Parameters:**

<b>handle:</b>	handle of the communication. This is the value returned by the <code>FR_OpenCommunication()</code> function.
<b>answer:</b>	buffer that will receive the answer (\0 terminated) of the instrument.
<b>maxlen:</b>	maximum number of characters to receive (must be less than or equal to the <code>answer</code> buffer length).
<b>timeout_ms:</b>	timeout, in milliseconds, after which the function returns even if a complete answer has not been received.

**Return value:**

0:	the function was successful.
Other than 0:	an error occurred. Call the <code>FR_GetLastErrorMessage()</code> function to get extended error information.

### Description:

Receives the answer sent by FlashRunner to the PC, in response to the `FR_SendCommand()` function. A `FR_GetAnswer()` function should always follow a `FR_SendCommand()` function.



## 8.5.4 FR\_GetFile

**Include file:**

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_ERR WINAPI FR_GetFileA
    (FR_COMM_HANDLE handle,
     const char *protocol,
     const char *src_filename,
     const char *dst_path,
     FR_FileTransferProgressProc
     progress);

FR_COMM_ERR WINAPI FR_GetFileW
    (FR_COMM_HANDLE handle,
     const wchar_t *protocol,
     const wchar_t *src_filename,
     const wchar_t *dst_path,
     FR_FileTransferProgressProc
     progress);
```

### Parameters:

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**protocol:** transfer protocol. Must be `"YMODEM"`.

**src\_filename:** name of the file (inclusive of the path) to be retrieved from FlashRunner, e.g. `"\\LOGS\\LOG.TXT"`.

**dst\_path:** local path where to save the file.

**progress:** address of a callback function which will receive the progress status of the file transfer operation. If not used, set this parameter to `NULL`.

### Return value:

0: the function was successful.

Other than 0: an error occurred. Call the `FR_GetLastErrorMessage()` function to get extended error information.

### Description:

Retrieves a file from the SD card of FlashRunner and stores it in a specified local path.



### 8.5.5 FR\_GetLastErrorMessage

**Include file:**

```
#include "FR_COMM.h"
```

**Function prototypes:**

```
void WINAPI FR_GetLastErrorMessageA  
    (char *error_msg,  
     unsigned long string_len);  
  
void WINAPI FR_GetLastErrorMessageW  
    (wchar_t *error_msg,  
     unsigned long string_len);
```

**Parameters:**

**error\_msg:** buffer that will receive the error message.  
**string\_len:** length of the buffer.

**Return value:**

none.

**Description:**

Most of the FlashRunner Interface Library functions return an **unsigned long** value which indicates whether the function was successfully executed (return value = 0) or not (return value other than 0). In the latter case it is possible to get extended error information by calling the function **FR\_GetLastErrorMessage()** function.

## 8.5.6 FR\_OpenCommunication

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_HANDLE WINAPI FR_OpenCommunicationA  
    (const char *port,  
     const char *settings);  
  
FR_COMM_HANDLE WINAPI FR_OpenCommunicationW  
    (const wchar_t *port,  
     const wchar_t *settings);
```

### Parameters:

**port:** communication port. Must be "COM1", "COM2", etc. for serial communication or "LAN" for Ethernet communication.

**settings:** baud rate for serial communication (e.g. "115200"), or IP address and port for Ethernet communication (e.g. "192.168.1.100:2101").

### Return value:

**>0:** handle of the communication.

**NULL:** an error occurred. Call the **FR\_GetLastErrorMessage()** function to get extended error information.

### Description:

Creates a communication link with the instrument. Returns a communication handle that must be used by successive FlashRunner Interface Library function calls.

### 8.5.7 FR\_SendCommand

#### Include file:

```
#include "FR_COMM.h"
```

#### Function prototypes:

```
FR_COMM_ERR WINAPI FR_SendCommandA  
    (FR_COMM_HANDLE handle,  
     const char *command);  
  
FR_COMM_ERR WINAPI FR_SendCommandW  
    (FR_COMM_HANDLE handle,  
     const wchar_t *command);
```

#### Parameters:

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**command:** string containing the FlashRunner command.

#### Return value:

0: the function was successful.

Other than 0: an error occurred. Call the `FR_GetLastErrorMessage()` function to get extended error information.

#### Description:

Sends a command to FlashRunner. To get the command answer, use the `FR_GetAnswer()` function.

## 8.5.8 FR\_SendFile

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_ERR WINAPI FR_SendFileA
    (FR_COMM_HANDLE handle,
     const char *protocol,
     const char *src_filename,
     const char *dst_path,
     FR_FileTransferProgressProc
     progress);

FR_COMM_ERR WINAPI FR_SendFileW
    (FR_COMM_HANDLE handle,
     const wchar_t *protocol,
     const wchar_t *src_filename,
     const wchar_t *dst_path,
     FR_FileTransferProgressProc
     progress);
```

### Parameters:

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**protocol:** transfer protocol. Must be `"YMODEM"`.

**src\_filename:** name of the file (inclusive of the path) to be sent to FlashRunner, e.g. `"C:\\MYBINARIES\\FLASH1.FRB"`.

**dst\_path:** SD card path where to save the file.

**progress:** address of a callback function which will receive the progress status of the file transfer operation. If not used, set this parameter to `NULL`.

### Return value:

0: the function was successful.

## FlashRunner Interface Library

---

Other than 0: an error occurred. Call the **FR\_GetLastErrorMessage()** function to get extended error information.

### **Description:**

Sends a file from the PC to a specified path of the SD card of FlashRunner.



## **8.6 Function Reference for FlashRunner FRPXIA3**

Each Interface Library function is listed alphabetically and explained in the following pages.

### 8.6.1 FR\_CloseCommunication

**Include file:**

```
#include "FR_COMM.h"
```

**Function prototypes:**

```
FR_COMM_ERR WINAPI FR_CloseCommunicationA  
    (FR_COMM_HANDLE handle);  
FR_COMM_ERR WINAPI FR_CloseCommunicationW  
    (FR_COMM_HANDLE handle);
```

**Parameters:**

**handle:** handle of the communication. This is the value returned by the **FR\_OpenCommunication()** function.

**Return value:**

0: the function was successful.  
Other than 0: an error occurred. Call the **FR\_GetLastErrorMessage()** function to get extended error information.

**Description:**

Closes the communication link with the instrument.

## 8.6.2 FR\_FirmwareUpgrade

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_ERR WINAPI FR_FirmwareUpgradeA  
    (FR_COMM_HANDLE handle,  
     const char *filename, unsigned  
     int eng_num,  
     FR_FileTransferProgressProc  
     progress);  
  
FR_COMM_ERR WINAPI FR_FirmwareUpgradeW  
    (FR_COMM_HANDLE handle,  
     const wchar_t *filename, unsigned  
     int eng_num,  
     FR_FileTransferProgressProc  
     progress);
```

### Parameters:

<b>handle:</b>	handle of the communication. This is the value returned by the <code>FR_OpenCommunication()</code> function.
<b>filename:</b>	name of the firmware file.
<b>eng_num:</b>	engine number which will be updated
<b>progress:</b>	address of a callback function which will receive the progress status of the upgrade operation. If not used, set this parameter to 0.

### Return value:

0:	the function was successful.
Other than 0:	an error occurred. Call the <code>FR_GetLastErrorMessage()</code> function to get extended error information.

**Description:**

Updates the selected engine firmware. FlashRunner FRPXIA3 will automatically reboot and will be ready for operation.

### 8.6.3 FR\_GetAnswer

**Include file:**

```
#include "FR_COMM.h"
```

**Function prototypes:**

```
FR_COMM_ERR WINAPI FR_GetAnswerA
    (FR_COMM_HANDLE handle,
     char *answer,
     unsigned long maxlen,
     unsigned long timeout_ms);

FR_COMM_ERR WINAPI FR_GetAnswerW
    (FR_COMM_HANDLE handle,
     wchar_t *answer,
     unsigned long maxlen,
     unsigned long timeout_ms);
```

**Parameters:**

<b>handle:</b>	handle of the communication. This is the value returned by the <code>FR_OpenCommunication()</code> function.
<b>answer:</b>	buffer that will receive the answer (\0 terminated) of the instrument.
<b>maxlen:</b>	maximum number of characters to receive (must be less than or equal to the <code>answer</code> buffer length).
<b>timeout_ms:</b>	timeout, in milliseconds, after which the function returns even if a complete answer has not been received.

**Return value:**

0:	the function was successful.
Other than 0:	an error occurred. Call the <code>FR_GetLastErrorMessage()</code> function to get extended error information.

### Description:

Receives the answer sent by FlashRunner to the PC, in response to the `FR_SendCommand()` function. A `FR_GetAnswer()` function should always follow a `FR_SendCommand()` function.

## 8.6.4 FR\_GetFile

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_ERR WINAPI FR_GetFileA
    (FR_COMM_HANDLE handle,
     const char *protocol,
     const char *src_filename,
     const char *dst_path,
     FR_FileTransferProgressProc
     progress);

FR_COMM_ERR WINAPI FR_GetFileW
    (FR_COMM_HANDLE handle,
     const wchar_t *protocol,
     const wchar_t *src_filename,
     const wchar_t *dst_path,
     FR_FileTransferProgressProc
     progress);
```

### Parameters:

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**protocol:** transfer protocol. Must be `"YMODEM"`.

**src\_filename:** name of the file (inclusive of the path) to be retrieved from FlashRunner, e.g. `"\\LOGS\\LOG.TXT"`.

**dst\_path:** local path where to save the file.

**progress:** address of a callback function which will receive the progress status of the file transfer operation. If not used, set this parameter to `NULL`.

### Return value:

0: the function was successful.

## FlashRunner Interface Library

---

Other than 0: an error occurred. Call the **FR\_GetLastErrorMessage()** function to get extended error information.

### **Description:**

Retrieves a file from the SD card of FlashRunner and stores it in a specified local path.



## 8.6.5 FR\_GetLastErrorMessage

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
void WINAPI FR_GetLastErrorMessageA  
    (char *error_msg,  
     unsigned long string_len);  
void WINAPI FR_GetLastErrorMessageW  
    (wchar_t *error_msg,  
     unsigned long string_len);
```

### Parameters:

**error\_msg:** buffer that will receive the error message.  
**string\_len:** length of the buffer.

### Return value:

none.

### Description:

Most of the FlashRunner Interface Library functions return an **unsigned long** value which indicates whether the function was successfully executed (return value = 0) or not (return value other than 0). In the latter case it is possible to get extended error information by calling the function **FR\_GetLastErrorMessage()** function.

### 8.6.6 FR\_OpenCommunication

**Include file:**

```
#include "FR_COMM.h"
```

**Function prototypes:**

```
FR_COMM_HANDLE WINAPI FR_OpenCommunicationA  
    (FR_PCI_SLOT pSlot);  
FR_COMM_HANDLE WINAPI FR_OpenCommunicationW  
    (FR_PCI_SLOT pSlot);
```

**Parameters:**

**pSlot:** slot identification of the FRPXIA3 device.

**Return value:**

**>0:** handle of the communication.

**NULL:** an error occurred. Call the **FR\_GetLastErrorMessage()** function to get extended error information.

**Description:**

Creates a communication link with the instrument. pSlot contains the slot and bus position of the FRPXIA3 device. This information are provided by ScanBus function.

## 8.6.7 FR\_ScanBus

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_ERR WINAPI FR_ScanBusA(FR_PCI_SLOT *scan) ;  
FR_COMM_ERR WINAPI FR_ScanBusW(FR_PCI_SLOT *scan) ;
```

### Parameters:

**scan:** pointer to array table which will be filled with all FRPXIA3 devices present on the bus.

### Return value:

**>0:** handle of the communication.  
**NULL:** an error occurred. Call the **FR\_GetLastErrorMessage()** function to get extended error information.

### Description:

Scans the bus searching for FRPXIA3 devices. scan is a pointer to an array which will be filled with all the devices found.

### 8.6.8 FR\_SendCommand

#### Include file:

```
#include "FR_COMM.h"
```

#### Function prototypes:

```
FR_COMM_ERR WINAPI FR_SendCommandA  
    (FR_COMM_HANDLE handle,  
     const char *command);  
  
FR_COMM_ERR WINAPI FR_SendCommandW  
    (FR_COMM_HANDLE handle,  
     const wchar_t *command);
```

#### Parameters:

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**command:** string containing the FlashRunner command.

#### Return value:

0: the function was successful.

Other than 0: an error occurred. Call the `FR_GetLastErrorMessage()` function to get extended error information.

#### Description:

Sends a command to FlashRunner. To get the command answer, use the `FR_GetAnswer()` function.

## 8.6.9 FR\_SendFile

### Include file:

```
#include "FR_COMM.h"
```

### Function prototypes:

```
FR_COMM_ERR WINAPI FR_SendFileA
    (FR_COMM_HANDLE handle,
     const char *protocol,
     const char *src_filename,
     const char *dst_path,
     FR_FileTransferProgressProc
     progress);

FR_COMM_ERR WINAPI FR_SendFileW
    (FR_COMM_HANDLE handle,
     const wchar_t *protocol,
     const wchar_t *src_filename,
     const wchar_t *dst_path,
     FR_FileTransferProgressProc
     progress);
```

### Parameters:

**handle:** handle of the communication. This is the value returned by the `FR_OpenCommunication()` function.

**protocol:** transfer protocol. Must be `"YMODEM"`.

**src\_filename:** name of the file (inclusive of the path) to be sent to FlashRunner, e.g. `"C:\\MYBINARIES\\FLASH1.FRB"`.

**dst\_path:** SD card path where to save the file.

**progress:** address of a callback function which will receive the progress status of the file transfer operation. If not used, set this parameter to `NULL`.

### Return value:

0: the function was successful.

## FlashRunner Interface Library

---

Other than 0: an error occurred. Call the **FR\_GetLastErrorMessage()** function to get extended error information.

### **Description:**

Sends a file from the PC to a specified path of the SD card of FlashRunner.

## 8.7 FlashRunner FRPXIA3 LabVIEW Interface Library

The LabVIEW interface Library is a set of VIs (Virtual Instruments) that includes the same library functions listed before. The library uses the “Call Library Function” VI node to include the DLL.

You can find the Flashrunner FRPXIA3 LabVIEW Interface Library in the installation package in **”Installation Directory” > FRPXIA3 > Utilities > Interface library > Examples > FR\_COMM\_lib.zip**



**Note:** The Call Library Function VI node points to the **frpxia3dll\_libapi.dll** compatible with 32bit machine. In case of 64bit machine, the Call Library Function VI node must point to the **frpxia3dll\_libapi\_64.dll** present in **FR\_COMM\_lib\data** folder





## 9 LAN Settings

### 9.1 Overview

A LAN enabled FlashRunner model needs to be properly configured if you want to use it through the Ethernet connection. In particular, its IP address may need to be changed in order to match your local area settings.



**Note:** *FlashRunner is factory programmed with the IP address 192.168.1.100, port 2101.*

Additionally, if more than one FlashRunner is connected to the same local area network, each instrument needs to have a unique IP address.

This chapter will guide you through the steps required to configure LAN settings on both FlashRunner and your PC. In particular, you will learn how to:

1. Set up your PC local area network settings to connect to FlashRunner for the first time;
2. Connect FlashRunner to the PC and test the Ethernet connection;
3. Change FlashRunner's IP address.

### 9.2 Setting Up PC Connection

The following steps are based on a Windows XP system.

1. Open the **Windows Control Panel** and open the “**Network Connections**” item.

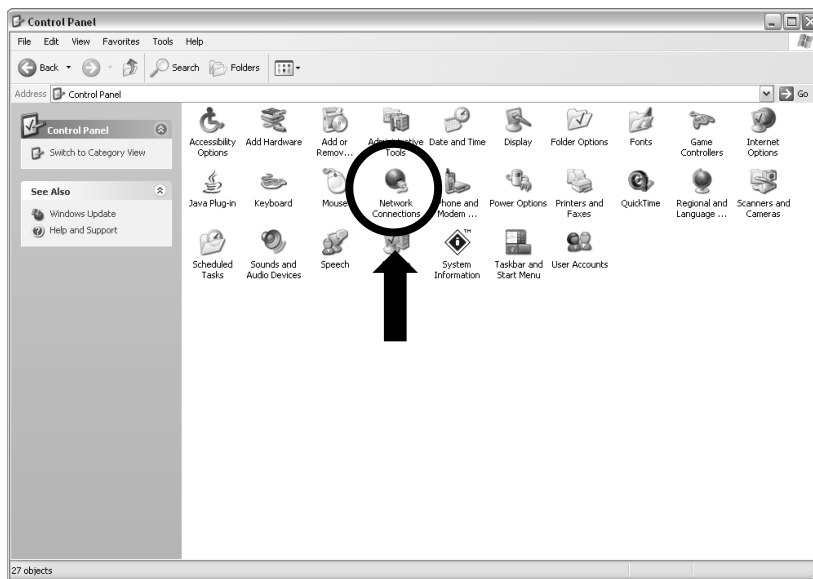


Figure 18: Windows Control Panel

2. If a **Local Area Connection** is already available, go to step 8. Otherwise, run the “**New Connection**” wizard.

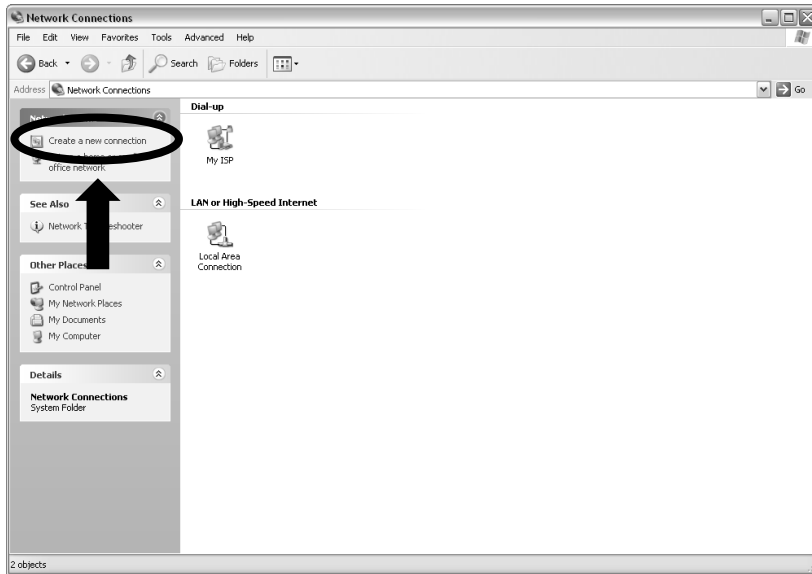


Figure 19: Network Connections

3. The following dialog box will appear. Click Next.



Figure 20: New Connection Wizard, Step 1

4. On the next dialog box, select the **“Connect to the Internet”** option and click the **“Next”** button.

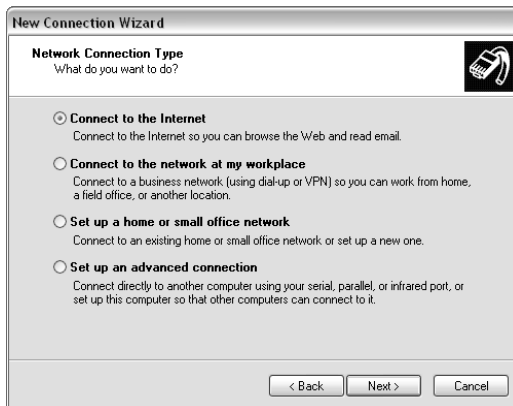


Figure 21: New Connection Wizard, Step 2

5. On the next dialog box, select the **“Set up my connection manually”** option and click the **“Next”** button.



Figure 22: New Connection Wizard, Step 3

- On the next dialog box, select the **“Connect using a broadband connection that is always on”** option and click the **“Next”** button.

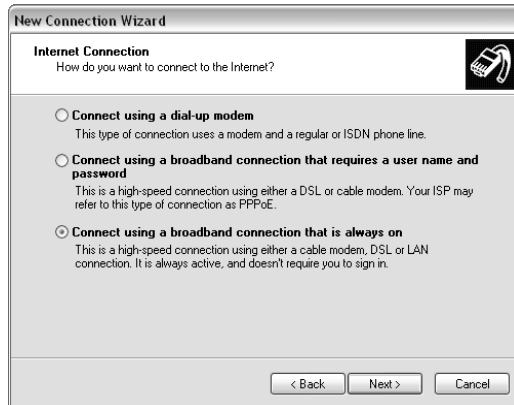


Figure 23: New Connection Wizard, Step 4

- When the last dialog box appears, click the **“Finish”** button.



Figure 24: New Connection Wizard, Step 5

## LAN Settings

---

- Open the newly created connection (or the existing connection if you are coming from step 2) and click the **“Properties”** button.

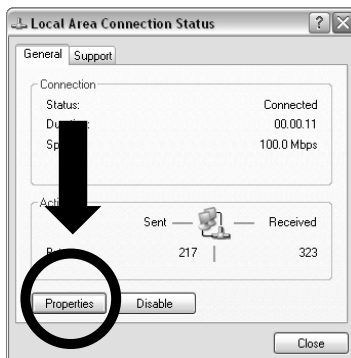


Figure 25: Local Area Connection Status

- Select the **“Internet Protocol (TCP/IP)”** item and click the **“Properties”** button.

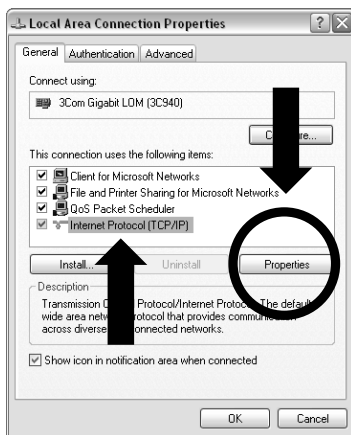


Figure 26: Local Area Connection Properties

10. If you have started from an existing connection, write down all of the TCP/IP parameters. You'll need them later to restore your LAN settings to the original parameters.
11. Select the manual settings and type in the following values:

IP address: **192.168.1.101**

Subnet mask: **255.255.255.0**

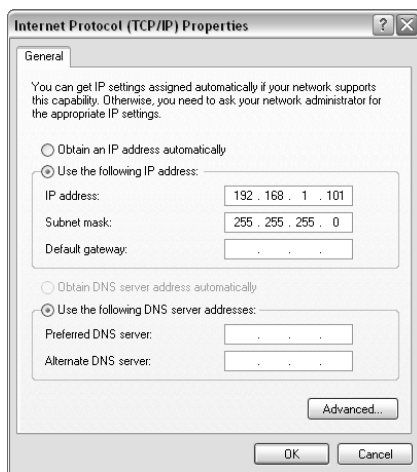


Figure 27: TCP/IP Properties

Click the “OK” button on all LAN setup dialog boxes.

9

### 9.3 Testing Board Connection

To verify that your local area network has been properly configured and that your PC can communicate with FlashRunner, follow the steps below.

1. Power up FlashRunner and verify that the “POWER” led turns on.
2. Connect FlashRunner to the PC through the provided Ethernet cross cable.



**Note:** the Ethernet cross cable provided can only be used to connect FlashRunner to a PC; if you want to connect FlashRunner to a hub, you must use an Ethernet patch cable (not provided).

3. Open a Command Prompt window (on Windows XP, from the Windows Start menu, select **Run**, enter **CMD** and click the “**OK**” button).
4. At the command prompt, type:

```
PING 192.168.1.100
```

If the connection has been properly setup, the PING command will return a positive feedback.

```
C:\WINDOWS\System32\CMD.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\>PING 192.168.1.100
Pinging 192.168.1.100 with 32 bytes of data:
Reply from 192.168.1.100: bytes=32 time<1ms TTL=100
Reply from 192.168.1.100: bytes=32 time<1ms TTL=100
Reply from 192.168.1.100: bytes=32 time<1ms TTL=100
Reply from 192.168.1.100: bytes=32 time<1ms TTL=100
Ping statistics for 192.168.1.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>_
```

Figure 28: Pinging FlashRunner

## 9.4 Changing FlashRunner’s IP Address

To change FlashRunner’s IP address, you need to connect to FlashRunner’s built-in web server.

1. Launch your favorite web browser.



2. You must disable the proxy server, if one is enabled. To disable this feature (in Internet Explorer), open the **Internet Options** dialog box, go to the **Connections** tab, and click the **“LAN Settings”** button.

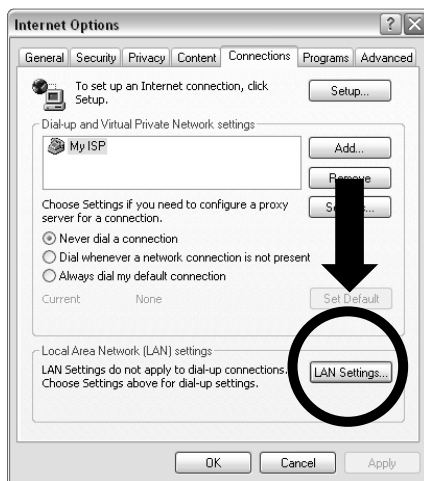


Figure 29: Disabling the Proxy Server, Step 1

Then, if the connection has a proxy enabled, make sure that the **“Use proxy server for your LAN”** option is disabled.

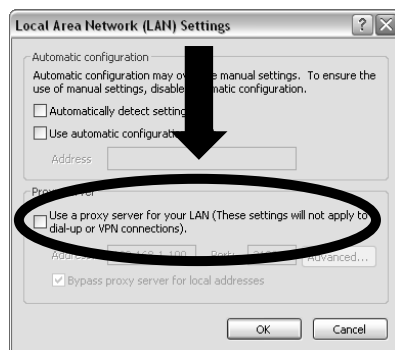


Figure 30: Disabling the Proxy Server, Step 2

## LAN Settings

3. Connect to the following address: **192.168.1.100** (this is FlashRunner's factory programmed IP address). The FlashRunner configuration page will appear.

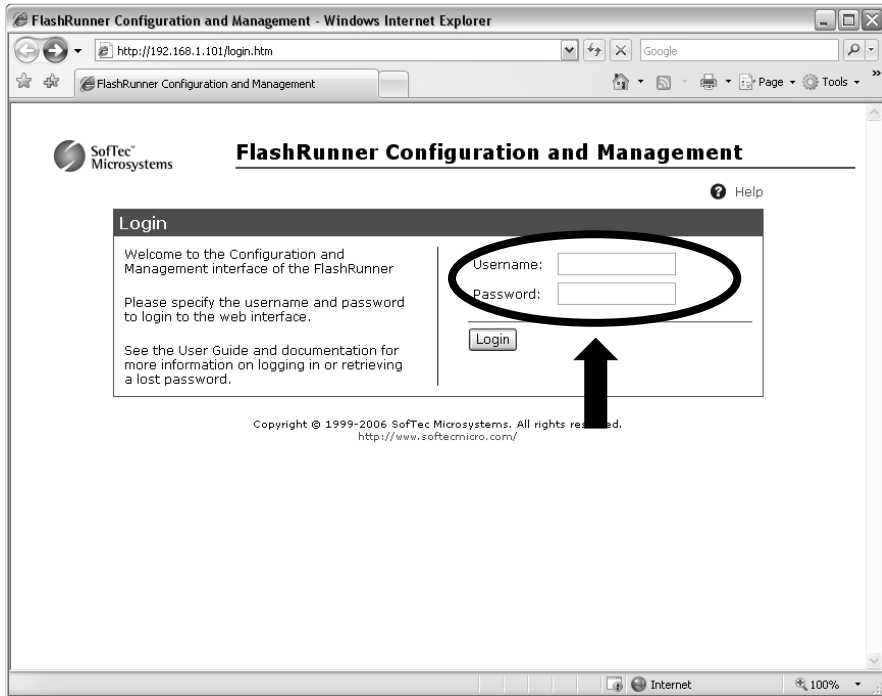


Figure 31: LAN Enabled FlashRunner Models, Configuration Page

Type in the username and password to login:

Username: **customer**

Password: **customer**

4. Click the **“Login”** button. The following page will appear.

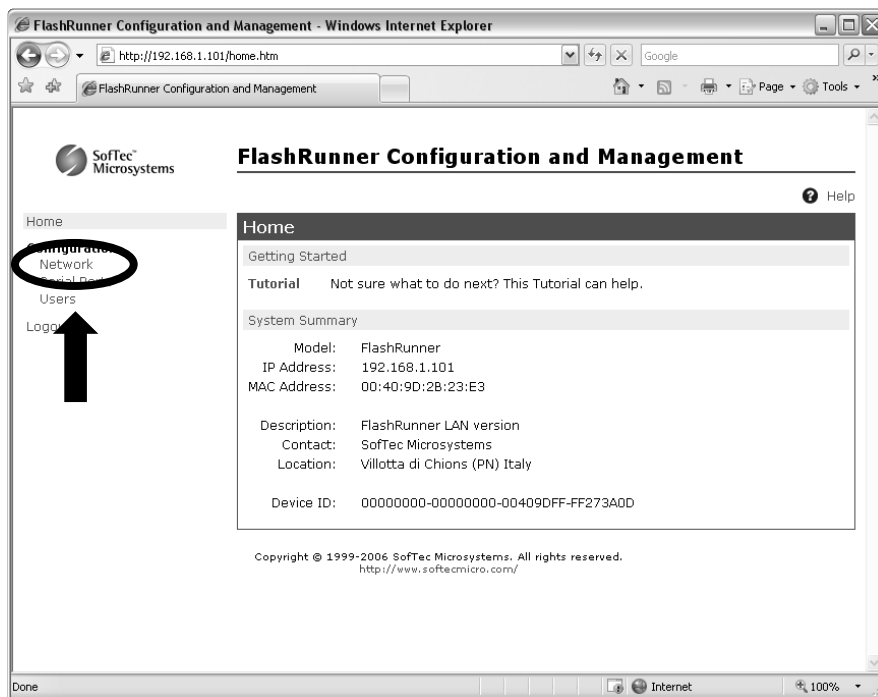


Figure 32: LAN Enabled FlashRunner Models, Network Summary

Click the **“Network”** link to access the network configuration page.

## LAN Settings

5. Change the “**IP Address**”, “**Subnet Mask**” and “**Default Gateway**” parameters as needed, or choose to use DHCP to automatically obtain an IP address from your DHCP server.

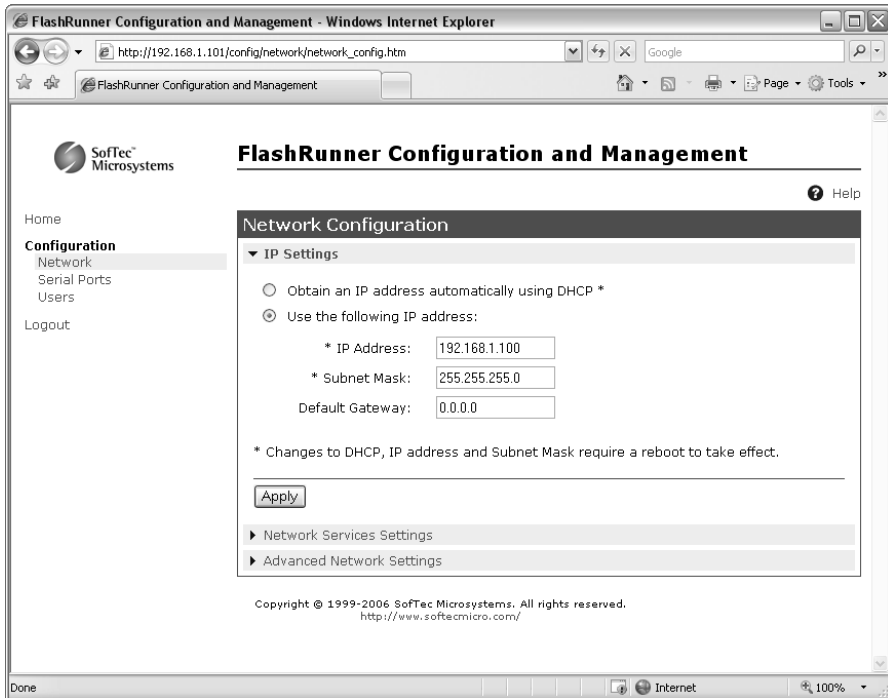


Figure 33: LAN Enabled FlashRunner Models, Network Configuration

Click the “**Apply**” button to submit your changes. FlashRunner will use the new IP address the next time it is powered on.



**Note:** if more than one FlashRunner is connected to the same local area network, each instrument needs to have a unique IP address.

# 10 Target-Specific Commands and Connections

## 10.1 Overview

The arguments of the `TCPSETPAR` and `TCPCMD` commands are target-specific. The following sections lists these arguments in detail for each target family supported by FlashRunner.

Additionally, diagrams illustrate typical connection to those target devices. All lines shown are connected to FlashRunner “ISP” connector.



**Note:** *when you purchase a programming algorithm for a specific device, you get sample scripts for that particular device. You can use these scripts as a starting point for your specific programming needs.*

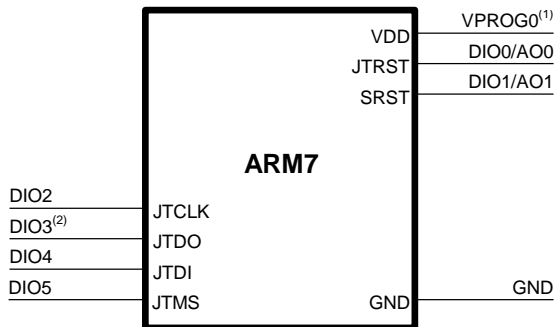
### 10.2 ARM7

#### 10.2.1 Specific Commands

Table 8: ARM7 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR FCPU <frequency Hz>	<b>Specifies the CPU clock frequency (Hz)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time $\mu$ s>	<b>Sets the Reset up time (<math>\mu</math>s)</b>
TCSETPAR RSTDOWN <time $\mu$ s>	<b>Sets the Reset down time (<math>\mu</math>s)</b>
TCSETPAR RSTDELAY <time $\mu$ s>	<b>Sets the Reset delay time (<math>\mu</math>s)</b>
<b>TPCMD Command</b>	
TPCMD SETJCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD SECTORERASE F <sect start addr>	<b>Erases a Flash sector</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD WRITE16 <tgt addr> <data>	<b>Writes a 16-bit data to a specific non-Flash address</b>
TPCMD READ16 <tgt addr>	<b>Reads a 16-bit data from a specific address</b>
TPCMD WRITE32 <tgt addr> <data>	<b>Writes a 32-bit data to a specific non-Flash address</b>
TPCMD READ32 <tgt addr>	<b>Reads a 32-bit data from a specific address</b>
TPCMD RUN	<b>Runs target application</b>

## 10.2.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> An external pull-up resistor is recommended when communicating at high frequencies

Figure 34: Example of Connection to an ARM7 Target Device

## 10.3 ARM9

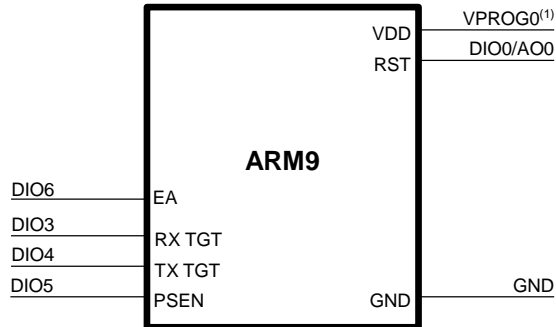
### 10.3.1 Specific Commands

Table 9: ARM9 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR CMODE	<b>Sets the communication mode</b>
TCSETPAR BAUDRATE <frequency Hz>	<b>Sets the baudrate (Hz)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash memory(F), Eeprom memory(E).</b>
TPCMD MASSErase F E	<b>Mass erases Flash memory (F), Eeprom memory(E).</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs Flash memory (F), Eeprom memory(E).</b>
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (F), Eeprom memory(E).</b>
TPCMD PROGRAM SSB <security level>	<b>Programs lock bits via UART.</b>
TPCMD READ SSB <security level>	<b>Reads lock bits via UART.</b>
TPCMD RUN	<b>Runs target application</b>



### 10.3.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 35: Example of Connection to an ARM9 Target Device

## 10.4 Atmel AVR8

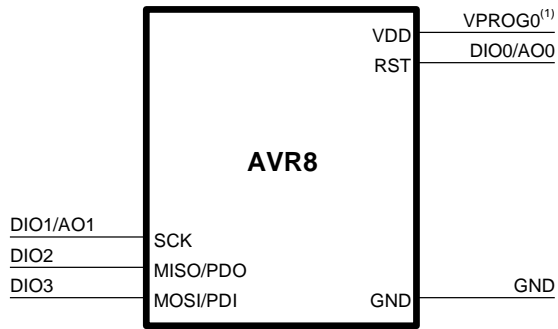
### 10.4.1 Specific Commands

Table 10: Atmel AVR8 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FCPU <frequency Hz>	Specifies the CPU clock frequency (Hz)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks memory (Flash or EEPROM)
TPCMD MASSERASE	Mass erases memory (both Flash and EEPROM)
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs memory (Flash or EEPROM)
TPCMD PROGRAM FLB FHB EFB <fuse byte>	Programs fuse bits (Fuse Low Bits, Fuse High Bits or Extended Fuse Bits)
TPCMD PROGRAM LKB <lock byte>	Programs Lock Bits
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	Verifies memory (Flash or EEPROM)
TPCMD VERIFY FLB FHB EFB <fuse byte>	Verifies fuse bits (Fuse Low Bits, Fuse High Bits or Extended Fuse Bits)
TPCMD VERIFY LKB <lock byte>	Verifies Lock Bits
TPCMD READ F E <tgt start addr> <len>	Reads len words starting from tgt start addr from Flash (F) or EEPROM (E) memory
TPCMD READ FLB FHB EFB LKB	Reads Fuse Low Bits, Fuse High Bits, Extended Fuse Bits or Lock Bits
TPCMD SAVECALIBRVALUE F E <addr> <frequency Hz>	Retrieves the factory programmed calibration value (for the internal RC oscillator) for the specified frequency and saves it to the specified Flash or EEPROM location
TPCMD RUN	Runs target application

10

### 10.4.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 36: Example of Connection to an AVR8 Target Device

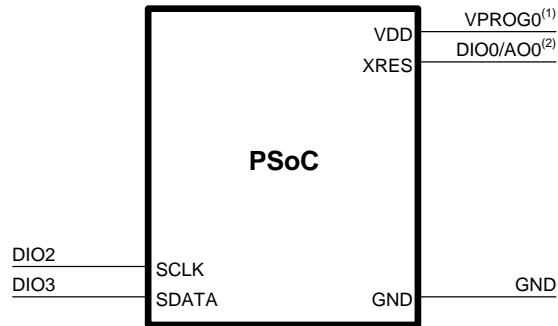
## 10.5 Cypress PSoC

### 10.5.1 Specific Commands

Table 11: Cypress PSoC Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR ENTRY_MODE RESET POWER_ON	<b>Specifies how FlashRunner will enter the programming mode (by hardware Reset or at board power on)</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the serial clock frequency</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM F SEC <src offset> <tgt start addr> <len>	<b>Programs Flash memory (F) or Security blocks (SEC)</b>
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len words starting from tgt start addr from Flash memory</b>
TPCMD READ S	<b>Returns the device checksum</b>

## 10.5.2 Connections ISSP



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Only for devices with the XRES pin

Figure 37: Example of Connection to a PsoC Target Device

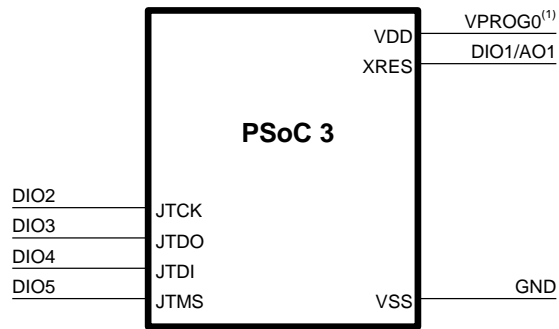
# 10.6 Cypress PSoC 3

## 10.6.1 Specific Commands

Table 12: Cypress PSoC 3 Specific Commands for JTAG interface.

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV).</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).</b>
TCSETPAR CLOCKOUT <frequency Hz>	<b>Sets the clock out frequency provided by the programmer.</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms).</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms).</b>
TCSETPAR RSTUP <time μs>	<b>Sets the reset up time (us).</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the reset down time (us).</b>
TCSETPAR CONFDATA <filename.FRB>	<b>Sets the Configuration Data filename.</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the serial clock frequency.</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE F E	<b>Erases Flash memory (F) or EEPROM memory (E).</b>
TPCMD ECC <src ECC addr>	<b>Checks for ECC enable/disable in the source file.</b>
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash memory (F) or EEPROM memory (E).</b>
TPCMD PROGRAM F E <src start addr> <tgt start addr> <len>	<b>Programs Flash memory (F) or EEPROM memory (E).</b>
TPCMD PROGRAM L W <src start addr>	<b>Programs the Non Volatile Latch (L) and the Write Once Non Volatile Latch (W).</b>
TPCMD VERIFY F E <src start addr> <tgt start addr> <len>	<b>Verifies Flash memory (F) or EEPROM memory (E).</b>
TPCMD PROTECT P <src start addr>	<b>Programs (P) the Protection data.</b>
TPCMD PROTECT V <src start addr>	<b>Verifies (V) the Protection data.</b>
TPCMD VERIFY S <src start addr> <tgt start addr> <len>	<b>Verifies the Checksum value (S).</b>
TPCMD READ F E <tgt start addr> <len>	<b>Reads Flash memory (F) or EEPROM memory (E).</b>
TPCMD RUN	<b>Executes the firmware programmed in the target device.</b>

## 10.6.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 38: Example of Connection to a Cypress PSoC 3 Target Device through JTAG interface.

## 10.7 ELMOS EL16

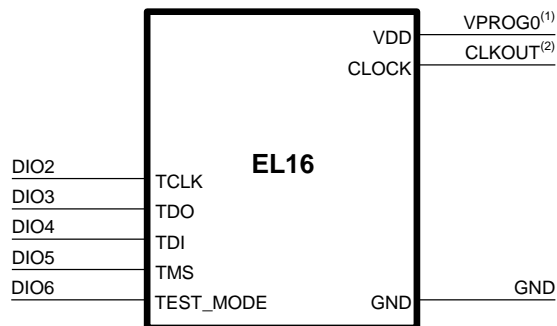
### 10.7.1 Specific Commands

Table 13: ELMOS EL16 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory by reading back all written data</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len words starting from tgt start addr from Flash memory</b>



## 10.7.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 39: Example of Connection to an EL16 Target Device

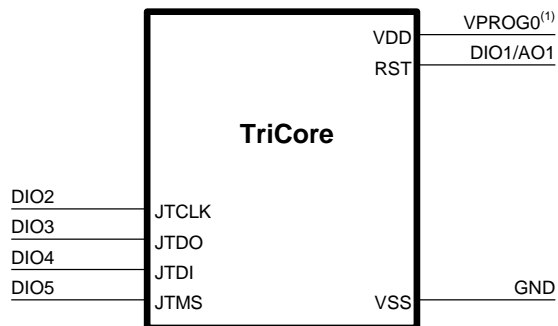
## 10.8 Infineon TriCore

### 10.8.1 Specific Commands

Table 14: Infineon TriCore Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV).</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms).</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms).</b>
<b>TCSETPAR RSTUP &lt;time μs&gt;</b>	<b>Sets the reset up time (us).</b>
<b>TCSETPAR RSTDOWN &lt;time μs&gt;</b>	<b>Sets the reset down time (us).</b>
TCSETPAR FCPU <frequency Hz>	<b>Sets the internal CPU frequency.</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the serial clock frequency.</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE P D	<b>Erases P-Flash or D-Flash memory.</b>
TPCMD SECTORERASE P D U <tgt start addr>	<b>Sector erases P-Flash or D-Flash memory or UCBlock.</b>
TPCMD BLANKCHECK P D <tgt start addr> <len>	<b>Blank checks P-Flash or D-Flash memory.</b>
TPCMD PROGRAM P D <src offset> <tgt start addr> <len>	<b>Programs P-Flash memory (P) or D-Flash memory (D).</b>
TPCMD VERIFY P D R S <src offset> <tgt start addr> <len>	<b>Verifies P-Flash memory (P) or D-Flash memory (D). Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).</b>
TPCMD WRITE32 <tgt start addr> <data>	<b>Write 32 bits data.</b>

## 10.8.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 40: Example of Connection to a TriCore Target Device

## 10.9 Freescale 56F800E

### 10.9.1 Specific Commands

Table 15: Freescale 56F800E Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR INIT_JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency during initialization operations (Hz)</b>
TCSETPAR PROG_JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency during programming operations (Hz)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE C	<b>Unsecures the device and mass erases Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory (doesn't work if device is secured)</b>
TPCMD ERASE F <tgt start addr> <len>	<b>Erases one or more Flash memory sectors</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory by reading back all written data</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads Flash memory</b>
TPCMD RUN	<b>Runs target application</b>



**Note:** the S-Rec file used to create the FlashRunner Binary File must use byte addressing. This can be done by enabling the following CodeWarrior linker options:

- *Generate S-Record File;*
- *Generate Byte Addresses.*

Additionally, **“EOL Character”** should be set to **“DOS”** (see figure below).

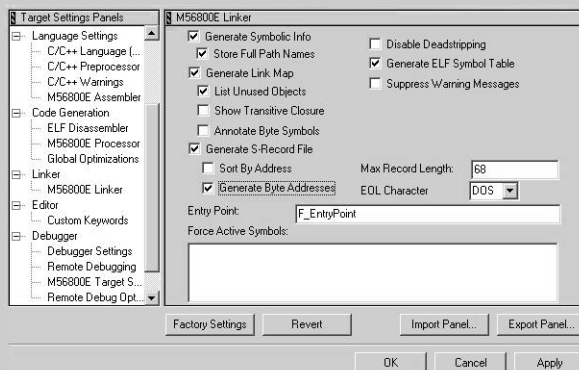
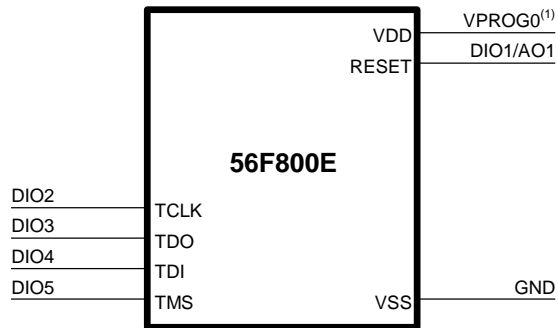


Figure 41: Correct Linker Settings for 56F800E S-Rec File Generation

### 10.9.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 42: Example of Connection to a 56F800E Target Device

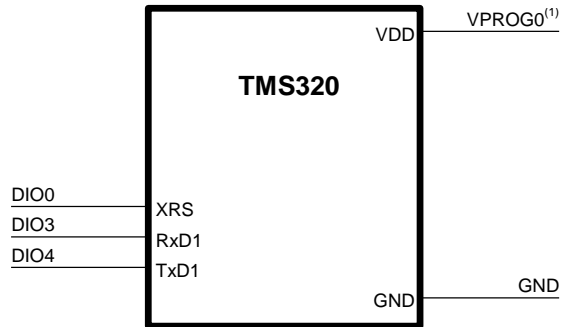
## 10.10 Texas Instruments TMS320 Devices

### 10.10.1 Specific Commands

Table 16: Texas Instruments TMS320 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets the auxiliary VDD voltage (mV)</b>
TCSETPAR RSTDOWN <time us>	<b>Reset down time (uS)</b>
TCSETPAR RSTUP <time us>	<b>Reset up time (uS)</b>
TCSETPAR RSTDRV	<b>Reset driving mode</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR CMODE	<b>Set the communication mode</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TCSETPAR FPLL <frequency Hz>	<b>Set the internal system clock frequency (Hz)</b>
TCSETPAR BAUDRATE <bit/s>	<b>Set the UART communication baudrate, (bit/s)</b>
TCSETPAR PWD SRC <addr> TCSETPAR PWD CONST <word> <word> <word> <word> <word> <word> <word> <word>	<b>Set the PWL values necessary to unlock the device</b>
<b>TPCMD Command</b>	
TPCMD CONNECT	<b>Power on the board and connect the programmer to it</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory. Addresses and length are expressed in 16-bit word.</b>
TPCMD VERIFY F E R S <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Addresses and length are expressed in 16-bit word.</b>
TPCMD DISCONNECT	<b>Disconnect the programmer to the board and power off it</b>
TPCMD RUN	<b>Runs target application</b>

### 10.10.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 43: Example of Connection to TMS320 Target Device



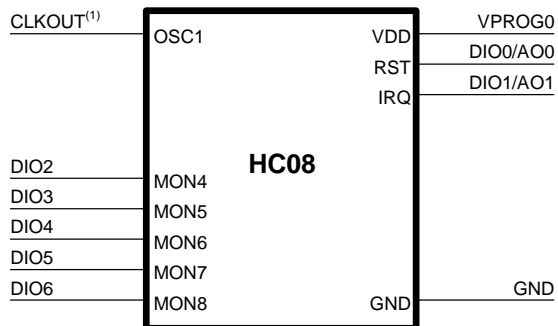
## 10.11 Freescale HC08

### 10.11.1 Specific Commands

Table 17: Freescale HC08 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR FDIV <divisor>	Sets the external oscillator frequency divisor
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR CLKOUT 25000000 12500000 6250000 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD SETPWD CONST <B0> ... <B7> TPCMD SETPWD SRC <src offset>	Sets the security bytes to be used to perform subsequent operations. The security bytes can be passed to FlashRunner directly or read from the source specified by the <code>TPSETSRC</code> command.
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks memory (Flash or EEPROM)
TPCMD MASSERASE F E	Mass erases memory (Flash or EEPROM)
TPCMD TRIM <frequency Hz> <addr>	Trims internal oscillator. Calculates the trimming value for the specified frequency, and prepares to program it at the specified address. The calculated value will be actually programmed during the next <code>TPCMD PROGRAM</code> command involving the trimming location.
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs memory (Flash or EEPROM). If the memory range includes the trimming location specified by the <code>TPCMD TRIM</code> command, this location will be programmed with the calculated trimming value.
TPCMD VERIFY F E R S <src offset> <tgt start addr> <len>	Verifies memory (Flash or EEPROM). Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).
TPCMD RUN	Runs target application

### 10.11.2 Connections



<sup>(1)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 44: Example of Connection to an HC08 Target Device

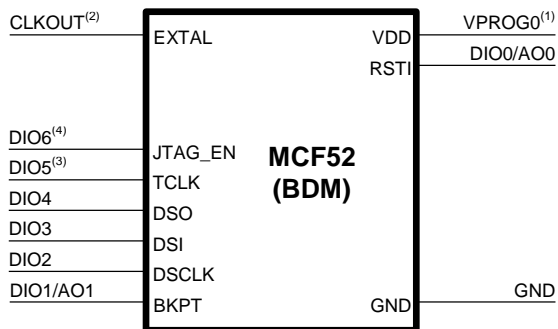
## 10.12 Freescale MCF52

### 10.12.1 Specific Commands

Table 18: Freescale MCF52 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR PROG_MODE BDM EZPORT	<b>Sets the programming method</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR PLLFREQ <frequency Hz>	<b>Sets the PLL frequency (Hz). A frequency of 0 disables the PLL.</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the serial clock frequency</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD UNSECURE	<b>Unsecures protected devices</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD PAGEERASE F <tgt page addr>	<b>Erases one Flash page</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

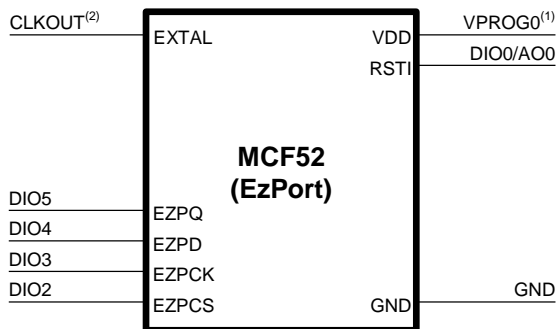
### 10.12.2 Connections (BDM Communication Mode)



- (1) Connect this line if you want FlashRunner to automatically power the target device
- (2) Connect this line if you want the target device to be clocked by FlashRunner
- (3) Connect this line only if you want to unsecure the device
- (4) Connect this line only if you want to unsecure the device, otherwise JTAG\_EN must be connected to GND

Figure 45: Example of Connection to a MCF52 Target Device (BDM Communication)

### 10.12.3 Connections (EzPort Communication Mode)



- (1) Connect this line if you want FlashRunner to automatically power the target device
- (2) Connect this line if you want the target device to be clocked by FlashRunner

Figure 46: Example of Connection to an MCF52 Target Device (EzPort Communication)

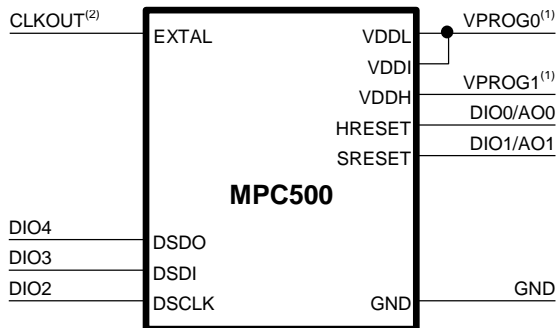
## 10.13 Freescale MPC500

### 10.13.1 Specific Commands

Table 19: Freescale MPC500 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz)
TCSETPAR SCLK <frequency Hz>	Sets the serial clock frequency
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSH/PULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F S <tgt start addr> <len>	Blank checks Flash (F) or Shadow (S) memory
TPCMD MASSERASE F	Mass erases Flash and Shadow memory
TPCMD ERASE F S <tgt page addr> <len>	Erases one or more Flash (F) or Shadow (S) pages
TPCMD PROGRAM	Programs Flash (F) or Shadow (S) memory
TPCMD VERIFY	Verifies Flash (F) or Shadow (S) memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).
TPCMD READ F S <tgt start addr> <len>	Reads len bytes starting from tgt start addr from Flash (F) or Shadow (S) memory
TPCMD CHANGESENSOR A B 0 1 2 3	Changes the value of the two sensor bits for Flash module A or B
TPCMD WRITE8 <tgt addr> <data>	Writes a 8-bit data to a specific non-Flash address
TPCMD READ8 <tgt addr>	Reads a 8-bit data from a specific address
TPCMD WRITE16 <tgt addr> <data>	Writes a 16-bit data to a specific non-Flash address
TPCMD READ16 <tgt addr>	Reads a 16-bit data from a specific address
TPCMD WRITE32 <tgt addr> <data>	Writes a 32-bit data to a specific non-Flash address
TPCMD READ32 <tgt addr>	Reads a 32-bit data from a specific address

### 10.13.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 47: Example of Connection to a MPC500 Target Device

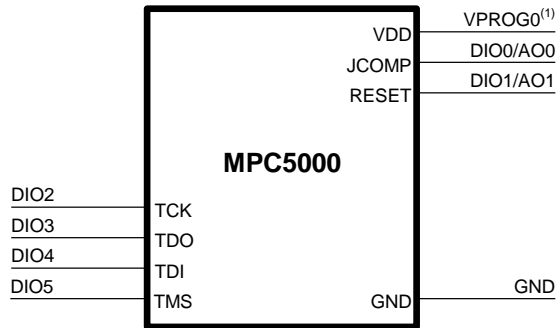
## 10.14 Freescale MPC5000

### 10.14.1 Specific Commands

Table 20: Freescale MPC5000 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR PLLFREQ <frequency Hz>	<b>Sets the PLL frequency (Hz)</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHFULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR LSM YES NO	<b>Select Lock Step Mode (only for devices that support Lock Step Mode)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F S <tgt start addr> <len>	<b>Blank checks Flash (F) or Shadow (S) memory</b>
TPCMD MASSERASE F S	<b>Mass erases Flash (F) or Shadow (S) memory</b>
TPCMD ERASE F S <tgt page addr> <len>	<b>Erases one or more Flash (F) or Shadow (S) pages</b>
TPCMD PROGRAM	<b>Programs Flash (F) or Shadow (S) memory</b>
TPCMD VERIFY	<b>Verifies Flash (F) or Shadow (S) memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).</b>
TPCMD DUMP F E S <src offset> <tgt start addr>	<b>Creates a binary dump file of the selected memory type</b>

### 10.14.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 48: Example of Connection to a MPC5000 Target Device



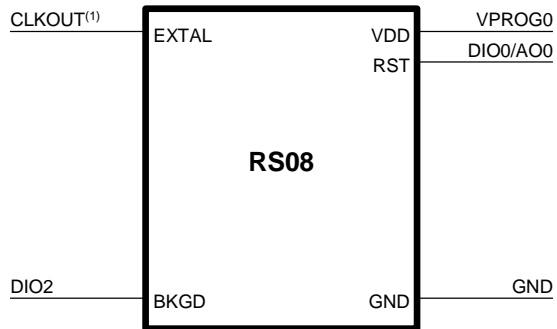
## 10.15 Freescale RS08

### 10.15.1 Specific Commands

Table 21: Freescale RS08 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR BDM_ENTRY_MODE HW_RST SW_RST PWUP_RST	Specifies how FlashRunner will enter the BDM mode (by hardware Reset, by software Reset or by power-up Reset)
TCSETPAR CLKOUT 25000000 12500000 6250000 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks memory (Flash or EEPROM)
TPCMD MASSERASE F E	Mass erases memory (Flash or EEPROM)
TPCMD TRIM <frequency Hz> <addr> <tolerance_pct>	Trims internal oscillator to a given precision. Calculates the trimming value for the specified frequency, and prepares to program it at the specified address. The calculated value will be actually programmed during the next TPCMD PROGRAM command involving the trimming location.
TPCMD PROGRAM	Programs memory (Flash or EEPROM). If the memory range includes the trimming location specified by the TPCMD TRIM command, this location will be programmed with the calculated trimming value.
TPCMD VERIFY	Verifies memory (Flash or EEPROM)
TPCMD IS_DEVICE_SECURED	Returns whether the device is secured (1) or not (0)
TPCMD RUN	Runs target application

### 10.15.2 Connections



<sup>(1)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 49: Example of Connection to an RS08 Target Device

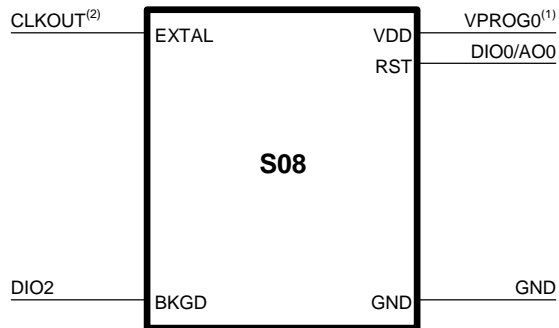
## 10.16 Freescale S08

### 10.16.1 Specific Commands

Table 22: Freescale S08 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FLL_OSC <frequency Hz>	<b>Sets the FLL frequency (Hz). A frequency of 0 disables the FLL.</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time $\mu$ s>	<b>Sets the Reset up time (<math>\mu</math>s)</b>
TCSETPAR RSTDOWN <time $\mu$ s>	<b>Sets the Reset down time (<math>\mu</math>s)</b>
TCSETPAR RSTDRV PUSHFULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR BDM_ENTRY_MODE HW_RST SW_RST PWUP_RST	<b>Specifies how FlashRunner will enter the BDM mode (by hardware Reset, by software Reset or by power-up Reset)</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks memory (Flash or EEPROM)</b>
TPCMD MASSERASE F E	<b>Mass erases memory (Flash or EEPROM)</b>
TPCMD TRIM <frequency Hz> <addr> <tolerance_pct>	<b>Trims internal oscillator to a given precision. Calculates the trimming value for the specified frequency, and prepares to program it at the specified address. The calculated value will be actually programmed during the next TPCMD PROGRAM command involving the trimming location.</b>
TPCMD PROGRAM	<b>Programs memory (Flash or EEPROM). If the memory range includes the trimming location specified by the TPCMD TRIM command, this location will be programmed with the calculated trimming value.</b>
TPCMD VERIFY	<b>Verifies memory (Flash or EEPROM)</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len bytes starting from tgt start addr from Flash memory</b>
TPCMD DUMP	<b>Dumps memory content (Flash or EEPROM) to the specified destination</b>
TPCMD IS_DEVICE_SECURED	<b>Returns whether the device is secured (1) or not (0)</b>
TPCMD RUN	<b>Runs target application</b>

## 10.16.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 50: Example of Connection to a S08 Target Device

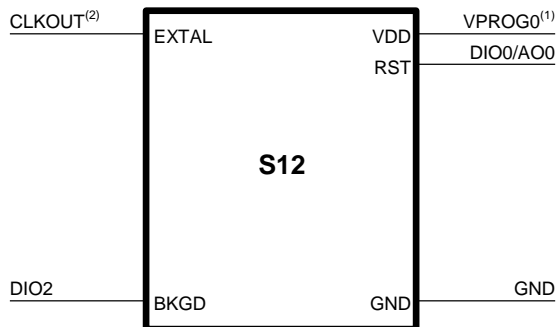
## 10.17 Freescale S12

### 10.17.1 Specific Commands

Table 23: Freescale S12 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSHFULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz). A frequency of 0 disables the PLL.
TCSETPAR REFDIV <divisor>	Sets the PLL frequency divisor register
TCSETPAR SYNRM <multiplier>	Sets the PLL frequency multiplier register
TCSETPAR CLKOUT 25000000 12500000 6250000 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD UNSECURE	Unsecures protected devices
TPCMD BLANKCHECK F E <tgt start addr> <len> P G	Blank checks memory (Flash or EEPROM). Target addressing is paged (P) or global (G).
TPCMD MASSERASE F E	Mass erases memory (Flash or EEPROM)
TPCMD PROGRAM	Programs memory (Flash or EEPROM). Source data addressing is banked (B) or linear (L). Target addressing is paged (P) or global (G).
TPCMD VERIFY	Verifies memory (Flash or EEPROM). Source data addressing is banked (B) or linear (L). Target addressing is paged (P) or global (G). Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).
TPCMD READ F E <tgt start addr> <len> P G	Reads len bytes starting from tgt start addr from Flash (F) or EEPROM (E) memory. Target addressing is paged (P) or global (G).
TPCMD RUN	Runs target application

## 10.17.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 51: Example of Connection to a S12 Target Device

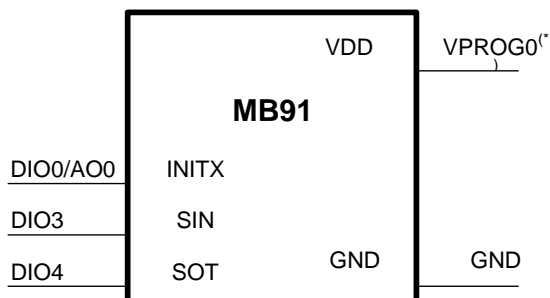
## 10.18 Fujitsu MB91

### 10.18.1 Specific Commands

Table 24: Fujitsu MB91 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
<b>TCSETPAR RSTUP &lt;time μs&gt;</b>	Sets the reset up time (us)
<b>TCSETPAR RSTDOWN &lt;time μs&gt;</b>	Sets the reset down time (us)
TCSETPAR RSTDRV PUSH_PULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR CMODE UART MDI	Communication mode settings
TCSETPAR FOSC <value>	External clock source frequency
TCSETPAR DIVM <value>	Set the DIVM divisor (0 to 16)
TCSETPAR DIVN <value>	Set the DIVN divisor (0 to 64)
TCSETPAR DIVG <value>	Set the DIVG divisor (0 to 16). This parameter must be greater than DIVM
TCSETPAR MULG <value>	Set the MULG multiplier (0 to 256)
TCSETPAR FPLL <value>	Sets the PLL frequency (Hz)
TCSETPAR BAUDRATE <value>	Sets the baud rate (bps)
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator
<b>TPCMD Command</b>	
TPCMD CONNECT	Connect the programmer to the target and test the communication
TPCMD DISCONNECT	Disconnect the programmer from the target
TPCMD MASSERASE F D	Mass erases Flash memory, Data flash(D)
TPCMD BLANKCHECK F <tgt_start_addr> <len>	Blank checks Flash memory, Data flash(D)
TPCMD PROGRAM F <src_offset> <tgt_start_addr> <len>	Programs Flash memory, Data flash(D)
TPCMD VERIFY F R <src_offset> <tgt_start_addr> <len>	Verifies Flash memory, Data flash, reading back all the written data
TPCMD READ F <tgt_start_addr> <len>	Reads len locations of Flash memory starting from the tgt_start_addr address. len specifies the number of locations to be read.
TPCMD RUN	Runs target application

### 10.18.2 Connections UART

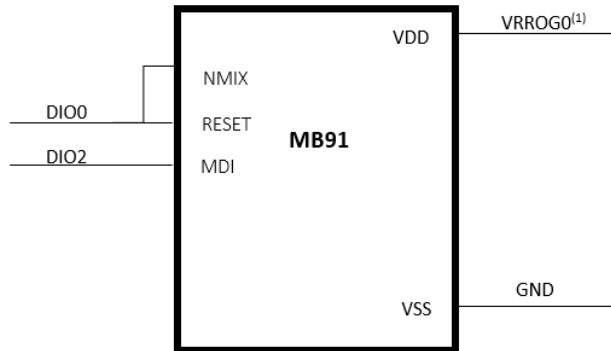


<sup>(1)</sup>Connect only if necessary. If you power the board externally, doesn't connect it to the board.

Figure 52: Example of Connection to a MB91 Target Device through UART

### 10.18.3 Connections MDI





<sup>(1)</sup> Connect this line if you want Flash Runner to automatically power the target device

Figure 53: Example of Connection to a MB91 Target Device through MDI

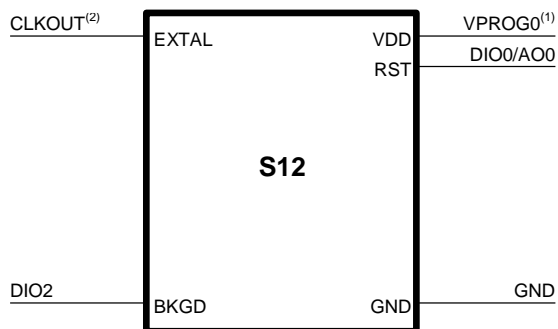
## 10.19 Freescale S12 (FSL\_D Algorithm)

### 10.19.1 Specific Commands

Table 25: Freescale S12 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz). A frequency of 0 disables the PLL.
TCSETPAR REFDIV <divisor>	Sets the PLL frequency divisor register
TCSETPAR SYNRM <multiplier>	Sets the PLL frequency multiplier register
TCSETPAR CLKOUT 25000000 12500000 6250000 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD UNSECURE	Unsecures protected devices
TPCMD BLANKCHECK F E <tgt start addr> <len> P G	Blank checks memory (Flash or EEPROM). Target addressing is paged (P) or global (G).
TPCMD MASSErase F E	Mass erases memory (Flash or EEPROM)
TPCMD PROGRAM	Programs memory (Flash, EEPROM, System or Emulated EEPROM). Source data addressing is banked (B) or linear (L). Target addressing is paged (P) or global (G).
TPCMD VERIFY F E EEE R S <src offset> <tgt start addr>	Verifies memory (Flash, EEPROM or Emulated EEPROM). Source data addressing is banked (B) or linear (L). Target addressing is paged (P) or global (G). Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).
TPCMD READ F E S <tgt start addr> <len> P G	Reads len bytes starting from tgt start addr from Flash (F), EEPROM (E) or System(S) memory. Target addressing is paged (P) or global (G).
TPCMD DUMP F E S <tgt start addr> <len> P G	Dumps len bytes starting from tgt start addr from Flash (F), EEPROM (E) or System(S) memory. Target addressing is paged (P) or global (G).

## 10.19.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 54: Example of Connection to a S12 Target Device

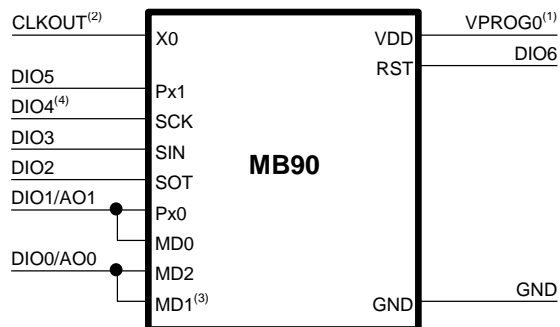
## 10.20 Fujitsu MB90

### 10.20.1 Specific Commands

Table 26: Fujitsu MB90 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CMODE UART	<b>Sets the communication mode</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the UART mode baud rate (bps)</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR PLLFREQ <frequency Hz>	<b>Sets the PLL frequency (Hz)</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSErase F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

## 10.20.2 Connections



- (1) Connect this line if you want FlashRunner to automatically power the target device  
(2) Connect this line if you want the target device to be clocked by FlashRunner  
(3) Connect this line only if MD1 is not already tied to VDD  
(4) Connect this line only for devices with synchronous communication

Figure 55: Example of Connection to a MB90 Target Device

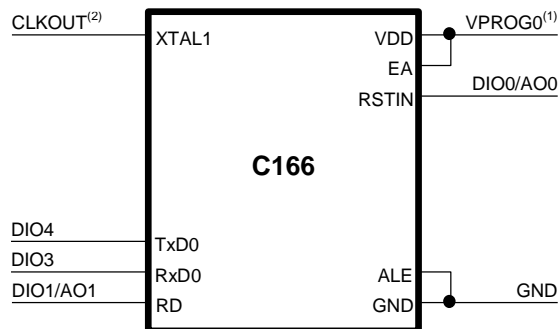
## 10.21 Infineon C166

### 10.21.1 Specific Commands

Table 27: Infineon C166 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FCPU <frequency Hz>	<b>Sets the CPU frequency (Hz)</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR PLLFREQ <frequency Hz>	<b>Sets the PLL frequency (Hz)</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the communication baud rate (bit/s)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSErase F	<b>Mass erases Flash memory</b>
TPCMD SECTORERASE F <tgt page addr>	<b>Erases one Flash page</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

## 10.21.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device. Lines EA and ALE must always be connected to VDD and GND, respectively.
- <sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 56: Example of Connection to a C166 Target Device

## 10.22 Infineon XC166

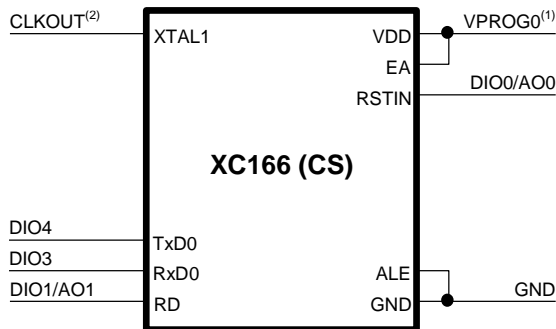
### 10.22.1 Specific Commands

Table 28: Infineon XC166 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE JTAG UART	<b>Sets the communication mode</b>
TCSETPAR FCPU <frequency Hz>	<b>Sets the CPU frequency (Hz)</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR PLLIDIV <value>	<b>Sets the PLL input divider (0 to 3)</b>
TCSETPAR PLODIV <value>	<b>Sets the PLL output divider (0 to 14)</b>
TCSETPAR PLLMUL <value>	<b>Sets the PLL multiplication factor (7 to 31)</b>
TCSETPAR PLLFREQ <frequency Hz>	<b>Sets the PLL frequency (Hz)</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the UART communication baud rate (bit/s)</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD PROTECT <word 1> <word 2> <word 3> <word 4>	<b>Protects the device</b>
TPCMD UNPROTECT <word 1> <word 2> <word 3> <word 4>	<b>Unprotects the device</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD SECTORERASE F <tgt page addr>	<b>Erases one Flash page</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

### 10.22.2 Connections (UART Communication Mode, CS Series)

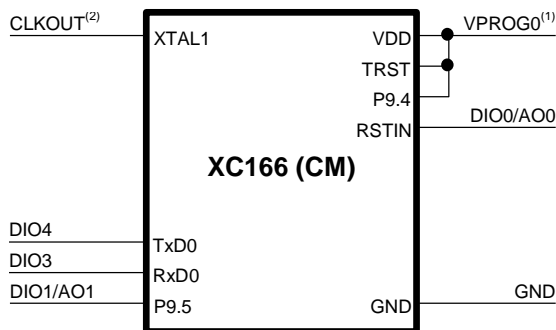




- (1) Connect this line if you want FlashRunner to automatically power the target device. Lines EA and ALE must always be connected to VDD and GND, respectively.
- (2) Connect this line if you want the target device to be clocked by FlashRunner

Figure 57: Example of Connection to a XC166 (UART Mode, CS Series) Target Device

### 10.22.3 Connections (UART Communication Mode, CM Series)



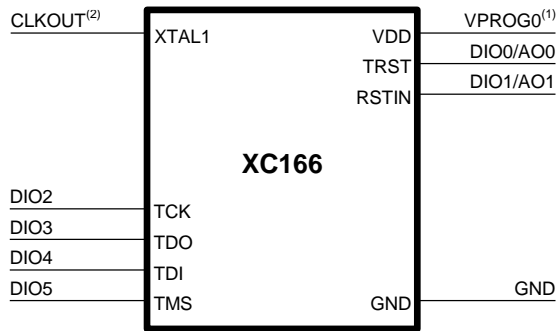
- (1) Connect this line if you want FlashRunner to automatically power the target device. Lines TRST and P9.4 must always be connected to VDD.
- (2) Connect this line if you want the target device to be clocked by FlashRunner

Figure 58: Example of Connection to a XC166 (UART Mode, CM Series) Target Device

### 10.22.4 Connections (JTAG Communication Mode)

## Target-Specific Commands and Connections

---



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 59: Example of Connection to a XC166 Target Device (JTAG Mode)

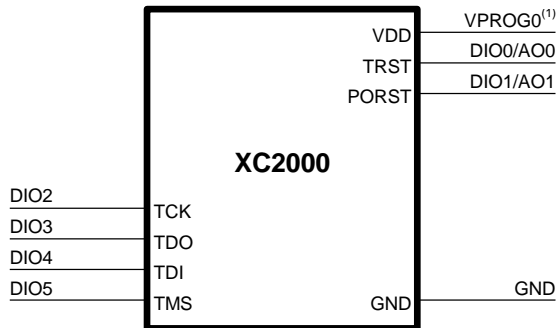
## 10.23 Infineon XC2000

### 10.23.1 Specific Commands

Table 29: Infineon XC2000 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR JTCLK <frequency Hz>	Sets the JTAG clock frequency (Hz)
TCSETPAR PLLPDIV <value>	Sets the PLL input divider (1 to 16)
TCSETPAR PLLKDIV <value>	Sets the PLL output divider (1 to 1024)
TCSETPAR PLLNDIV <value>	Sets the PLL multiplication factor (1 to 64)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	Verifies Flash memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).
TPCMD RUN	Runs target application

### 10.23.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 60: Example of Connection to a XC2000 Target Device

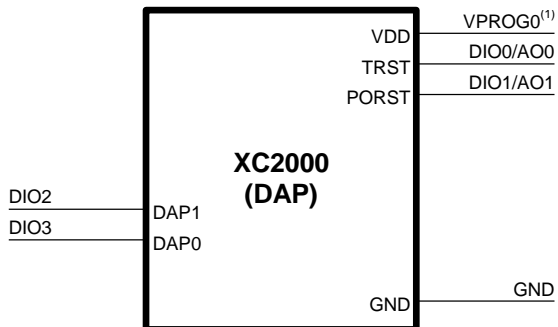
## 10.24 Infineon XC2000 (DAP Protocol)

### 10.24.1 Specific Commands

Table 30: Infineon XC2000 (DAP Protocol) Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR DAPCLK <frequency Hz>	Sets the DAP clock frequency (Hz)
TCSETPAR PLLPDIV <value>	Sets the PLL input divider (1 to 16)
TCSETPAR PLLKDIV <value>	Sets the PLL output divider (1 to 1024)
TCSETPAR PLLNDIV <value>	Sets the PLL multiplication factor (1 to 64)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
TCSETPAR CMODE DAP	Sets the DAP communication mode
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory
TPCMD VERIFY F R S MR MS <src offset> <tgt start addr> <len>	Verifies Flash memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure). MR Margin Readout method. MS Margin Checksum method
TPCMD DUMP F <dst start addr> <tgt start addr> <len>	Creates a binary dump file with the Flash memory Content
TPCMD RUN	Runs target application

### 10.24.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 61: Example of Connection to a XC2000 (DAP protocol) Target Device

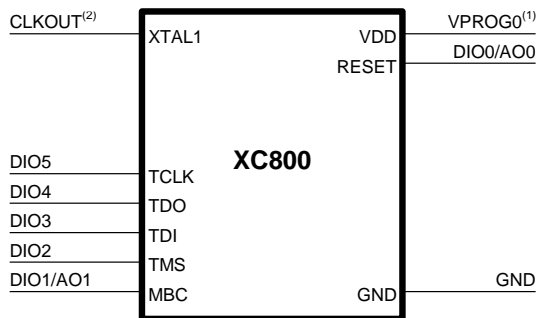
## 10.25 Infineon XC800 (JTAG)

### 10.25.1 Specific Commands

Table 31: Infineon XC800 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD SECTORERASE F <tgt page addr>	<b>Erases one Flash page</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len bytes starting from tgt start addr from Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

### 10.25.2 Connections JTAG

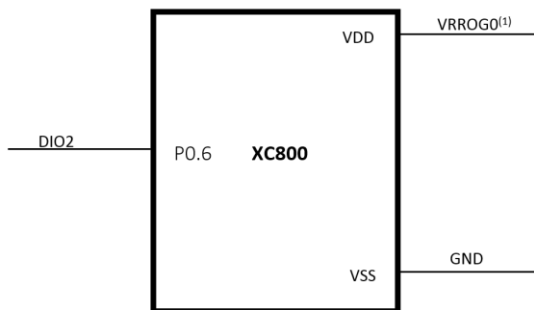


<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 62: Example of Connection to a XC800 Target Device

### 10.25.3 Connections UART



<sup>(1)</sup> Connect this line if you want Flash Runner to automatically power the target device

Figure 63: Example of Connection to a XC800 Target Device



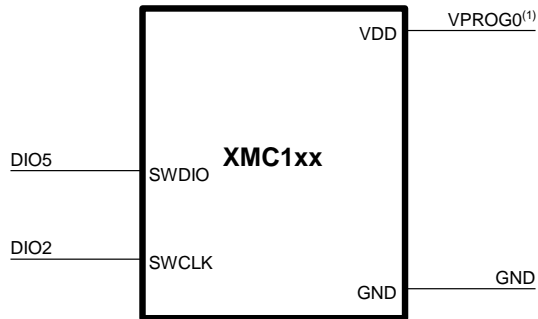
## 10.26 Infineon XMC1xx

### 10.26.1 Specific Commands

Table 32: Infineon XMC1xx Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time $\mu$ s>	<b>Sets the Reset up time (<math>\mu</math>s)</b>
TCSETPAR RSTDOWN <time $\mu$ s>	<b>Sets the Reset down time (<math>\mu</math>s)</b>
TCSETPAR RSTDRV PUSH/PULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD SECTORERASE F <tgt page addr>	<b>Erases one Flash page</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len bytes starting from tgt start addr from Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

## 10.26.2 Connections SWD



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 64: Example of Connection to a XMC1xx Target Device

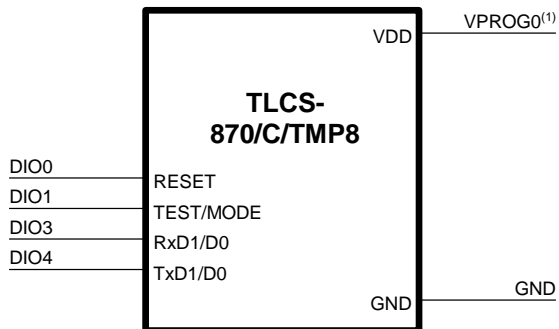
## 10.27 Toshiba TLCS-870/C/TMP8

### 10.27.1 Specific Commands

Table 33: Toshiba TLCS-870/C/TMP8 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the auxiliary VDD voltage (mV)
TCSETPAR RSTDOWN <time us>	Reset down time (uS)
TCSETPAR RSTUP <time us>	Reset up time (uS)
TCSETPAR RSTDRV	Reset driving mode
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR CLOCKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR FPLL <frequency Hz>	Set the internal system clock frequency (Hz)
TCSETPAR BAUDRATE <bit/s>	Set the UART communication baudrate, (bit/s)
TCSETPAR PWD_CNT_ADDR <address>	Sets the Flash address where the password size is stored
TCSETPAR PWD_FLASH_ADDR <address>	Sets the Flash address where the password is stored
TCSETPAR PWD <size> <data 1> <data 2> <data 3> <data 4> <data 5> <data 6> <data 7> <data 8>	Specifies the size of the password and the data that form the password
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory. Addresses and length are expressed in 16-bit word.
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	Verifies Flash memory. Addresses and length are expressed in 16-bit word.
TPCMD SECURE	Secures the Flash memory of the device
TPCMD RUN	Runs target application

## 10.27.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 65: Example of Connection to a Toshiba TLCS-870/C/TMP8 Target Device

## 10.28 Renesas RX62

### 10.28.1 Specific Commands

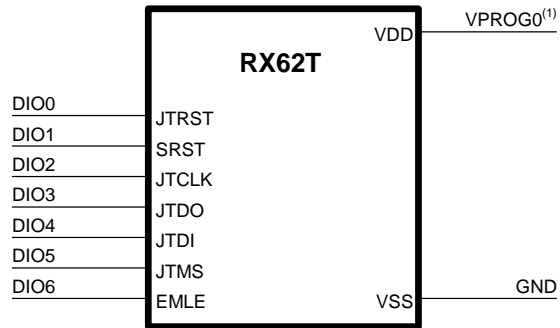
Table 34: Renesas RX62 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR FOSC <frequency Hz>	Sets the External Frequency Oscillator mounted on target board
TCSETPAR KEY_CODE CONST <value1> <value2> <value3> <value4> <value5> <value6> <value7> <value8> <value9> <valueA> <valueB> <valueC> <valueD> <valueE> <valueF> <value10>	Sets up the password as to be able to access to the device. It is necessary to specify all the 16 bytes chosen in the protection phase.
TCSETPAR KEY_CODE SRC <address>	Sets up the address of the FRB source file where the password is stored. If the device is not protected it is possible to comment both the commands TCSETPAR KEY_CODE CONST and TCSETPAR KEY_CODE SRC; it is also possible to specify a random password: if the device is not protected, it will be ignored.

## Target-Specific Commands and Connections

TCSETPAR CLKOUT 25000000 12500000 6250000 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks Flash memory (User or Data) or EEPROM memory (Data)
TPCMD MASSErase F E	Mass erases all Flash memory (F) or EEPROM memory (E)
TPCMD ERASE F E <tgt start addr> <len>	It erases blocks of locations in the Flash memory (F) or in the EEPROM memory (E). <b>tgt start address</b> is the start address of a block. <b>len</b> is such as the final address of the range to be erased is exactly the final address of a memory block. The start address is the logical address specified in each single device datasheet or User's Guide under "Memory" chapter.
TPCMD SETJCLK <frequency Hz>	selects the JTAG clock frequency (Hertz); the maximum value is 12500000Hz; it must be lower than the external oscillator frequency.
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs Flash memory (User or Data) or EEPROM memory (Data)
TPCMD VERIFY F E R <src offset> <tgt start addr> <len>	Verifies Flash memory (User or Data) or EEPROM memory (Data)
TPCMD READ F E <tgt start addr> <len>	Reads len locations of Flash memory or EEPROM memory starting from the <b>tgt start addr</b> address. <b>len</b> specifies the number of locations to be read.
TPCMD READ IDCODE	Reads the IDCODE value of the device
TPCMD READ32 <tgt start addr>	Reads a sequence of 32 bits of Flash or EEPROM memory starting from the <b>tgt start addr</b> address.
TPCMD RUN	Runs target application

## 10.28.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 66: Example of Connection to RX62T Target Device

## 10.29 NXP P89C51R

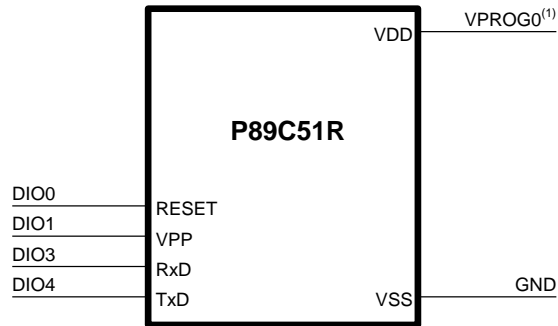
### 10.29.1 Specific Commands

Table 35: NXP P89C51R Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV).</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms).</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms).</b>
<b>TCSETPAR RSTUP &lt;time μs&gt;</b>	<b>Sets the reset up time (us).</b>
<b>TCSETPAR RSTDOWN &lt;time μs&gt;</b>	<b>Sets the reset down time (us).</b>
<b>TCSETPAR RSTDRV OPENDRAIN PUSHPULL</b>	<b>Sets the reset drive mode.</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the target oscillator frequency.</b>
TCSETPAR CLOCKOUT <frequency Hz>	<b>Sets the clock frequency provided by the programmer.</b>
TCSETPAR BAUDRATE <baudrate bit/s>	<b>Sets the baudrate.</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE C SB	<b>Erases the whole chip (C) or the Status Byte (SB).</b>
TPCMD ERASE F <tgt start addr> <len>	<b>Sector erases Flash (F).</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory (F).</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory (F).</b>
TPCMD PROGRAM BV SEC FX2 <value>	<b>Programs Boot Vector (BV), Security Bits (SEC), Oscillator divider register (FX2).</b>
TPCMD PROGRAM SB	<b>Programs Status Byte (SB).</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (F). Reads back all written data (R).</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads Flash memory (F).</b>
TPCMD READ BV SB SEC FX2	<b>Reads Boot Vector (BV), Status Byte (SB), Security Bits (SEC), Oscillator divider register (FX2).</b>
TPCMD RUN	<b>Executes the firmware programmed in the target device memory.</b>



## 10.29.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 67: Example of Connection to a NXP P89C51R Target Device

## 10.30 XC800 (DAP and UART 1 wire protocols)

### 10.30.1 Specific Commands

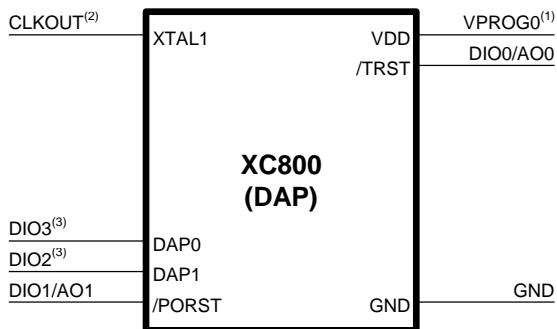
Table 36: Infineon XC800 DAP Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR JTCLK <frequency Hz>	<b>Sets the DAP clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR CMODE DAP	<b>Sets the DAP communication mode</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD SECTORERASE F <tgt page addr>	<b>Erases one Flash page</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len bytes starting from tgt start addr from Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

Table 37: Infineon XC82x UART 1 Wire Specific Commands

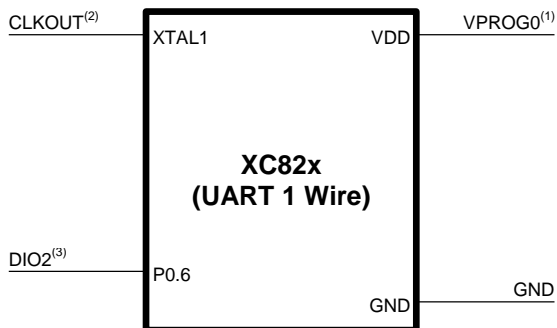
Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the UART mode baud rate (bps)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSH/PULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR CMODE UART	<b>Sets the UART communication mode</b>
<b>TPCMD Command</b>	
TPCMD ID_READ 1 0	<b>Gets USER_ID (1) data or CHIP_ID data (0)</b>
TPCMD USER_ID_CONFIG <bmi> <user_id_byte_1> <user_id_byte_2> <user_id_byte_3>	<b>Programs USER_ID bytes</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

### 10.30.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.  
<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner  
<sup>(3)</sup> An external pulldown resistor (4.7K $\Omega$ ) is required on this line

Figure 68: Example of Connection to a XC800 Target Device (DAP)



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.  
<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner  
<sup>(3)</sup> An external pullup resistor (4.7K $\Omega$ ) is required on this line

Figure 69: Example of Connection to a XC800 Target Device (UART 1 Wire)

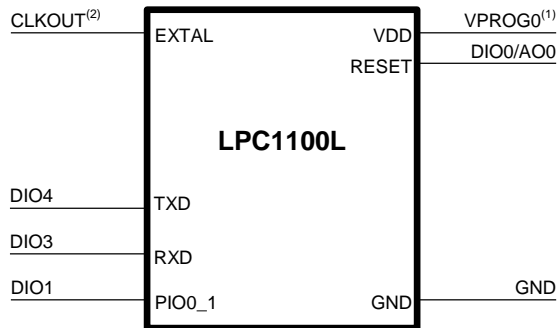
## 10.31 NXP LPC1100L Algorithm

### 10.31.1 Specific Commands

Table 38: NXP LPC1100L Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the UART baud rate (bps)</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory.</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM	<b>Programs Flash memory</b>
TPCMD VERIFY F <src offset> <tgt start addr> <len>	<b>Verifies Flash memory.</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len bytes starting from tgt start addr from Flash.</b>
TPCMD RUN	<b>Runs target application</b>

### 10.31.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 70: Example of Connection to NXP LPC1100L Target Device

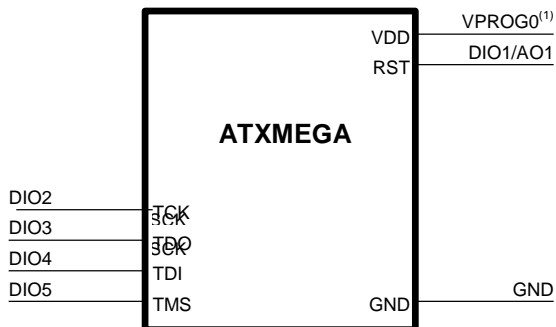
## 10.32 Atmel ATXMega

### 10.32.1 Specific Commands

Table 39: ATMEL ATXMEGA Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR SCLK <frequency Hz>	<b>Sets the serial clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR CMODE	<b>Stes communication Protocol "JTAG" or "PDI"</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash (F) or EEPROM (E) memory. The start address is not related to PDI memory map but is the logical address specified in each single device datasheet under "Memory" chapter.</b> Flash memory is organized as 16-bit locations: len specifies the number of 16-bit words to be blankchecked. EEPROM memory is organized as 8-bit locations: len specifies the number of bytes to be blankchecked
TPCMD MASSERASE C F U	'C' parameter mass erases Flash memory, EEPROM memory and user signature. Fuse bits are not erased. 'F' mass erases Flash memory, 'U' mass erase user signature. MASSERASE C command automatically set EESAVE bit of FUSEBYTE5 to 1, instead MASSERASE F command automatically set EESAVE bit of FUSEBYTE5 to 0.
TPCMD PROGRAM F E U L <src offset> <tgt start addr> <len>	<b>Programs Flash (F), EEPROM (E) memory, User signature (U) or Fuses and Locks (L).</b>
TPCMD VERIFY F E R C <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. F E Specifies Flash (F) or EEPROM (E) memory. R S Specifies Readout (R) or CRC (S)</b>
TPCMD READ F E C L U <tgt start addr> <len>	<b>Reads len locations of Flash (F), EEPROM (E), memory starting from the tgt start addr address, or reads the specified calibration Row (C), Fuse or Lock Bits (L), User Signature (U).</b>
TPCMD RUN	<b>Runs target application</b>

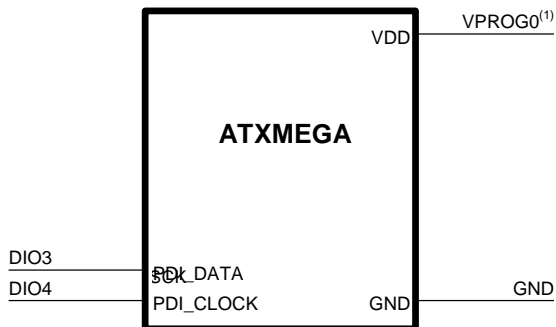
### 10.32.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 71: Example of Connection to ATXMEGA (JTAG) Target Device

### 10.32.3 Connections PDI



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 72: Example of Connection to ATXMEGA (PDI) Target Device



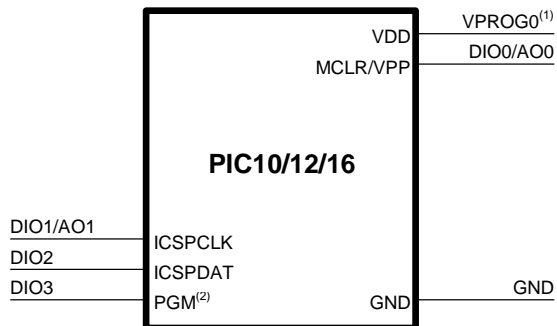
## 10.33 Microchip PIC10/12/16

### 10.33.1 Specific Commands

Table 40: Microchip PIC10/12/16 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR SCLK <frequency Hz>	Sets the serial clock frequency (Hz)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR PROGMODE HVP LVP HVD	Sets the programming mode high-voltage (HVP), low-voltage (LVP) and high-voltage VDD first (HVD) for the devices that foreseen both the high-voltage VPP first and the high-voltage VDD first
TCSETPAR VPP <voltage mV>	Sets the programming voltage (mV). Must be used only if <code>PROGMODE</code> has been set to <code>HVP</code> .
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks Flash (F) or EEPROM (E) memory. Word-addressing mode is used for Flash memory; byte-addressing mode is used for EEPROM memory.
TPCMD MASSERASE F E C	Mass erases memory (Flash, EEPROM or Complete (Flash, EEPROM, configuration word and user ID locations))
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory (including configuration word and user ID locations). Word-addressing mode is used.
TPCMD PROGRAM E B W <src offset> <tgt start addr> <len>	Programs EEPROM memory. Uses a byte (B) or word (W) addressing mode.
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash memory (including configuration word and user ID locations). Word-addressing mode is used.
TPCMD VERIFY E B W <src offset> <tgt start addr> <len>	Verifies EEPROM memory. Uses a byte (B) or word (W) addressing mode.
TPCMD READ F E <tgt start addr> <len>	Reads <code>len</code> words starting from <code>tgt start addr</code> from Flash (F) or EEPROM (E) memory. Word-addressing mode is used for Flash memory; byte-addressing mode is used for EEPROM memory.
TPCMD RUN	Runs target application

### 10.33.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Only for devices that support low-voltage programming mode

Figure 73: Example of Connection to a PIC10/12/16 Target Device

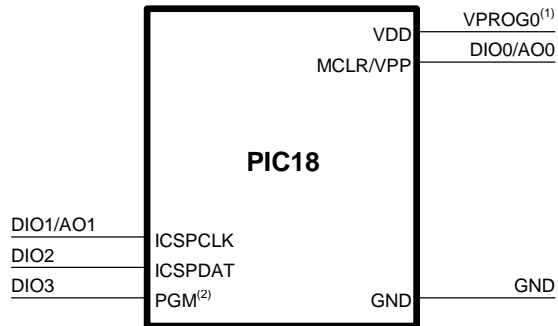
## 10.34 Microchip PIC18

### 10.34.1 Specific Commands

Table 41: Microchip PIC18 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR SCLK <frequency Hz>	Sets the serial clock frequency (Hz)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR PROGMODE HVP LVP	Sets the programming mode (high-voltage or low-voltage)
TCSETPAR VPP <voltage mV>	Sets the programming voltage (mV). Must be used only if PROGMODE has been set to HVP.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks Flash or EEPROM memory
TPCMD MASSERASE F <tgt row start addr> <len>	Mass erases Flash memory, if the device supports single-row Flash erasing
TPCMD MASSERASE E C	Mass erases memory (EEPROM or Complete (Flash, EEPROM, configuration words and user ID locations))
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs Flash memory (including configuration words and user ID locations) or EEPROM memory
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	Verifies Flash memory (including configuration words and user ID locations) or EEPROM memory
TPCMD READ F E <tgt start addr> <len>	Reads len bytes starting from tgt start addr from Flash (F) or EEPROM (E) memory
TPCMD RUN	Runs target application

### 10.34.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Only for devices that support low-voltage programming mode

Figure 74: Example of Connection to a PIC18 Target Device

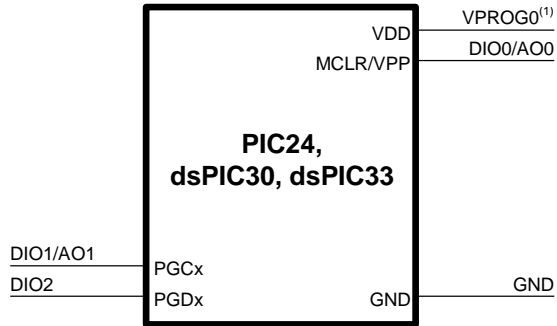
## 10.35 Microchip PIC24, dsPIC30, dsPIC33

### 10.35.1 Specific Commands

Table 42: Microchip PIC24, dsPIC30 and dsPIC33 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR SCLK <frequency Hz>	<b>Sets the serial clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VPP <voltage mV>	<b>Sets the programming voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash or EEPROM memory</b>
TPCMD ERASE F E <tgt row start addr> <len>	<b>Erases Flash or EEPROM memory</b>
TPCMD MASSERASE C	<b>Mass erases Flash, EEPROM and configuration words</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs Flash memory (including configuration words and user ID locations) or EEPROM memory</b>
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (including configuration words and user ID locations) or EEPROM memory</b>
TPCMD READ F E <tgt start addr> <len>	<b>Reads len words starting from tgt start addr from Flash (F) or EEPROM (E) memory</b>
TPCMD RUN	<b>Runs target application</b>

### 10.35.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 75: Example of Connection to a PIC24 Target Device

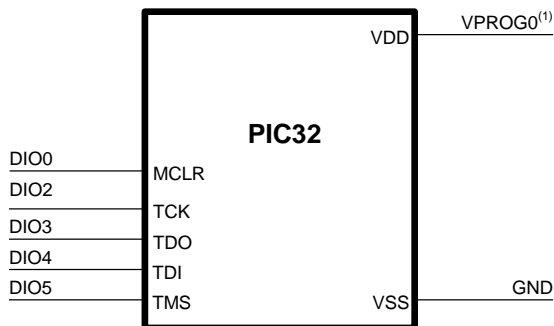
## 10.36 Microchip PIC32

### 10.36.1 Specific Commands

Table 43: Microchip PIC32 Specific Commands

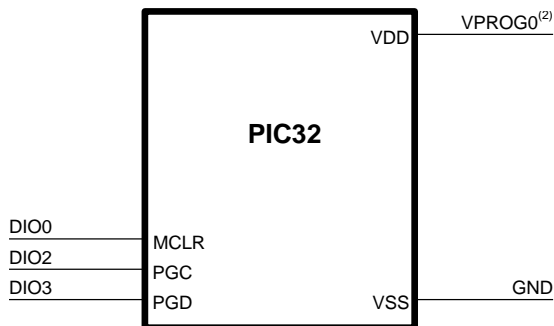
Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV).
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).
TCSETPAR PWUP <time ms>	Sets the power up time (ms).
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms).
<b>TCSETPAR RSTUP &lt;time μs&gt;</b>	Sets the reset up time (us).
<b>TCSETPAR RSTDOWN &lt;time μs&gt;</b>	Sets the reset down time (us).
TCSETPAR CMODE JTAG ICSP	Sets the communication protocol.
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Erases Flash (F) memory.
TPCMD ERASE F <tgt start addr> <len>	Erases Flash (F) memory with page granularity.
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash (F) memory.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F) memory.
TPCMD VERIFY F R C <src offset> <tgt start addr> <len>	Verifies Flash (F) memory. Reads back all written data (R, slow but secure) or compares a checksum (C, fast but not secure).
TPCMD READ F <tgt start addr> <len>	Reads Flash (F) memory.
TPCMD RUN	Executes the firmware programmed in the target device memory.
TPCMD SETJTCLK <value Hz>	Sets communication frequency.

### 10.36.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 76: Example of Connection to a Microchip PIC32 Target Device using EJTAG protocol.



<sup>(2)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 77: Example of Connection to a Microchip PIC32 Target Device using ICSP protocol.



## 10.37 Nec 78K

### 10.37.1 Specific Commands

Table 44: NEC 78K Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE CSI UART	<b>Sets the communication mode</b>
TCSETPAR BAUDRATE 76800 38400 31250 19200 9600	<b>Sets the baud rate (bps) for dual voltage devices in UART mode</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz)</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time $\mu$ s>	<b>Sets the Reset up time (<math>\mu</math>s)</b>
TCSETPAR RSTDOWN <time $\mu$ s>	<b>Sets the Reset down time (<math>\mu</math>s)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE	<b>Mass erases Flash memory</b>
TPCMD BLOCKERASE <tgt start addr> <len>	<b>Erases one or more Flash memory blocks</b>
TPCMD PROGRAM <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD PROTECT <security byte>	<b>Writes security flags</b>
TPCMD RUN	<b>Runs target application</b>

### 10.37.2 Connections (78K0R)

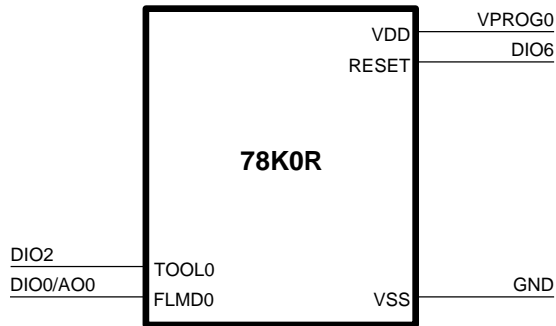


Figure 78: Example of Connection to a 78K0R Target Device

### 10.37.3 Connections (Kx1+ Only)

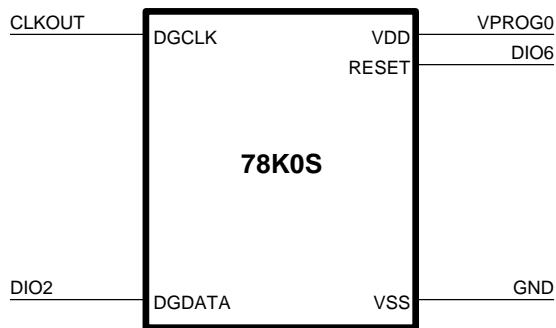
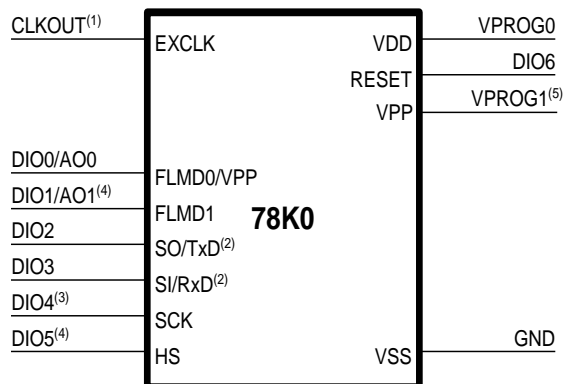


Figure 79: Example of Connection to a 78K0S (Kx1+ Only) Target Device

### 10.37.4 Connections (All Other Devices)



<sup>(1)</sup> Connect this line if you want the target device to be clocked by FlashRunner

<sup>(2)</sup> SI/SO or RxD/TxD depending on the communication mode (CSI or UART)

<sup>(3)</sup> Connect only for CSI communication mode

<sup>(4)</sup> Connect only on some devices (see device-specific script examples)

<sup>(5)</sup> For UPD70 device and DIO1 is Not Used

Figure 80: Example of Connection to a 78K0 Target Device

## 10.38 Nec V850

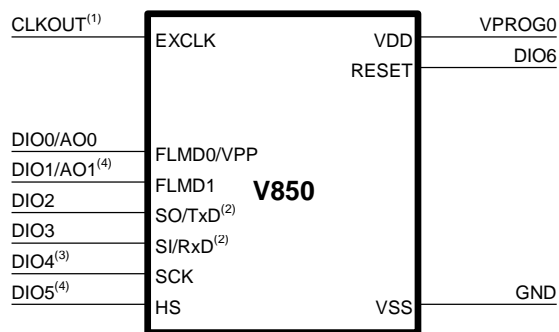
### 10.38.1 Specific Commands

Table 45: NEC V850 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE CSI CSIHS <pulse num>	<p>Sets the communication mode and, in the case more than 1 port available on the MCU, also the specific port:</p> <p>Pulse num. = 8 CSI mode on port 0 (CSIB0)                      Pulse num. = 9 CSI mode on port 3 (CSIB3)                      Pulse num. = 11 CSIHS mode on port 0 (CSIB0 + HS)                      Pulse num. = 12 CSIHS mode on port 3 (CSIB3 + HS)</p>
TCSETPAR CMODE UART	Sets the communication in UART mode
TCSETPAR BAUDRATE <baudrate bps>	Sets the baud rate (bps)
TCSETPAR SCLK <frequency Hz>	Sets the communication serial clock frequency (Hz)
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks Code (F) or Data (E) Flash memory
TPCMD MASSErase F E C	Mass erases Code Flash memory (F), Data Flash memory (E), or both (C)
TPCMD BLOCKERASE F E <tgt start addr> <len>	Erases one or more Code or Data Flash memory blocks
TPCMD PROGRAM	Programs Code (F) or Data (E) Flash memory. Programs by block (filling unused locations with \$FF) or by byte (max 256).
TPCMD VERIFY	Verifies Code (F) or Data (E) Flash memory. Reads back all written data (R, slow but secure), compares a checksum (S, fast but not secure), or performs an internal check (I, fast but not secure). The R option is available for all devices, while the S and I options are device-specific.

Command Syntax	Description
TPCMD PROTECT TPCMD PROTECT TPCMD PROTECT TPCMD PROTECT	Protects the device (syntax varies depending on the specific device)
TPCMD RUN	Runs target application

## 10.38.2 Connections



- (1) Connect this line if you want the target device to be clocked by FlashRunner  
 (2) SI/SO or RxD/TxD depending on the communication mode (CSI or UART)  
 (3) Connect only for CSI communication mode  
 (4) Connect only on some devices (see device-specific script examples)

Figure 81: Example of Connection to a V850 Target Device

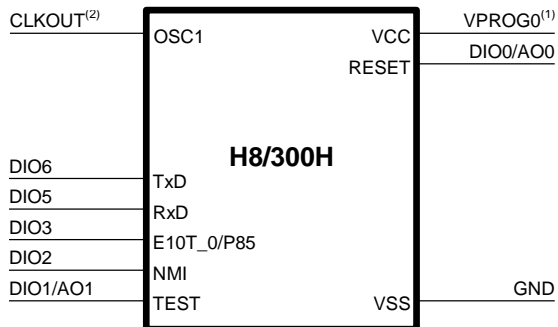
## 10.39 Renesas H8/300H

### 10.39.1 Specific Commands

Table 46: Renesas H8/300H Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE UART	<b>Sets the communication mode</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the communication baud rate (bps)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

## 10.39.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 82: Example of Connection to a H8/300H Target Device

## 10.40 Renesas H8S/H8SX/SuperH (Serial Boot Mode)

### 10.40.1 Specific Commands

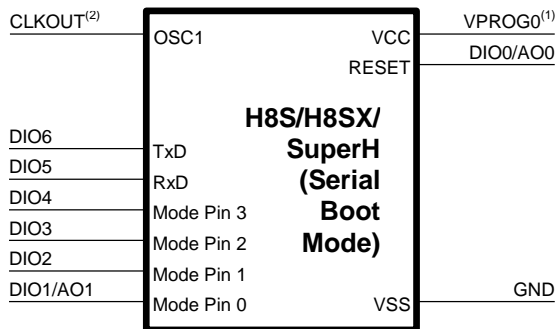
Table 47: Renesas H8S/H8SX/SuperH Specific Commands (Serial Boot Mode)

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE UART	Sets the communication mode
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR BAUDRATE <baudrate bps>	Sets the communication baud rate (bps)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR CLKOUT 5000000 12500000 6250000 3125000  562500 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator. 2 1
TCSETPAR CK_MODE <mode>	Clock mode
TCSETPAR CKM <multiplier>	Main clock multiplier
TCSETPAR CKP <multiplier>	Peripheral clock multiplier
TCSETPAR DIO1 DIO2 DIO3 DIO4 0 1	Specifies FlashRunner DIOx lines to set the device's mode pins to Serial Boot Mode
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD BLOCKERASE F <tgt start addr> <len>	Erases one or more Flash memory blocks
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	Verifies Flash memory. Reads back all written data (R, slow) or compares a checksum (S, fast).
TPCMD READ F <tgt start addr> <len>	Reads Flash memory
TPCMD RUN	Runs target application

10

### 10.40.2 Connections





<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 83: Example of Connection to a H8S/H8SX/SuperH Device (Serial Boot Mode)

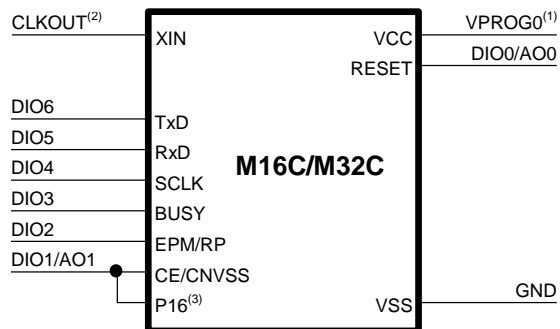
## 10.41 Renesas M16C/M32C

### 10.41.1 Specific Commands

Table 48: Renesas M16C/M32C Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE USART	<b>Sets the communication mode</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR ID_CODE CONST <id1> ... <id7> TCSETPAR ID_CODE SRC <src offset>	<b>Specifies the ID Code necessary for performing programming operations. The ID Code can be passed to FlashRunner directly or read from the source specified by the TPSETSRC command.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory (User or Data)</b>
TPCMD MASSERASE F	<b>Mass erases all non-locked User and Data Flash memory blocks</b>
TPCMD BLOCKERASE F <tgt start addr> <len>	<b>Erases one or more non-locked User or Data Flash memory blocks</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory (User or Data)</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (User or Data)</b>
TPCMD LOCK <tgt start addr> <len>	<b>Locks one or more User Flash memory blocks</b>
TPCMD UNLOCK	<b>Unlocks all locked memory blocks</b>
TPCMD RUN	<b>Runs target application</b>

## 10.41.2 Connections



- (1) Connect this line if you want FlashRunner to automatically power the target device.  
(2) Connect this line if you want the target device to be clocked by FlashRunner  
(3) Connect this line only if available on the target device

Figure 84: Example of Connection to a M16C/M32C Target Device

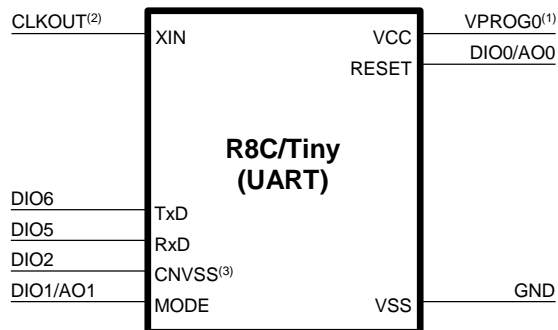
## 10.42 Renesas R8C/Tiny

### 10.42.1 Specific Commands

Table 49: Renesas R8C/Tiny Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE USART UART SINGLE_WIRE	<b>Sets the communication mode</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz) for the USART communication mode</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the baud rate (bps) for UART and SINGLE_WIRE communication modes</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDELAY <time ms>	<b>Reset delay time (ms) which extends both Reset up and Reset down time. Change this value as needed.</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR ID_CODE CONST <id1> ... <id7> TCSETPAR ID_CODE SRC <src offset>	<b>Specifies the ID Code necessary for performing programming operations. The ID Code can be passed to FlashRunner directly or read from the source specified by the TPSETSRC command.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory (User or Data)</b>
TPCMD MASSErase F	<b>Mass erases all non-locked User and Data Flash memory blocks</b>
TPCMD BLOCKERASE F <tgt start addr> <len>	<b>Erases one or more non-locked User or Data Flash memory blocks</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory (User or Data)</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (User or Data)</b>
TPCMD RUN	<b>Runs target application</b>

## 10.42.2 Connections

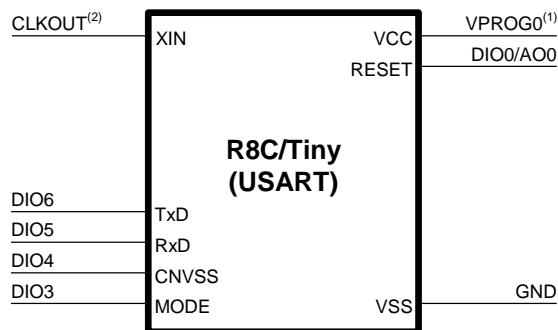


(1) Connect this line if you want FlashRunner to automatically power the target device.

(2) Connect this line if you want the target device to be clocked by FlashRunner

(3) Connect this line only if available on the target device

Figure 85: Example of Connection to a R8C/Tiny Target Device (UART Comm. Mode)



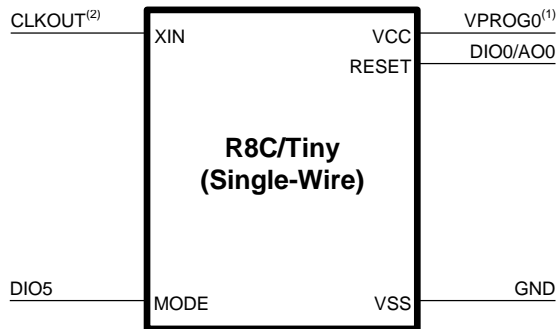
(1) Connect this line if you want FlashRunner to automatically power the target device.

(2) Connect this line if you want the target device to be clocked by FlashRunner

Figure 86: Example of Connection to a R8C/Tiny Target Device (USART Comm. Mode)

## Target-Specific Commands and Connections

---



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 87: Example of Connection to a R8C/Tiny Target Device (Single-Wire Comm. Mode)

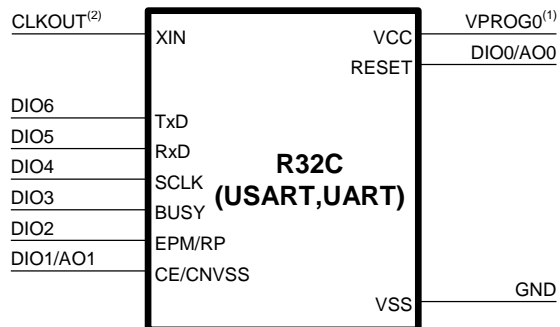
## 10.43 Renesas R32C

### 10.43.1 Specific Commands

Table 50: Renesas R32C Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE USART UART	<b>Sets the communication mode</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDELAY <time ms>	<b>Reset delay time (ms) which extends both Reset up and Reset down time. Change this value as needed.</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR ID_CODE CONST <id1> ... <id7> TCSETPAR ID_CODE SRC <src offset>	<b>Specifies the ID Code necessary for performing programming operations. The ID Code can be passed to FlashRunner directly or read from the source specified by the TPSETSRC command.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash memory (User or Data) or EEPROM memory (Data)</b>
TPCMD MASSERASE F E	<b>Mass erases all non-locked User and Data Flash memory blocks or EEPROM memory (Data)</b>
TPCMD BLOCKERASE F E <tgt start addr> <len>	<b>Erases one or more non-locked User or Data Flash memory blocks or EEPROM memory (Data)</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs Flash memory (User or Data) or EEPROM memory (Data)</b>
TPCMD VERIFY F E R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (User or Data) or EEPROM memory (Data)</b>
TPCMD LOCK <tgt start addr> <len>	<b>Locks one or more User Flash memory blocks</b>
TPCMD UNLOCK	<b>Unlocks all locked memory blocks</b>
TPCMD RUN	<b>Runs target application</b>

### 10.43.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner



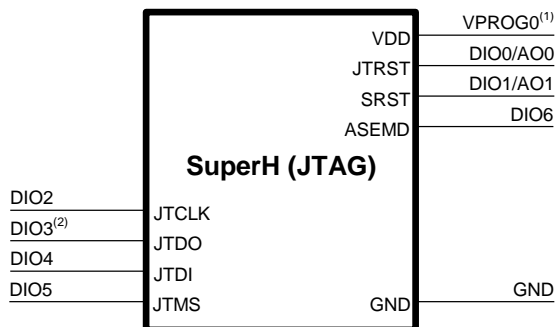
## 10.44 Renesas SuperH (JTAG Interface)

### 10.44.1 Specific Commands

Table 51: Renesas SuperH Specific Commands (JTAG Interface)

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR FOSC <frequency Hz>	Specifies the oscillator frequency (Hz)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR KEY_CODE CONST <key1> ... <key8> TCSETPAR KEY_CODE SRC <src offset>	Specifies the Key Code necessary for performing programming operations. The Key Code can be passed to FlashRunner directly or read from the source specified by the <code>TPSETSRC</code> command.
<b>TPCMD Command</b>	
TPCMD SETJCLK <frequency Hz>	Sets the JTAG clock frequency (Hz)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD ERASE F <tgt start addr> <len>	Erases one or more Flash sectors
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	Verifies Flash memory. Reads back all written data (R, slow) or compares a checksum (S, fast).
<b>TPCMD VERIFY IDCODE</b>	Verifies that the device code (IDCODE) is that of the device specified by the <code>TCSETDEV</code> command
TPCMD READ F <tgt start addr> <len>	Reads Flash memory
TPCMD READ IDCODE	Reads the device code
TPCMD RUN	Runs target application

### 10.44.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> An external pull-up resistor is recommended when communicating at high frequencies

Figure 88: Example of Connection to a SuperH Target Device (JTAG Interface)

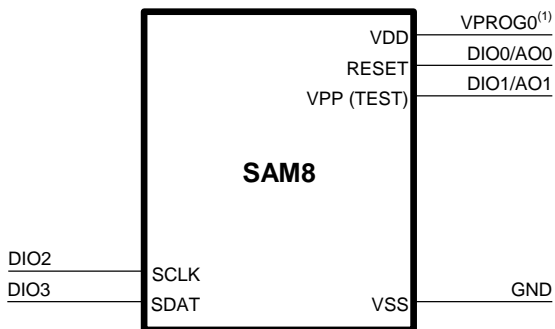
## 10.45 Samsung SAM8

### 10.45.1 Specific Commands

Table 52: Samsung SAM8 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR WRITE_CLOCK <frequency Hz>	Sets the serial clock frequency for write operations (Hz)
TCSETPAR READ_CLOCK <frequency Hz>	Sets the serial clock frequency for read operations (Hz)
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F S <tgt start addr> <len>	Blank checks Flash memory (F) or Smart Option (S)
TPCMD MASSERASE C	Mass erases memory (Flash, Smart Option and protection registers)
TPCMD PROGRAM  S <src offset> <tgt start addr> <len>	Programs Flash memory (F) or Smart Option (S)
TPCMD PROGRAM  H L <src offset> <tgt start addr> len>	Programs protection registers: R ReadProtection (R), HardLock (H) or LDCProtection (L)
TPCMD VERIFY L R H <src offset> <tgt start addr> len>	F Verifies Flash memory using the specified V <sub>TH</sub> level (R: 50% V <sub>TH</sub> , L: 25% V <sub>TH</sub> , H: 75% V <sub>TH</sub> )
TPCMD VERIFY  R H L R <src offset> <tgt start addr> len>	Verifies Smart Option (S), ReadProtection (R), S HardLock (H) or LDCProtection (L) registers
TPCMD READ F S <tgt start addr> <len>	Reads Flash memory (F) or Smart Option (S)
TPCMD READ R H L <tgt start addr> <len>	Reads protection registers: ReadProtection (R), HardLock (H) or LDCProtection (L)
TPCMD RUN	Runs target application

### 10.45.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 89: Example of Connection to a SAM8 Target Device

## 10.46 Serial Memories (SPI Interface)

### 10.46.1 Specific Commands

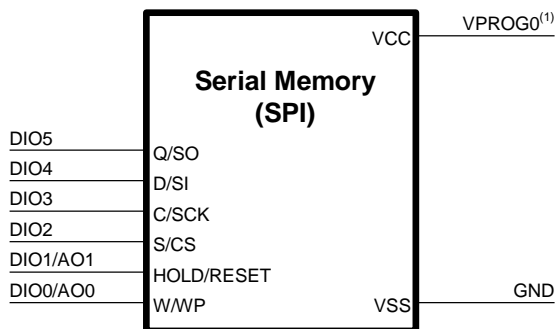
Table 53: Serial Memories (SPI Interface) Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV).</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms).</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms).</b>
<b>TCSETPAR PAGE_MODE STD PO2</b>	<b>Sets the Page size mode – Special Addressing of AT45 Data Flash command.</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the serial clock frequency.</b>
TCSETPAR SKIP_IDCODE_CHECK <value>	<b>Sets the option to check the ID code.</b>
<b>TPCMD Command</b>	
TPCMD CHIPERASE	Mass erases memory <b>(only for devices supporting mass erasing)</b>
<b>TPCMD SECTOR_ERASE F E &lt;tgt start addr&gt; &lt;len&gt;</b>	Erases a sector (Flash or EEPROM) <b>(only for devices supporting sector erasing)</b>
<b>TPCMD BLOCKERASE F E &lt;tgt start addr&gt; &lt;len&gt;</b>	Erases a Flash or EEPROM block <b>(only for devices supporting block erasing)</b>
<b>TPCMD ERASE SP MAIN IPAGE</b>	Erases the Section Protection register <b>(only for devices with this register); Main erase and IPAGE erase for Marvell devices.</b>
TPCMD PAGE_ERASE F E <tgt start addr> <len>	Erases a page of Flash or EEPROM <b>(only for devices supporting page erasing)</b>
TPCMD SUB_SECTOR_ERASE F E <tgt start addr> <len>	Erase a sub-sector of Flash or EEPROM <b>(only for devices supporting sub sector erasing)</b>
<b>TPCMD BLANKCHECK F E X P I &lt;tgt start addr&gt; &lt;len&gt;</b>	Blank checks memory (Flash, EEPROM, Security ID, Info Page for Nordic Semiconductor devices and Information Memory for Marvell devices)
<b>TPCMD PROGRAM F E X P I &lt;src offset&gt; &lt;tgt start addr&gt; &lt;len&gt;</b>	Programs memory (Flash, EEPROM, Security ID, Info Page for Nordic Semiconductor devices and Information Memory for Marvell devices)
<b>TPCMD PROGRAM S &lt;status byte&gt;</b>	Programs the Status register <b>(only for devices with this register)</b>
<b>TPCMD PROGRAM SP &lt;src offset&gt; &lt;len&gt;</b>	Programs the Section Protection register <b>(only for devices with this register)</b>
<b>TPCMD PROGRAM SR &lt;src offset&gt; &lt;len&gt;</b>	Programs the Security register <b>(only for devices with Security register)</b>
<b>TPCMD PROGRAM CONFIG1 CONFIG2 CONFIG3 CONFIG4 &lt;data&gt;</b>	Programs Special Register available on some Marvell devices
<b>TPCMD PROGRAM L &lt;tgt start addr&gt; &lt;data&gt;</b>	Programs the Lock Register

## Target-Specific Commands and Connections

	(only for devices with this register)
TPCMD VERIFY F E X P I R <src offset> <tgt start addr> <len>	Verifies memory (Flash, EEPROM, Security ID, Info Page for Nordic Semiconductor devices and Information Memory for Marvell devices)
TPCMD VERIFY S <status byte>	Verifies the Status register (only for devices with this register)
TPCMD VERIFY SP <src offset> <len>	Verifies the Section Protection register (only for devices with this register)
TPCMD VERIFY SR <src offset> <len>	Verifies the Security register (only for devices with this register)
TPCMD VERIFY CONFIG1 CONFIG2 CONFIG3 CONFIG4 <data>	Verifies Special Register available on some Marvell devices
TPCMD VERIFY L <tgt start addr> <data>	Verifies the Lock Register (only for devices with this register)
TPCMD READ F E X P I <tgt start addr> <len>	Reads memory (Flash, EEPROM, Security ID, Info Page for Nordic Semiconductor devices and Information Memory for Marvell devices)
TPCMD READ CONFIG1 CONFIG2 CONFIG3 CONFIG4 IDCODE	Reads Special Register available on some Marvell devices or the ID code.
TPCMD READ S SP SR SLD	Reads the Status register (S), Section Protection register (SP), Security register (SR) or Sector Lock Down register (SLD) (only for devices with these registers)
TPCMD READ L <tgt start addr>	Reads the Lock Register (only for devices with this register)
TPCMD ENABLE SP	Enables the Section Protection register (only for devices with this register)
TPCMD ENABLE SLD <addr>	Enables the Sector Lock Down register (SLD) (only for devices with this register)
TPCMD DISABLE SP	Disables the Section Protection register (only for devices with this register)
TPCMD PROTECT F <tgt start addr> <len>	Protects Flash
TPCMD PROTECT C	Protects Chip for Nordic Semiconductor Devices
TPCMD UNPROTECT F <tgt start addr> <len>	Unprotects Flash
TPCMD SER_NUMBERING F <tgt start addr> <len> <data_0> <data_len-1>	Programs the values specified (e.g.: for serial numbering purposes)
TPCMD LOCK X	Locks the Security ID (only for devices with Security ID)
TPCMD DUMP F E <dst start addr> <tgt start addr> <len>	Creates a binary dump file with the Flash or EEPROM Memory content

## 10.46.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 90: Example of Connection to a Serial Memory (SPI Interface)

## 10.47 Serial Memories (Microwire Interface)

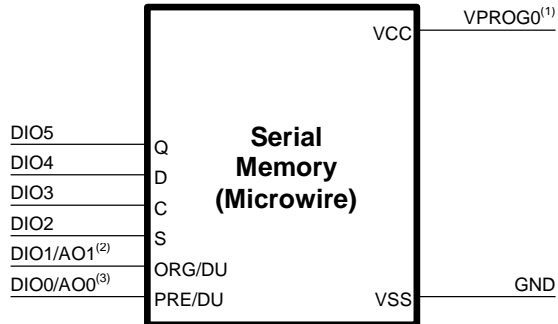
### 10.47.1 Specific Commands

Table 54: Serial Memories (Microwire Interface) Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz)</b>
TCSETPAR DATA_WIDTH 8 16	<b>Sets the memory data width (8 or 16 bits)</b>
TCSETPAR PAGEMODE STD PO2	<b>Standard (factory value) or Power of 2 (one-time programmable) page size</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks memory (Flash or EEPROM)</b>
TPCMD CHIP_ERASE	<b>Mass erases memory (only for devices supporting mass erasing)</b>
TPCMD SECTOR_ERASE <tgt start addr> <len>	<b>Erases a sector (only for devices supporting sector erasing)</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs memory (Flash or EEPROM)</b>
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	<b>Verifies memory (Flash or EEPROM)</b>
TPCMD READ F E <tgt start addr> <len>	<b>Reads memory (Flash or EEPROM)</b>



## 10.47.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Connect this line only on target devices with an ORG pin  
<sup>(3)</sup> Connect this line only on target devices with a PRE pin

Figure 91: Example of Connection to a Serial Memory (Microwire Interface)

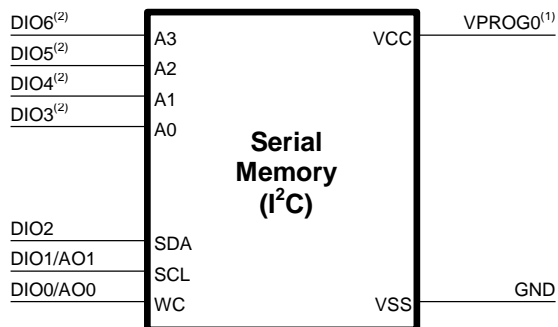
## 10.48 Serial Memories (I<sup>2</sup>C Interface)

### 10.48.1 Specific Commands

Table 55: Serial Memories (I<sup>2</sup>C Interface) Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz)</b>
TCSETPAR DEVICE_ADDR <address>	<b>Selects the target device at the specified address (\$00 to \$0F)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks memory (Flash or EEPROM)</b>
TPCMD CHIP_ERASE	<b>Mass erases memory (only for devices supporting mass erasing)</b>
TPCMD SECTOR_ERASE <tgt start addr> <len>	<b>Erases a sector (only for devices supporting sector erasing)</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs memory (Flash or EEPROM)</b>
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	<b>Verifies memory (Flash or EEPROM)</b>
TPCMD READ F E <tgt start addr> <len>	<b>Reads memory (Flash or EEPROM)</b>

## 10.48.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Connect these lines only if necessary

Figure 92: Example of Connection to a Serial Memory (I<sup>2</sup>C Interface)

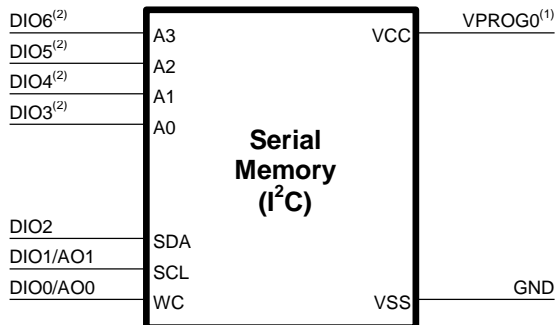
## 10.49 Serial Memories (I2C Interface) SERMEM\_B Algorithm

### 10.49.1 Specific Commands

Table 56: Serial Memories (I2C Interface) Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV).
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).
TCSETPAR PWUP <time ms>	Sets the power up time (ms).
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms).
TCSETPAR SCLK <frequency Hz>	Sets the serial clock frequency.
TCSETPAR SKIP_IDCODE_CHECK <value>	Sets the option to check the ID code.
TCSETPAR DEVICE_ADDR <address>	Selects the target device at the specified address (\$00 to \$0F)
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E IDP <tgt start addr> <len>	Blank checks memory (Flash, EEPROM or ID page)
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs memory (Flash or EEPROM)
TPCMD VERIFY F E R <src offset> <tgt start addr> <len>	Verifies memory (Flash or EEPROM)
TPCMD READ F E IDP <tgt start addr> <len>	Reads memory (Flash, EEPROM or ID page)

## 10.49.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Connect these lines only if necessary

Figure 93: Example of Connection to a Serial Memory Sermem\_B Algorithm (I<sup>2</sup>C Interface)

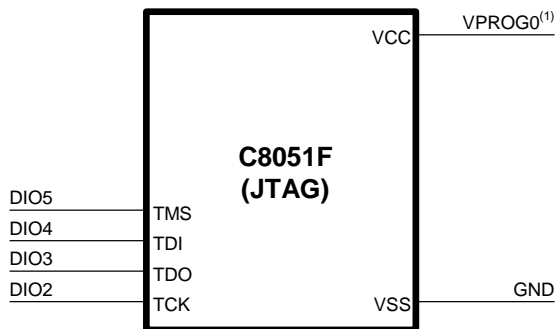
## 10.50 Silicon Labs C8051F

### 10.50.1 Specific Commands

Table 57: Silicon Labs C8051F Specific Commands

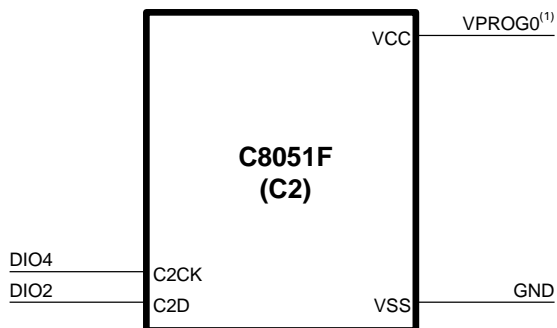
Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FCPU <frequency Hz>	<b>Sets the CPU frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR CMODE JTAG C2	<b>Sets the communication mode</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (only for the JTAG communication mode)</b>
TCSETPAR C2CLK <frequency Hz>	<b>Sets the C2 clock frequency (only for the C2 communication mode)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE	<b>Mass erases the whole memory</b>
TPCMD ERASE F E <tgt start addr> <len>	<b>Erases Flash (F) or EEPROM (E) memory</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs Flash (F) or EEPROM (E) memory</b>
TPCMD VERIFY F E R <src offset> <tgt start addr> <len>	<b>Verifies Flash (F) or EEPROM (E) memory</b>
TPCMD READ F E <src offset> <tgt start addr> <len>	<b>Reads len bytes from Flash (F) or EEPROM (E) memory starting from tgt start addr location</b>

## 10.50.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 94: Example of Connection to a C8051F Target Device (JTAG Interface)



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 95: Example of Connection to a C8051F Target Device (C2 Interface)

## 10.51 STMicroelectronics ST10

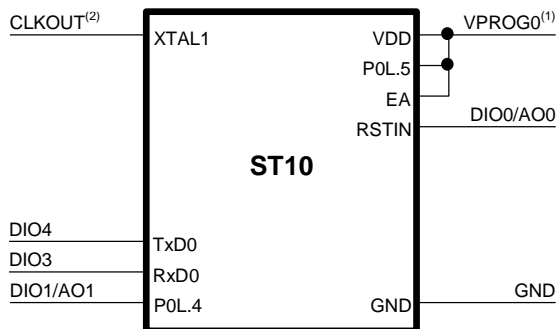
### 10.51.1 Specific Commands

Table 58: STMicroelectronics ST10 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FCPU <frequency Hz>	<b>Sets the CPU frequency (Hz)</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the communication baud rate (bit/s)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSErase F	<b>Mass erases Flash memory</b>
TPCMD SECTORERASE F <tgt page addr>	<b>Erases one Flash page</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory</b>
TPCMD RUN	<b>Runs target application</b>



## 10.51.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device. Lines POL.5 and EA must always be connected to VDD.

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 96: Example of Connection to a ST10 Target Device

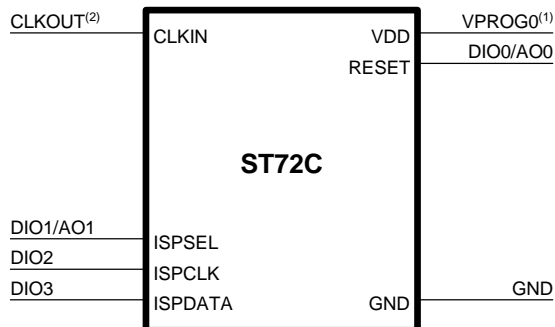
## 10.52 STMicroelectronics ST72C

### 10.52.1 Specific Commands

Table 59: STMicroelectronics ST72C Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD UNPROTECT	<b>Unprotects protected devices</b>
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks memory (Flash or EEPROM)</b>
TPCMD MASSErase F E	<b>Mass erases memory (Flash or EEPROM)</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs memory (Flash or EEPROM)</b>
TPCMD PROGRAM OPTIONS <B0> [<B1>]	<b>Programs Option Byte(s)</b>
TPCMD READ OPTIONS	<b>Reads Option Byte(s)</b>
TPCMD VERIFY F E R <src offset> <tgt start addr> <len>	<b>Verifies memory (Flash or EEPROM)</b>
TPCMD VERIFY OPTIONS <B0> [<B1>]	<b>Verifies Option Byte(s)</b>
TPCMD RUN	<b>Runs target application</b>

## 10.52.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner

Figure 97: Example of Connection to an ST72C Target Device

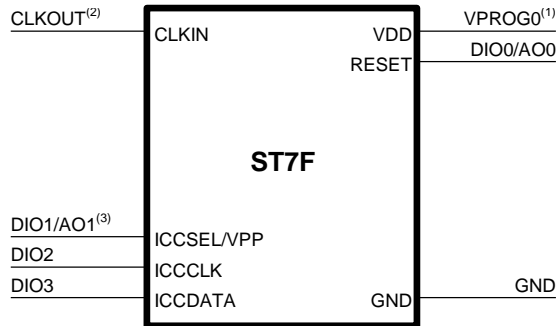
## 10.53 STMicroelectronics ST7F

### 10.53.1 Specific Commands

Table 60: STMicroelectronics ST7F Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR ICPMODE ENABLED DISABLED	<p>Sets whether Option Bytes are used when entering the ICP mode.</p> <p>ENABLED: the ST7 takes into account the Option Bytes configuration. In this case, make sure that the selected clock source is available in the application.</p> <p>DISABLED: the ST7 doesn't take into account the Option Bytes configuration and an external clock has to be provided.</p>
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR CLKOUT 25000000 12500000 6250000 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD UNPROTECT	Unprotects protected devices
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks memory (Flash or EEPROM)
TPCMD MASSErase F E	Mass erases memory (Flash or EEPROM)
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs memory (Flash or EEPROM)
TPCMD PROGRAM OPTIONS <B0> [<B1>]	Programs Option Byte(s)
TPCMD READ M F E <tgt start addr> <len>	M: Memory (RAM, Registers e Reserved Area); F: Flash; E: EEPROM
TPCMD READ OPTIONS	Reads Option Byte(s)
TPCMD VERIFY F E R <src offset> <tgt start addr> <len>	Verifies memory (Flash or EEPROM)
TPCMD VERIFY OPTIONS <B0> [<B1>]	Verifies Option Byte(s)
TPCMD RESTORE_RC_CALIBRATION	Restores RC calibration bytes
TPCMD TRIM equency Hz> <tgt start addr> lerance_pct>	<p>Trims internal oscillator to a given precision. Calculates the trimming value for the specified frequency, and programs it at the specified address. Only works with TCSETPAR ICPMODE ENABLED.</p>
TPCMD RUN	Runs target application

## 10.53.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner  
<sup>(3)</sup> Connect this line only if the target device has the ICCSEL/VPP pin

Figure 98: Example of Connection to an ST7F Target Device

## 10.54 STMicroelectronics STM32

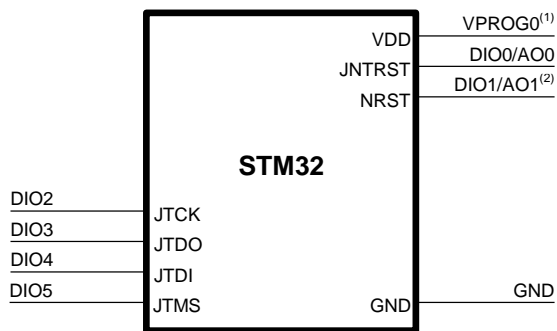
### 10.54.1 Specific Commands

Table 61: STM32 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSH/PULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR RSTTYPE SW HW	<b>Sets the Reset type (software or hardware)</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR FCPU <frequency Hz>	<b>Specifies the CPU clock frequency (Hz)</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F O	<b>Mass erases Flash memory (F) or Option Bytes (O)</b>
TPCMD ERASE F <tgt start addr> <len>	<b>Erases Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD PROGRAM <src offset> <tgt start addr> <len>  W	<ul style="list-style-type: none"> <li>O <b>Programs Option Bytes. Uses byte (B) or word (W) addressing mode, depending on how the Option Bytes were generated by the compiler.</b></li> <li>B</li> </ul>
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Reads back all written data (R, slow) or compares a checksum (S, fast).</b>
TPCMD VERIFY R S <src offset> <tgt start addr> <len>  W	<ul style="list-style-type: none"> <li>O <b>Verifies Option Bytes memory. Reads back all written data (R, slow) or compares a checksum (S, fast). Uses byte (B) or word (W) addressing mode.</b></li> <li>B</li> </ul>
TPCMD RUN	<b>Runs target application</b>

10

### 10.54.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Optional

Figure 99: Example of Connection to a STM32 Target Device

## 10.55 STMicroelectronics STM8

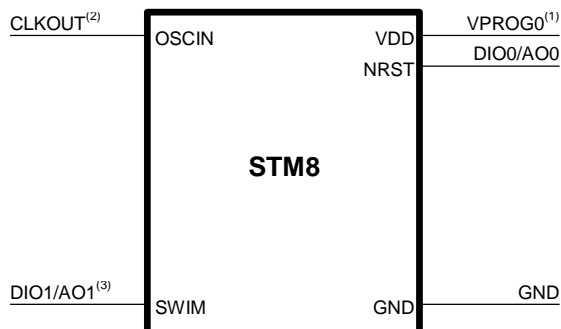
### 10.55.1 Specific Commands

Table 62: STMicroelectronics STM8 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT 25000000 12500000 6250000 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD CONNECT	<b>Powers on the board and enters the programming mode</b>
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash (F) EEPROM (E) memory</b>
TPCMD MASSErase F E	<b>Mass erases Flash (F) EEPROM (E) memory</b>
TPCMD ERASE F E <tgt start addr> <len>	<b>Erases Flash (F) EEPROM (E) memory</b>
TPCMD PROGRAM	<b>Programs Flash (F), EEPROM (E) or Option Bytes (O) memory</b>
TPCMD VERIFY	<b>Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory</b>
TPCMD READ F E <tgt start addr> <len>	<b>Reads len bytes starting from tgt start addr from Flash (F) EEPROM (E) memory</b>
TPCMD DISCONNECT	<b>Exits the programming mode and powers off the board</b>
TPCMD RUN	<b>Runs target application</b>
TPCMD SWIM_RST	<b>Executes a reset of the MCU leaving it in OFF mode</b>



## 10.55.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Connect this line if you want the target device to be clocked by FlashRunner  
<sup>(3)</sup> An external pull-up resistor (~300Ω) is required on this line. Pull-up resistor value may be different, compatibly with voltage and current specifications of SWIM pin.

Figure 100: Example of Connection to a STM8 Target Device

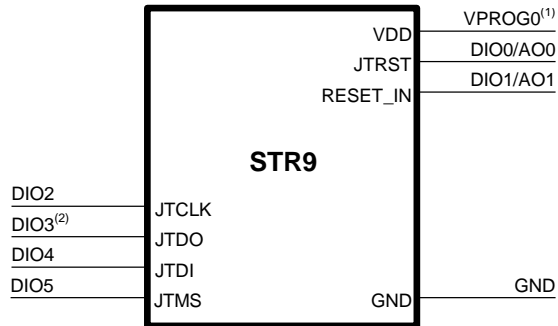
## 10.56 STMicroelectronics STR9

### 10.56.1 Specific Commands

Table 63: STMicroelectronics STR9 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDELAY <time μs>	<b>Sets the Reset delay time (μs)</b>
<b>TPCMD Command</b>	
TPCMD SETJTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F USER CONF	<b>Mass erases Flash memory, User Code locations (USER) or Configuration locations (CONF)</b>
TPCMD SECTORERASE F <sect start addr>	<b>Erases a Flash sector</b>
TPCMD PROGRAM	<b>Programs Flash memory (F), One-Time Programmable memory (OTP), User Code locations (USER) or Configuration locations (CONF)</b>
TPCMD VERIFY	<b>Verifies Flash memory (F), One-Time Programmable memory (OTP), User Code locations (USER) or Configuration locations (CONF)</b>
TPCMD LOCK OTP	<b>Locks OTP memory (writes the OTP lock bit)</b>
TPCMD SECURE	<b>Secures the device memory (writes the security bit)</b>
TPCMD UNSECURE	<b>Unsecures the device memory (erases Flash, User Code locations and Configuration locations)</b>
TPCMD READ F OTP USER CONF <tgt start addr> <len>	<b>Reads len bytes from the specified memory (Flash, OTP, User Code locations or Configuration locations) starting from tgt start addr location</b>
TPCMD WRITE16 <tgt addr> <data>	<b>Writes a 16-bit data to a specific non-Flash address</b>
TPCMD READ16 <tgt addr>	<b>Reads a 16-bit data from a specific address</b>
TPCMD WRITE32 <tgt addr> <data>	<b>Writes a 32-bit data to a specific non-Flash address</b>
TPCMD READ32 <tgt addr>	<b>Reads a 32-bit data from a specific address</b>
TPCMD RUN	<b>Runs target application</b>

## 10.56.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> An external pull-up resistor is recommended when communicating at high frequencies

Figure 101: Example of Connection to an STR9 Target Device

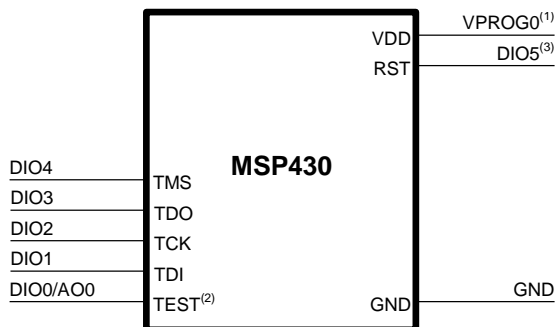
## 10.57 Texas Instruments MSP430/SC4C12

### 10.57.1 Specific Commands

Table 64: Texas Instruments MSP430/SC4C12 Specific Commands

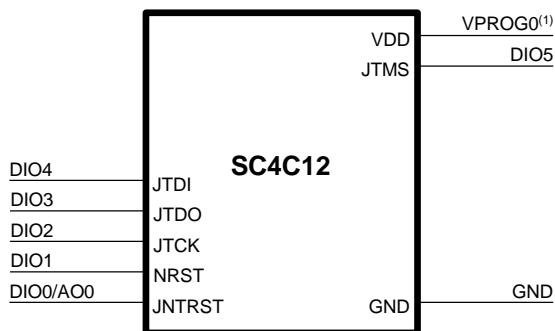
Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE MAIN ALL ALL_PRESERVE_CALIBRATION	<b>Mass erases memory main memory (MAIN), main and info memories (ALL) or main and info memories without erasing the calibration data (ALL_PRESERVE_CALIBRATION)</b>
TPCMD SEGMENTERASE F <sect start addr>	<b>Erases a Flash segment</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R C <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Reads back all written data (R, slow but secure) or compares a CRC (C, fast but not secure).</b>
TPCMD WRITE8 <tgt addr> <data>	<b>Writes a 8-bit data to a specific non-Flash address</b>
TPCMD READ8 <tgt addr>	<b>Reads a 8-bit data from a specific address</b>
TPCMD WRITE16 <tgt addr> <data>	<b>Writes a 16-bit data to a specific non-Flash address</b>
TPCMD READ16 <tgt addr>	<b>Reads a 16-bit data from a specific address</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len bytes starting from tgt start addr from Flash (F)</b>
TPCMD BLOWFUSE	<b>Blows protection fuse</b>
TPCMD RUN	<b>Runs target application</b>

## 10.57.2 Connections JTAG



- (1) Connect this line if you want FlashRunner to automatically power the target device  
 (2) Connect this line only on devices with the TEST signal  
 (3) Connect this line only on devices supporting the SBW interface

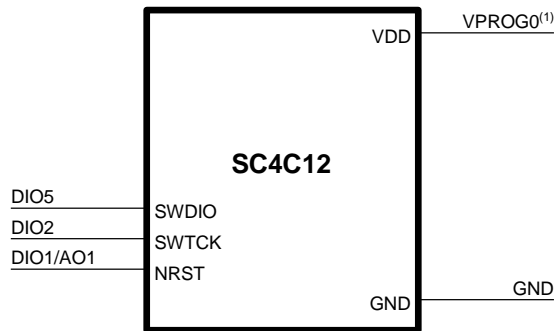
Figure 102: Example of Connection to an MSP430 Target Device



- (1) Connect this line if you want FlashRunner to automatically power the target device

Figure 103: Example of Connection to an SC4C12 Target Device

### 10.57.3 Connections SWD



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 104: Example of Connection to an SC4C12 Target Device

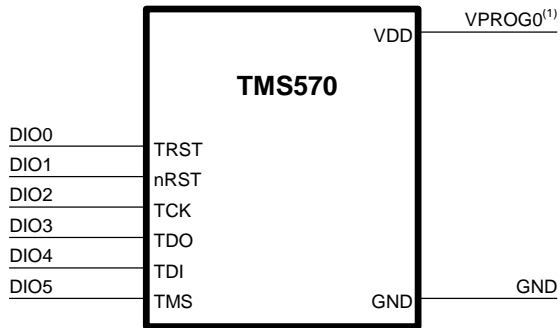
## 10.58 Texas Instruments TMS570

### 10.58.1 Specific Commands

Table 65: Texas Instruments TMS570 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets the JTAG clock frequency (Hz)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR OSCIN <frequency MHz>	<b>Input to the PLL machine, that will produce a 140 Mhz clock according to the oscillator input</b>
TCSETPAR RSTTYPE HW (optional)	<b>This command enables FlashRunner to use RST line to drive the target device</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash or EEPROM memory</b>
TPCMD MASSERASE F E C	<b>Compact and Erase flash memory F, EEPROM memory E, or both (if EEPROM available) C</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len> <ECC>	<b>Programs len byte locations of flash memory starting from the tgt start addr address. Add ECC parameter to automatically write ECC memory.</b>
TPCMD VERIFY F E R S <src offset> <tgt start addr> <len>	<b>Verifies len byte locations of flash memory starting from the tgt start addr address using ReadOut (R) or CheckSum (S) method (please see AN for more information).</b>
TPCMD READ F E <tgt start addr> <len>	<b>Reads len locations of flash or ECC flash memory starting from the tgt start addr address</b>
TPCMD RUN	<b>Runs target application</b>

### 10.58.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 105: Example of Connection to a TMS570 Target Device



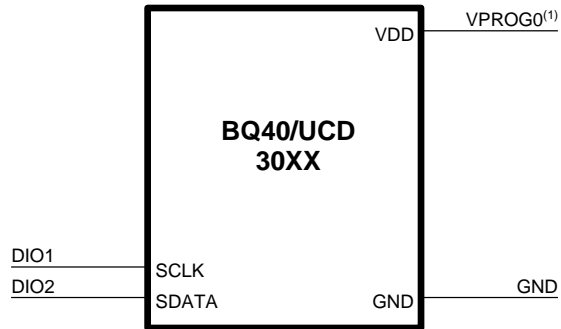
## 10.59 TI BQ40/UCD30XX

### 10.59.1 Specific Commands

Table 66: TI UCD30XX Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR SCLK <frequency Hz>	<b>This command is used to set up the communication frequency between FlashRunner and target microcontroller. Must be less than 400 KHz.</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blankchecks Program flash or Code flash memory. Blankchecks len locations starting from the address specified by tgt start address</b>
TPCMD MASSERASE F	<b>'C' parameter erase Program flash (F) and Code flash (E). 'F' erases Program flash, 'E' erases Code flash</b>
TPCMD ERASE F E <tgt start addr> <len>	<b>'F' parameter erases Program flash, 'E' parameter erases Code flash. Please note that number of erased locations will be approximated to the last flash block indicated by tgt start addr + len formula</b>
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	<b>Programs len locations of Program flash memory or Code flash memory starting from the tgt start addr address. Last 4 locations are dedicated to firmware checksum, if checksum is not correct, device will remain in ROM mode. If the checksum is correct but firmware doesn't run, device will be lost after resetting the device.</b>
TPCMD VERIFY F E R S <src offset> <tgt start addr> <len>	<b>Verifies len locations of Program flash or Code flash memory starting from the tgt start addr address.</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len locations of Program flash or Code flash memory starting from the tgt start addr address.</b>
TPCMD RUN	<b>Runs target application</b>
TPCMD ROMBACK <command> <delay> <device_id> <pw_length> <pw>	<b>This command is used to go back to ROM mode when a customer firmware is running on the device. Please note that command, device_ID code as well as pw are user definable. Moreover pw is optional and if not present, could be ignored putting '0' value to pw_length.</b>

## 10.59.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 106: Example of Connection to BQ40/UCD30XX Target Device

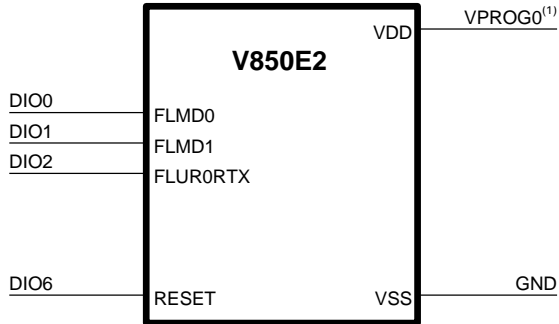
## 10.60 Renesas V850E2

### 10.60.1 Specific Commands

Table 67: Renesas V850E2 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	time μs: Reset rising time, expressed in microseconds.
TCSETPAR RSTDOWN <time μs>	time μs: Reset falling time, expressed in microseconds.
TCSETPAR RSTDRV <mode>	The Reset Drive Mode can be OPENDRAIN or PUSH/PULL.
TCSETPAR FOSC <frequency Hz>	This command is used to set up the frequency of the external oscillator mounted on the target board.
TCSETPAR CMODE <mode>	Select the protocol to use for communication with the LSI.
TCSETPAR BAUDRATE <value>	This command is used to set up the serial baudrate.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blankchecks Flash memory or EEPROM memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD MASSERASE C F E	It erases all the device (C), Flash memory (F) or EEPROM memory (E).
TPCMD ERASE F E <tgt start addr> <len>	It erases blocks of locations in the Flash memory (F) or in the EEPROM memory (E). tgt start address is the start address of a block. len is such as the final address of the range to be erased is exactly the final address of a memory block.
TPCMD PROGRAM F E O D <src offset> <tgt start addr> <len> ID_TAG	It programs len locations of Flash memory or EEPROM memory starting from the tgt start addr address.
TPCMD VERIFY F E R <src offset> <tgt start addr> <len>	It verifies len locations of Flash memory or EEPROM memory starting from the tgt start addr address. len specifies the number of locations to be verified.
TPCMD PROTECT <security flag> <FSW start block> <FSW end block>	It sets the security flag and the boot swap location.
TPCMD READ F <tgt start addr> <len>	Reads len locations of Program flash or Code flash memory starting from the tgt start addr address.
TPCMD RUN	Runs target application

### 10.60.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 107: Example of Connection to a V850E2 Target Device

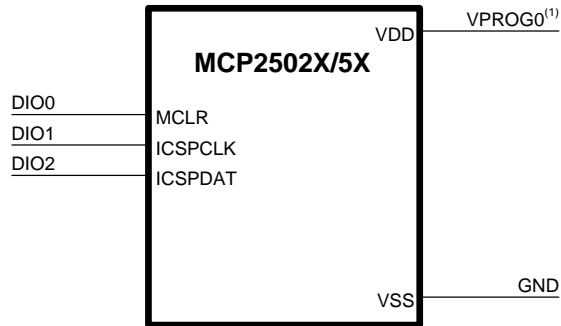
## 10.61 Microchip MCP2502X/5X

### 10.61.1 Specific Commands

Table 68: Microchip MCP2502X/5X Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR SCLK <clock>	<b>Serial clock frequency, Hz</b>
TCSETPAR VPP <value>	<b>Programming Voltage value, mV</b>
<b>TPCMD Command</b>	
TPCMD PROGRAM F <src offset> <tgt start addr> <len> ID_TAG	<b>Programs Eprom Memory / Configuration Word</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Eprom Memory</b>

## 10.61.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 108: Example of Connection to a Microchip MCP2502X/5X Target Device



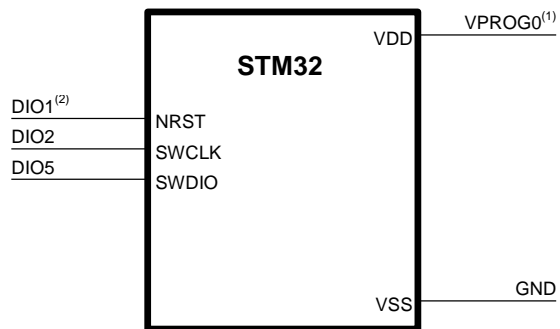
## 10.62 STMicroelectronics STM32 (Serial Wire Debug)

### 10.62.1 Specific Commands

Table 69: STM32 SWD Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX < voltage mV >	<b>Sets the VDD Auxiliary voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time us >	<b>Sets reset rising up time (us)</b>
TCSETPAR RSTDOWN <time us >	<b>Sets reset falling up time (us)</b>
TCSETPAR RSTDRV <mode>	<b>Set the Reset drive mode</b>
TCSETPAR RSTTYPE <mode>	<b>Set the Reset type mode</b>
TCSETPAR SWCLK <clock>	<b>Sets Serial wire debug clock frequency, Hz</b>
TCSETPAR FCPU <Hz>	<b>Sets CPYU core frequency, Hz</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE F O	<b>Issues masserase of Flash memory (F) or Option bytes (O)</b>
TPCMD ERASE F <tgt start addr> <len>	<b>Erase len locations starting from the address specified by tgt start address.</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.</b>
TPCMD PROGRAM F O <src offset> <tgt start addr> <len> B W	<b>Programs len locations of Flash memory (F) or option byte (O) starting from the tgt start addr address. B W to be used only with option bytes (Byte or word storage format in source file)</b>
TPCMD VERIFY F O R S <src offset> <tgt start addr> <len> B W	<b>Verifies len locations of Flash memory (F) or Option Byte (O) memory with Readout @ or checksum (S) method starting from the tgt start addr address. len specifies the number of locations to be verified. B W to be used only with option bytes (Byte or word storage format in source file)</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len locations of Flash memory starting from the tgt start addr address. len specifies the number of locations to be read.</b>
TPCMD RUN	<b>Runs the target application</b>

### 10.62.2 Connections



- <sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device  
<sup>(2)</sup> Optional

Figure 109: Example of Connection to a STM32 Cortex Target Device MCU through SWD



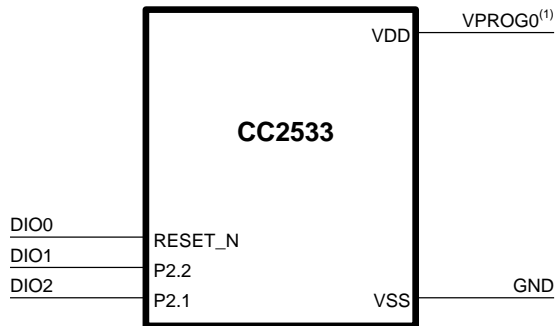
## 10.63 Texas Instruments CC2533

### 10.63.1 Specific Commands

Table 70: Texas Instruments CC2533 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV).
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).
TCSETPAR PWUP <time ms>	Sets the power up time (ms).
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms).
<b>TCSETPAR RSTUP &lt;time μs&gt;</b>	Sets the reset up time (us).
<b>TCSETPAR RSTDOWN &lt;time μs&gt;</b>	Sets the reset down time (us).
TCSETPAR SCLK <frequency Hz>	Sets the serial clock frequency.
<b>TPCMD Command</b>	
TPCMD MASSERASE C	Erases Flash memory.
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory (E).
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory (E).
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	Verifies Flash memory (E) . Reads back all written data (E).
TPCMD READ F <tgt start addr> <data>	Reads Flash memory (E).
<b>TPCMD UNLOCK</b>	
TPCMD RUN	Executes the firmware programmed in the target device memory.

### 10.63.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 110: Example of Connection to a Texas Instruments CC2533 Target Device

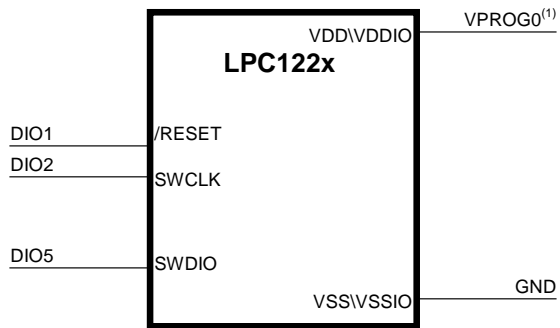
## 10.64 NXP LPC122x (Serial Wire Debug)

### 10.64.1 Specific Commands

Table 71: NXP LPC122x SWD Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time $\mu$ s >	<b>Sets reset rising up time (us)</b>
TCSETPAR RSTDOWN <time $\mu$ s >	<b>Sets reset falling up time (us)</b>
TCSETPAR RSTDRV <mode>	<b>Set the Reset drive mode</b>
TCSETPAR RSTTYPE <mode>	<b>Set the Reset type mode</b>
TCSETPAR SWCLK <clock>	<b>Sets Serial wire debug clock frequency, Hz</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE F	<b>Issues masserase of Flash memory (F)</b>
TPCMD ERASE F <tgt start addr> <len>	<b>Erase len locations starting from the address specified by tgt start address.</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.</b>
TPCMD SECTOR_BLANKCHECK F <tgt start addr> <len>	<b>It blankchecks the content of a single sector or multiple sectors.</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len> B W	<b>Programs len locations of Flash memory starting from the tgt start addr address</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies len locations of Flash memory with Read out method (R) starting from the tgt start addr address. len specifies the number of locations to be verified</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads len locations of Flash memory starting from the tgt start addr address. len specifies the number of locations to be read.</b>
TPCMD RUN	<b>Runs the target application</b>

### 10.64.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 111: Example of Connection to a LPC 122x Target Device through SWD

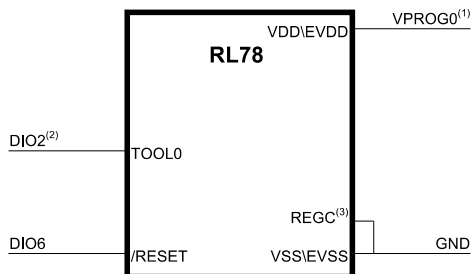
## 10.65 Renesas RL78 Devices

### 10.65.1 Specific Commands

Table 72: Renesas RL78 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time $\mu$ s >	<b>Sets reset rising up time (us)</b>
TCSETPAR RSTDOWN <time $\mu$ s >	<b>Sets reset falling up time (us)</b>
TCSETPAR RSTDRV <mode>	<b>Set the Reset drive mode</b>
TCSETPAR RSTTYPE <mode>	<b>Set the Reset type mode</b>
<b>TCSETPAR CMODE &lt;communication mode&gt;</b>	<b>select the flash programming serial protocol UART1 or UART2</b>
<b>TCSETPAR FOSC &lt;frequency Hz&gt;</b>	<b>set up the frequency (Hertz) of the external oscillator mounted on the target board.</b>
<b>TCSETPAR BAUDRATE &lt;bps&gt;</b>	<b>set up the serial communication bit/rate. (bit/s). Possible bps values are: 1000000,500000, 250000 or 115200 bps</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE C D	<b>Issues masserase of Code Flash memory (C) or Data Flash memory (D)</b>
TPCMD BLOCKERASE C D <tgt start addr> <len>	<b>Erase len locations of Code Flash memory (C) or Data Flash memory (D) starting from the address specified by tgt start address.</b>
TPCMD BLANKCHECK C D <tgt start addr> <len>	<b>Blankchecks Code Flash memory (C) or Data Flash memory (D). Blankchecks len locations starting from the address specified by tgt start address.</b>
TPCMD PROGRAM C D <src offset> <tgt start addr> <len>	<b>Programs len locations of Code Flash memory (C) or Data Flash memory (D) starting from the tgt start addr address</b>
TPCMD VERIFY C D <src offset> <tgt start addr> <len>	<b>Verifies len locations of Code Flash memory (C) or Data Flash memory (D) with Read out method starting from the tgt start addr address. len specifies the number of locations to be verified</b>
<b>TPCMD RUN</b>	<b>Runs the target application</b>

## 10.65.2 Connections for UART Single Wire (1Wire) e UART 2Wire

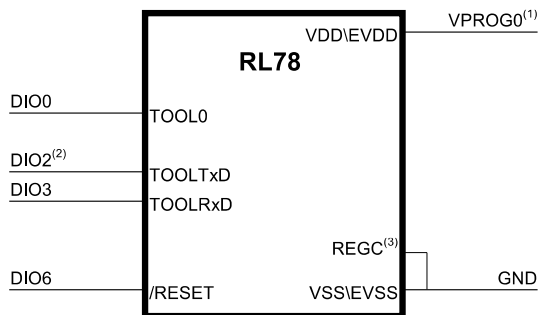


(1) Connect this line if you want FlashRunner to automatically power the target device

(2) An external pullup resistor ( $1K\Omega$ ) is required on this line

(3) Connect the REGC pin to VSS via a capacitor (0.47 to 1  $\mu$ F).

Figure 112: Example of Connection to RL78 Device UART Single-Wire



(1) Connect this line if you want FlashRunner to automatically power the target device

(2) An external pullup resistor ( $1K\Omega$ ) is required on this line

(3) Connect the REGC pin to VSS via a capacitor (0.47 to 1  $\mu$ F).

Figure 113: Example of Connection to RL78 Device UART 2Wires

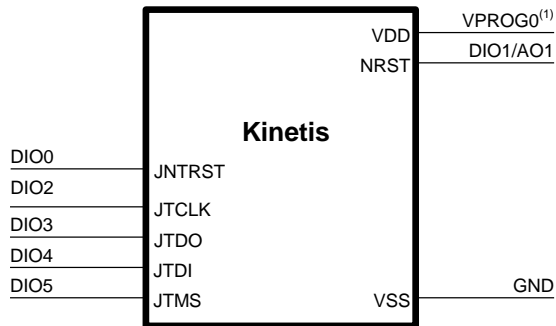
## 10.66 Freescale Kinetis

### 10.66.1 Specific Commands

Table 73: Freescale Kinetis Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV).
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).
TCSETPAR CLOCKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms).
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms).
TCSETPAR RSTUP <time μs>	Sets the reset up time (us).
TCSETPAR RSTDOWN <time μs>	Sets the reset down time (us).
TCSETPAR RSTTYPE SW HW	Sets the reset type.
TCSETPAR RSTDRV OPENDRAIN PUSHPULL	Sets the reset drive mode.
TCSETPAR CMODE JTAG SWD	Sets the communication protocol.
TCSETPAR JTCLK <frequency Hz>	Sets the serial clock frequency.
<b>TPCMD Command</b>	
TPCMD MASSERASE F C E	Erases Flash (F), the whole device (C) or the Flex memory (E).
TPCMD ERASE F E <tgt start addr> <len>	Erases Flash (F) or the Flex memory (E).
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks Flash (F) or the Flex memory (E).
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs Flash (F) or the Flex memory (E).
TPCMD VERIFY F E R S <src offset> <tgt start addr> <len>	Verifies Flash (F) or the Flex memory (E). Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).
TPCMD READ F E <tgt start addr> <len>	Reads Flash (F) or the Flex memory (E).
TPCMD RUN	Executes the firmware programmed in the target device memory.
TPCMD WRITE F E <tgt start addr> <len>	Writes data to memory map

### 10.66.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 114: Example of Connection to a Freescale Kinetis Target Device



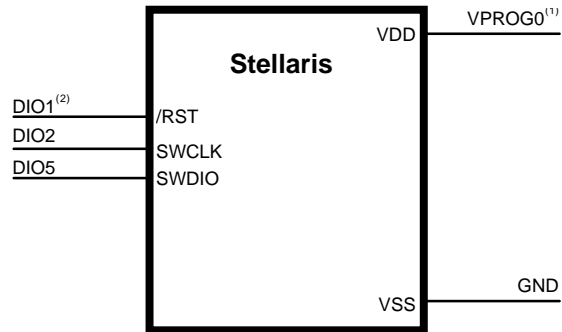
## 10.67 Texas Instruments Stellaris (Serial Wire Debug)

### 10.67.1 Specific Commands

Table 74: TI Stellaris SWD Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX < voltage mV >	Sets the VDD Auxiliary voltage (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time us >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time us >	Sets reset falling up time (us)
TCSETPAR RSTDRV <mode>	Set the Reset drive mode
TCSETPAR SWCLK <clock>	Sets Serial wire debug clock frequency, Hz
TCSETPAR FCPU <Hz>	Sets CPYU core frequency, Hz
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD ERASE F <tgt start addr> <len>	Erase <i>len</i> locations starting from the address specified by <i>tgt start address</i> .
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks <i>len</i> locations starting from the address specified by <i>tgt start address</i> .
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs <i>len</i> locations of Flash memory (F) starting from the <i>tgt start addr</i> address.
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	Verifies <i>len</i> locations of Flash memory (F) with Readout or checksum (S) method starting from the <i>tgt start addr</i> address. <i>len</i> specifies the number of locations to be verified.
TPCMD READ F <tgt start addr> <len>	Reads <i>len</i> locations of Flash memory starting from the <i>tgt start addr</i> address. <i>len</i> specifies the number of locations to be read.
TPCMD RUN	Runs the target application

### 10.67.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Optional

Figure 115: Example of Connection to a TI Stellaris Target Device MCU through SWD

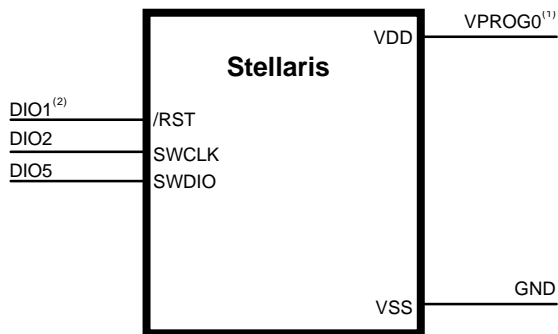
## 10.68 Analog Devices ADP104w

### 10.68.1 Specific Commands

Table 75: Analog devices ADP104w Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR PMB_ADDR <time $\mu$ s >	<b>Sets reset rising up time (us)</b>
TCSETPAR EE_PSW <time $\mu$ s >	<b>Sets reset falling up time (us)</b>
TCSETPAR CHANGE_EE_PSW <mode>	<b>Set the Reset drive mode</b>
TCSETPAR SCLK <clock>	<b>Sets Serial wire debug clock frequency, Hz</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE E	<b>Issues masserase of Flash memory (F)</b>
TPCMD UNLOCK E	<b>Unlocks Eeprom memory (E)</b>
TPCMD BLANKCHECK E <tgt start addr> <len>	<b>Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.</b>
TPCMD PROGRAM E <src offset> <tgt start addr> <len>	<b>Programs len locations of Flash memory (F) starting from the tgt start addr address.</b>
TPCMD VERIFY E R S <src offset> <tgt start addr> <len>	<b>Verifies len locations of Flash memory (F) with Readout or checksum (S) method starting from the tgt start addr address. len specifies the number of locations to be verified.</b>
TPCMD LOCK E	<b>Locks Eeprom memory (E).</b>
TPCMD CHANGE_PSW E	<b>Changes Eeprom memory (E) password.</b>
TPCMD RUN	<b>Runs the target application</b>

### 10.68.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

<sup>(2)</sup> Optional

Figure 116: Example of Connection to Analog devices ADP1048w Target Device MCU through SWD

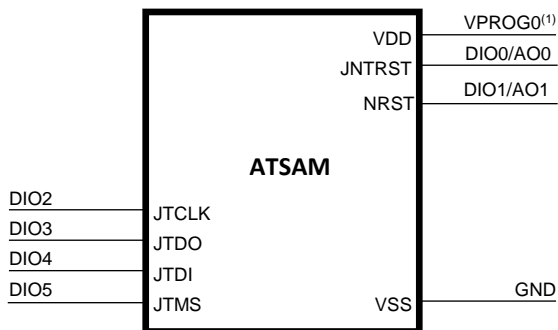
## 10.69 AMTEL ATSAM

### 10.69.1 Specific Commands

Table 76: ATMEL ATSAM via Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX < voltage mV >	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time μs >	Sets reset falling up time (us)
TCSETPAR RSTDRV OPENDRAIN   PUSHPULL	Sets reset driving mode (PUSHPULL or OPENDRAIN)
TCSETPAR CMODE JTAG SWD	Sets communication mode JTAG or SWD
TPCMD JTCLK <frequency Hz>	Sets frequency for the JTAG channel (Hz)
TPCMD SWDCLK <frequency Hz>	Sets frequency for the SWD channel (Hz)
<b>TPCMD Command</b>	
TPCMD MASSERASE F U	Issues masserase of Flash memory (F), User signature (U)
TPCMD BLANKCHECK F U <tgt start addr> <len>	Blankchecks Flash memory, User signature (U). Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F U <src offset> <tgt start addr> <len>	Programs Flash (F) User signature (U).
TPCMD VERIFY F U <src offset> <tgt start addr> <len>	Verifies Flash (F), User signature (U).
TPCMD SETGPNVM <bit number>	Sets GPNVM bit
TPCMD CLEARGPNVM <bit number>	Clears GPNVM bit
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

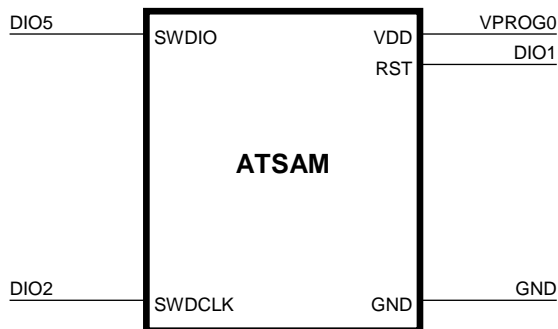
### 10.69.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 117: Example of Connection to ATMEL ATSAM Target Device MCU through JTAG

### 10.69.3 Connections SWD



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 118: Example of Connection to ATMEL ATSAM Target Device MCU through SWD

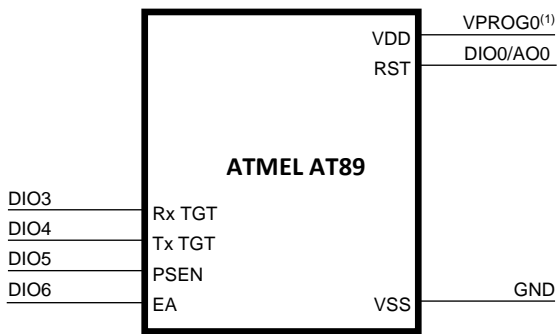
## 10.70 ATMEL AT89/AT90/ATMEGA/ATTINY

### 10.70.1 Specific Commands

Table 77: ATMEL AT89 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR CMODE JTAG SPI UART	Sets communication mode JTAG or SPI or UART
TCSETPAR JTCLK <frequency Hz>	Sets frequency for JTAG channel (Hz)
TCSETPAR BAUDRATE <baudrate>	Sets baudrate in Hz (Baudrate 9600 Only for UART protocol)
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD PROGRAM SSB <security level>	Programs Lock bits via UART
TPCMD READ SSB	Reads Lock bits via UART
TPCMD PROGRAM FLB FHB EFB <values>	Programs Fuse high bits, Fuse low bits or Extended fuse bits
TPCMD VERIFY FLB FHB EFB <values>	Verifies Fuse high bits, Fuse low bits or Extended fuse bits
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

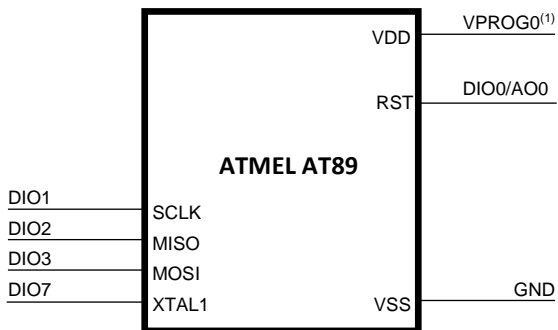
### 10.70.2 Connections UART



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 119: Example of Connection to ATMEEL AT89 Target Device MCU through UART

### 10.70.3 Connections SPI

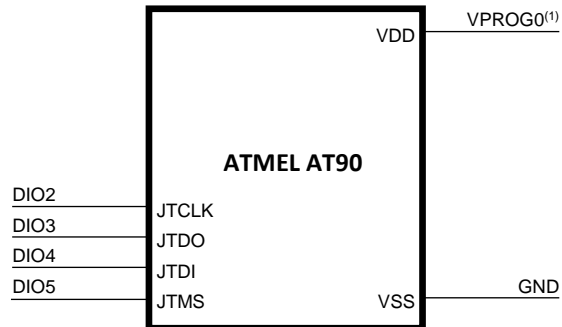


<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 120: Example of Connection to ATMEEL AT89 Target Device MCU through SPI



### 10.70.4 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 121: Example of Connection to ATMEEL AT89 Target Device MCU through JTAG

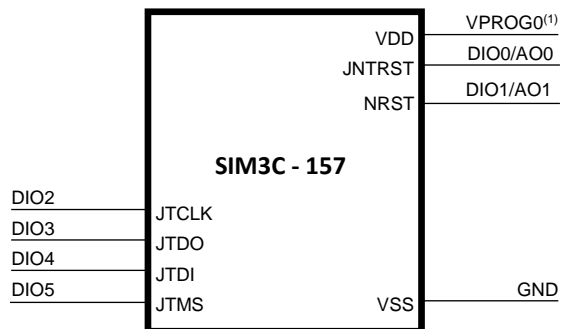
## 10.71 Silicon labs SIM3C-157

### 10.71.1 Specific Commands

Table 78: SIM3C-157 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time $\mu$ s >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time $\mu$ s >	Sets reset falling up time (us)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets reset driving mode (PUSHPULL or OPENDRAIN)
TCSETPAR JTCLK <frequency Hz>	Sets frequency for the JTAG channel (Hz) It must be less than HCLK frequency
TCSETPAR FCPU <frequency Hz>	Sets HCLK frequency i.e., the frequency of the core. Note: use this parameter if you want to speed-up the programming performance, otherwise leaves it commented and the HCLK clock will be equal to the HSI oscillator clock. See in the device documentation the minimum and the maximum value for the HCLK clock.
<b>TPCMD Command</b>	
TPCMD MASSERASE F E	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F E O <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F E O <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

## 10.71.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 122: Example of Connection to SIM3C - 157 Target Device MCU through JTAG

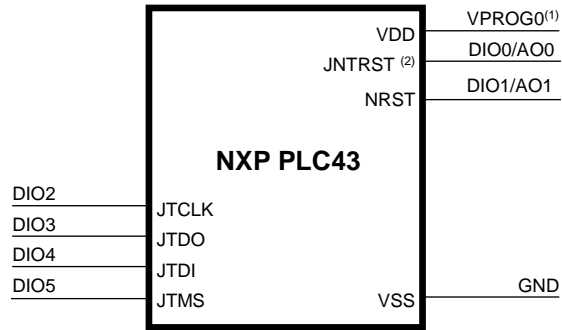
## 10.72 NXP LPC43

### 10.72.1 Specific Commands

Table 79: NXP LPC43 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time μs >	Sets reset falling up time (us)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets reset driving mode (PUSHPULL or OPENDRAIN)
TCSETPAR JTCLK <frequency Hz>	Sets frequency for the JTAG channel (Hz) It must be less than HCLK frequency
TCSETPAR FCPU <frequency Hz>	Sets HCLK frequency i.e., the frequency of the core. Note: use this parameter if you want to speed-up the programming performance, otherwise leaves it commented and the HCLK clock will be equal to the HSI oscillator clock. See in the device documentation the minimum and the maximum value for the HCLK clock.
<b>TPCMD Command</b>	
TPCMD MASSERASE F E	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F E O <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F E O <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

## 10.72.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

<sup>(2)</sup> If present on device JTAG implementation

Figure 123: Example of Connection to NXP LPC43 Target Device MCU through JTAG

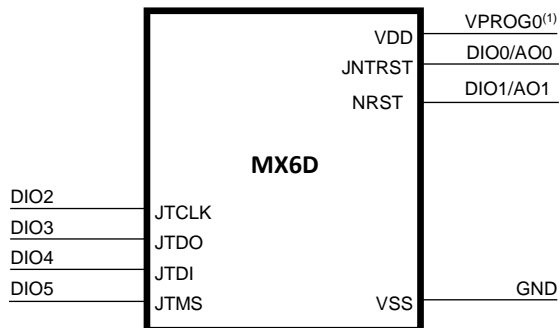
## 10.73 Freescale MX6D

### 10.73.1 Specific Commands

Table 80: Freescale MX6D Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time $\mu$ s >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time $\mu$ s >	Sets reset falling up time (us)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets reset driving mode (PUSHPULL or OPENDRAIN)
TCSETPAR JTCLK <frequency Hz>	Sets frequency for the JTAG channel (Hz)
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD WRITE O <start addr> <len>	Programs OTP fuses
TPCMD READ O <start addr> <len>	Reads OTP fuses
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

## 10.73.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 124: Example of Connection to FREESCALE MX6D Target Device MCU through JTAG

## 10.74 Silicon Labs BGM111

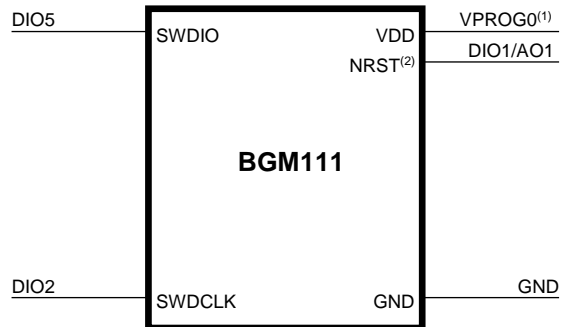
### 10.74.1 Specific Commands

Table 81: BGM111 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time $\mu$ s >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time $\mu$ s >	Sets reset falling up time (us)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets reset driving mode (PUSHPULL or OPENDRAIN)
TCSETPAR SWCLK <frequency Hz>	Sets frequency for the SWD channel (Hz)
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

### 10.74.2 Connections SWD





<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

<sup>(2)</sup> Remember to add the TCSETPAR RSTTYPE HW line if you want to use the NRST line.

Figure 125: Example of Connection to SILABS BGM111 Target Device MCU through SWD

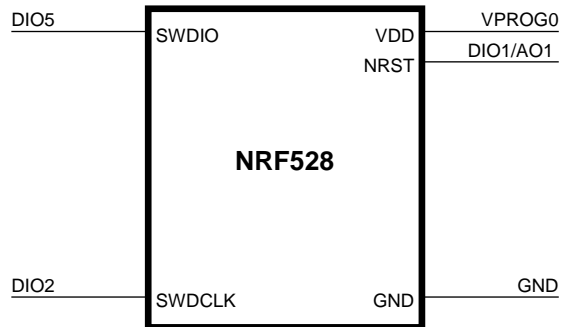
## 10.75 Nordic NRF528

### 10.75.1 Specific Commands

Table 82: Nordic NRF528 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time $\mu$ s >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time $\mu$ s >	Sets reset falling up time (us)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets reset driving mode (PUSHPULL or OPENDRAIN)
TCSETPAR SWCLK <frequency Hz>	Sets frequency for the SWD channel (Hz)
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

### 10.75.2 Connections SWD



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 126: Example of Connection to Nordic NRF 528 Target Device MCU through SWD

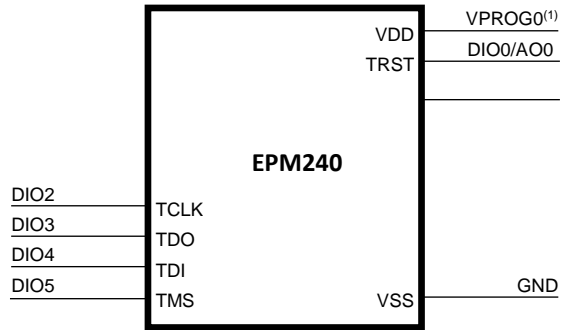
## 10.76 ALTERA EPM240

### 10.76.1 Specific Commands

Table 83: Altera EPM240 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time $\mu$ s >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time $\mu$ s >	Sets reset falling up time (us)
TCSETPAR JTCLK <frequency Hz>	Sets frequency for the JTAG channel (Hz)
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

## 10.76.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 127: Example of Connection to Altera EPM420 CPLD\_A Target Device MCU through JTAG

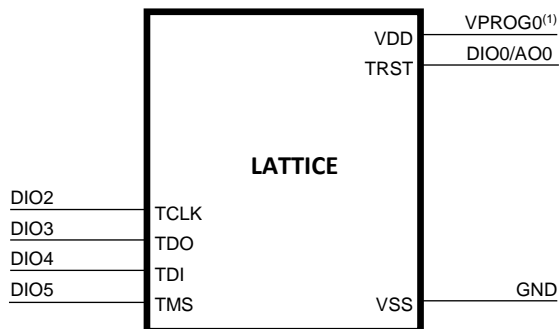
## 10.77 Lattice

### 10.77.1 Specific Commands

Table 84: Lattice Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time $\mu$ s >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time $\mu$ s >	Sets reset falling up time (us)
TCSETPAR JTCLK <frequency Hz>	Sets frequency for the JTAG channel (Hz)
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application

## 10.77.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 128: Example of Connection to LATTICE LA-XP2-5E Target Device MCU through JTAG

## 10.78 Cast R8051XC2

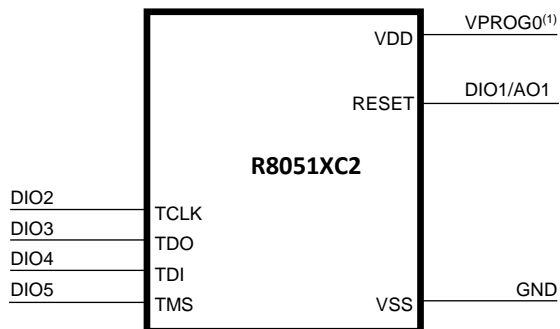
### 10.78.1 Specific Commands

Table 85: Cast R8051XC2 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets the VDD Auxiliary voltage (mV)
TCSETPAR CLKOUT <frequency Hz>	Sets the clock frequency provided by the programmer.
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time $\mu$ s >	Sets reset rising up time (us)
TCSETPAR RSTDOWN <time $\mu$ s >	Sets reset falling up time (us)
TCSETPAR INIT_JTCLK <frequency Hz>	Sets frequency for the JTAG channel during initialization (Hz)
TCSETPAR PROG_JTCLK <frequency Hz>	Sets frequency for the JTAG channel during programming (Hz)
<b>TPCMD Command</b>	
TPCMD MASSERASE F	Issues masserase of Flash memory (F)
TPCMD BLANKCHECK F <tgt start addr> <len>	Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory
TPCMD DISCONNECT	Exits the programming mode and powers off the board
TPCMD RUN	Runs target application



## 10.78.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 129: Example of Connection to CAST R8051XC2 Target Device MCU through JTAG

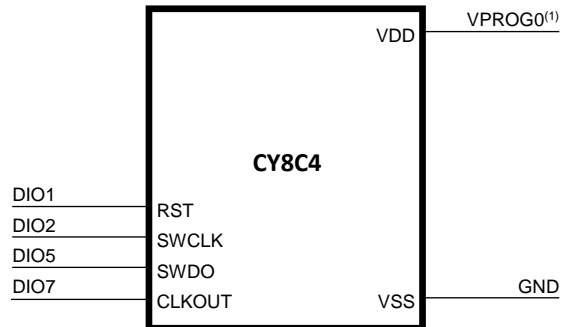
## 10.79 Cypress CY8C4

### 10.79.1 Specific Commands

Table 86: Cypress Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets the VDD Auxiliary voltage (mV)</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Sets the clock frequency provided by the programmer.</b>
TCSETPAR ENTRYMODE RESET POWERCYCLE	<b>Entry mode settings</b>
TCSETPAR SWDCLK <frequency Hz>	<b>Sets frequency for the SWD channel during programming (Hz)</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE F	<b>Issues masserase of Flash memory (F)</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash (F), EEPROM (E) or Option Bytes (O) memory</b>
TPCMD VERIFY F <src offset> <tgt start addr> <len>	<b>Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory</b>
TPCMD PROGRAM PS <start addr>	<b>Programs protection settings</b>
TPCMD VERIFY PS <start addr>	<b>Verifies protection settings</b>
TPCMD DISCONNECT	<b>Exits the programming mode and powers off the board</b>
TPCMD RUN	<b>Runs target application</b>

## 10.79.2 Connections SWD



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 130: Example of Connection to CY8C4 Target Device MCU through SWD

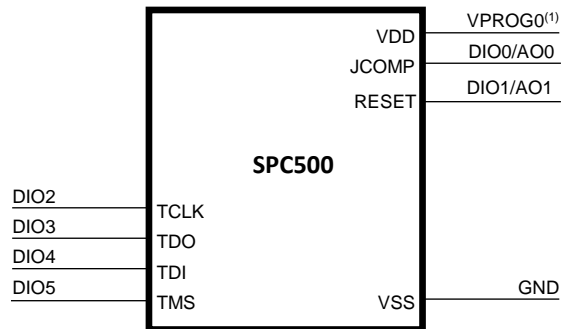
## 10.80 STMicroelectronics SPC500

### 10.80.1 Specific Commands

Table 87: Freescale SPC500 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz)
TCSETPAR JTCLK <frequency Hz>	Sets the JTAG clock frequency
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F S <tgt start addr> <len>	Blank checks Flash (F) or Shadow (S) memory
TPCMD MASSERASE F S	Mass erases Flash (F) or Shadow (S) memory
TPCMD ERASE F S <tgt page addr> <len>	Erases one or more Flash (F) or Shadow (S) pages
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash (F) or Shadow (S) memory
TPCMD VERIFY F <src offset> <tgt start addr> <len>	Verifies Flash (F) or Shadow (S) memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).
TPCMD DUMP F E S <src offset> <tgt start addr> <len>	Creates a binary dump file of the selected memory type

## 10.80.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 131: Example of Connection to a SPC500 Target Device MCU through JTAG

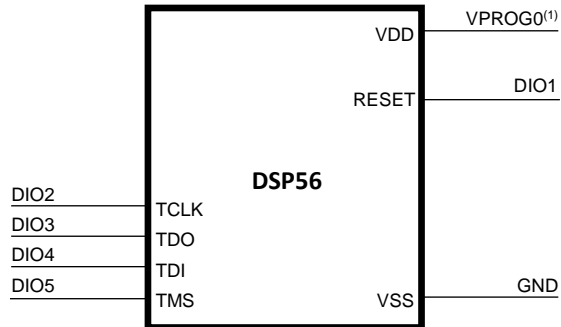
## 10.81 Freescale DSP56

### 10.81.1 Specific Commands

Table 88: Freescale SPC500 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time $\mu$ s >	<b>Sets reset rising up time (us)</b>
TCSETPAR RSTDOWN <time $\mu$ s >	<b>Sets reset falling up time (us)</b>
TCSETPAR INIT_JTCLK <frequency Hz>	<b>Sets frequency for the JTAG channel during initialization (Hz)</b>
TCSETPAR PROG_JTCLK <frequency Hz>	<b>Sets frequency for the JTAG channel during programming (Hz)</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE F D B	<b>Issues masserase of Flash memory (F), Data memory (D), Boot memory (B).</b>
TPCMD BLANKCHECK F D B <tgt start addr> <len>	<b>Blankchecks Flash memory (F), Data memory (D), Boot memory (B). Blankchecks len locations starting from the address specified by tgt start address.</b>
TPCMD PROGRAM F D B <src offset> <tgt start addr> <len>	<b>Programs Flash memory (F), Data memory (D), Boot memory (B).</b>
TPCMD VERIFY F D B <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (F), Data memory (D), Boot memory (B).</b>
TPCMD DISCONNECT	<b>Exits the programming mode and powers off the board</b>
TPCMD RUN	<b>Runs target application</b>

## 10.81.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 132: Example of Connection to a DSP56 Target Device MCU through JTAG

## 10.82 Freescale MKV30

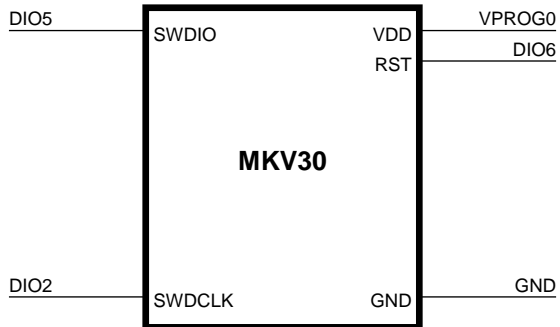
### 10.82.1 Specific Commands

Table 89: Freescale MKV30 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR SWDCLK <frequency Hz>	<b>Sets the SWD clock frequency</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F S <tgt start addr> <len>	<b>Blank checks Flash (F) or Shadow (S) memory</b>
TPCMD MASSERASE F S	<b>Mass erases Flash (F) or Shadow (S) memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash (F) or Shadow (S) memory</b>
TPCMD VERIFY F <src offset> <tgt start addr> <len>	<b>Verifies Flash (F) or Shadow (S) memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).</b>
TPCMD DUMP F E S <src offset> <tgt start addr> <len>	<b>Creates a binary dump file of the selected memory type</b>



## 10.82.2 Connections SWD



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 133: Example of Connection to a MKV30 Target Device MCU through SWD

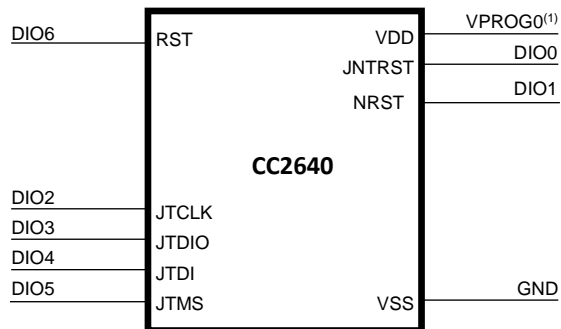
## 10.83 Texas Instruments CC2640

### 10.83.1 Specific Commands

Table 90: Texas Instruments CC2640 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV).</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms).</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms).</b>
TCSETPAR RSTDOWN <time ms>	<b>Sets the reset up time (ms).</b>
TCSETPAR RSTUP <time ms>	<b>Sets the reset down time (ms).</b>
TCSETPAR CMODE <mode>	<b>Sets the communication mode</b>
TCSETPAR JTCLK <frequency Hz>	<b>Sets frequency for the JTAG channel.</b>
<b>TPCMD Command</b>	
TPCMD MASSErase C	<b>Erases Flash memory.</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory (F).</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory (F).</b>
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	<b>Verifies Flash memory (F) . Reads back all written data (R).</b>
TPCMD READ F <tgt start addr> <data>	<b>Reads Flash memory (F).</b>
TPCMD RUN	<b>Executes the firmware programmed in the target device memory.</b>

## 10.83.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 134: Example of Connection to a Texas Instruments CC2640 Target Device MCU through JTAG

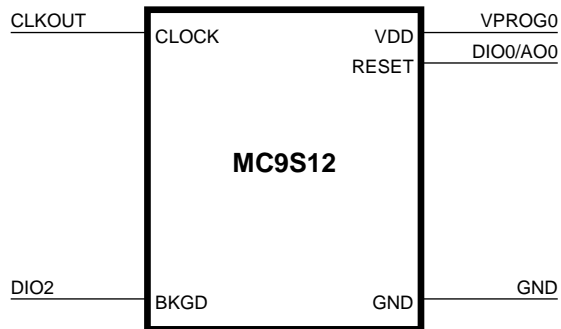
## 10.84 Freescale MC9S12/MC9RS0

### 10.84.1 Specific Commands

Table 91: Texas Instruments CC2640 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV).
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
TCSETPAR PWUP <time ms>	Sets the power up time (ms).
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms).
TCSETPAR RSTDOWN <time ms>	Sets the reset up time (ms).
TCSETPAR RSTUP <time ms>	Sets the reset down time (ms).
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz). A frequency of 0 disables the PLL.
TCSETPAR REFDIV <divisor>	Sets the PLL frequency divisor register
TCSETPAR SYNDIV <divisor>	Sets the PLL frequency divider register
TCSETPAR POSTDIV <divisor>	Sets the PLL frequency divider register
<b>TPCMD Command</b>	
TPCMD MASSERASE C	Erases Flash memory.
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks Flash memory, Eeprom memory.
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs Flash memory, Eeprom memory.
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	Verifies Flash memory, Eeprom memory.
TPCMD RUN	Executes the firmware programmed in the target device memory.

## 10.84.2 Connections BDM



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 135: Example of Connection to Freescale MC9S12 Target Device

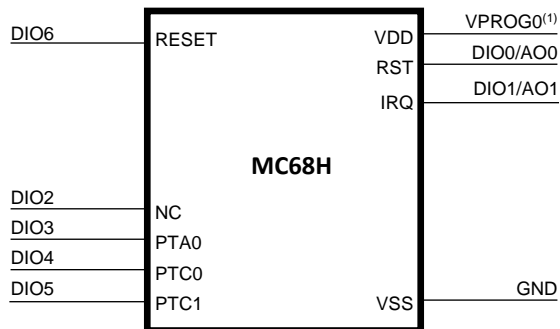
## 10.85 Freescale MC68H

### 10.85.1 Specific Commands

Table 92: Freescale MC68H Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV).
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV).
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.
TCSETPAR PWUP <time ms>	Sets the power up time (ms).
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms).
TCSETPAR RSTDOWN <time ms>	Sets the reset up time (ms).
TCSETPAR RSTUP <time ms>	Sets the reset down time (ms).
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR PLLFREQ <frequency Hz>	Sets the PLL frequency (Hz). A frequency of 0 disables the PLL.
TCSETPAR REFDIV <divisor>	Sets the PLL frequency divisor register
TCSETPAR SYNDIV <divisor>	Sets the PLL frequency divider register
TCSETPAR POSTDIV <divisor>	Sets the PLL frequency divider register
<b>TPCMD Command</b>	
TPCMD MASSERASE C	Erases Flash memory.
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory (F).
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory (F).
TPCMD VERIFY F R <src offset> <tgt start addr> <len>	Verifies Flash memory (F) . Reads back all written data (R).
TPCMD READ F <tgt start addr> <data>	Reads Flash memory (F).
TPCMD RUN	Executes the firmware programmed in the target device memory.

## 10.85.2 Connections JTAG



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 136: Example of Connection to Freescale MC68H Target Device

## 10.86 Infineon TLE9000

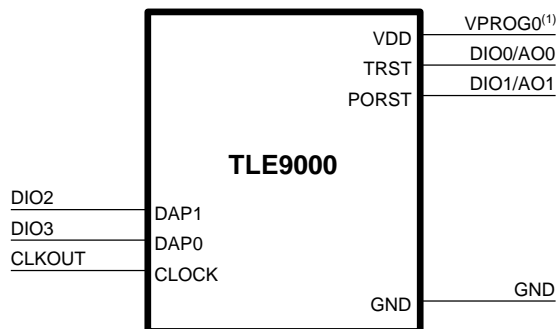
### 10.86.1 Specific Commands

Table 93: Infineon TLE9000 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR SWCLK <frequency Hz>	<b>Sets the SWD clock frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSH_PULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR ENTRY_MODE RESET POR	
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSErase F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Reads back all written data (R, slow but secure) or compares a checksum (S, fast but not secure).</b>
TPCMD PROGRAM N <src offset> <tgt start addr> <len>	<b>Programs NVMCS memory</b>
TPCMD VERIFY N R <src offset> <tgt start addr> <len>	<b>Verifies NVMCS memory</b>
TPCMD RUN	<b>Runs target application</b>



## 10.86.2 Connections SWD



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 137: Example of Connection to TLE9000 Target Device MCU through DAP

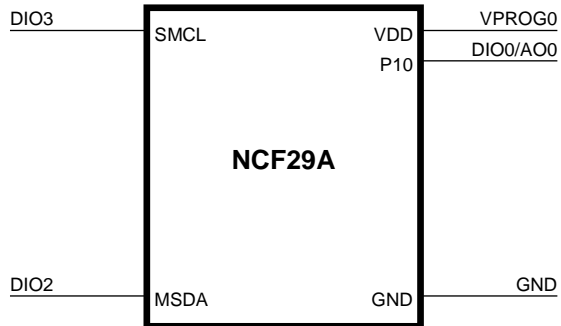
## 10.87 NXP NCF29A

### 10.87.1 Specific Commands

Table 94: NXP NCF29A Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the UART baud rate (bps)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blank checks Flash memory, Eeprom memory.</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD PROGRAM F E <src offset> <tgt start addr>	<b>Programs Flash memory, Eeprom memory.</b>
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	<b>Verifies Flash memory, Eeprom memory.</b>
PCMD RUN	<b>Runs target application</b>

## 10.87.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 138: Example of Connection to NXP NCF29A Target Device

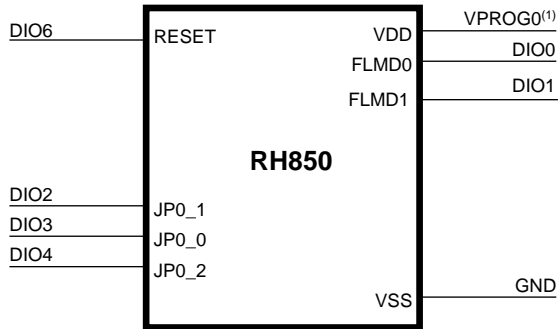
## 10.88 Renesas RH850

### 10.88.1 Specific Commands

Table 95: Renesas RH850 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE CSI	<b>Sets the communication mode</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz)</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD MASSERASE F E	<b>Issues masserase of Flash memory (F)</b>
TPCMD BLOCKERASE U <tgt start addr> <len>	<b>Erases one or more blocks of user area flash memory.</b>
TPCMD BLANKCHECK F E <tgt start addr> <len>	<b>Blankchecks Flash memory. Blankchecks len locations starting from the address specified by tgt start address.</b>
TPCMD PROGRAM F E U X <src offset> <tgt start addr> <len>	<b>Programs Flash (F), EEPROM (E) or Option Bytes (O) memory</b>
TPCMD VERIFY F E U X <src offset> <tgt start addr> <len>	<b>Verifies Flash (F), EEPROM (E) or Option Bytes (O) memory</b>
TPCMD DISCONNECT	<b>Exits the programming mode and powers off the board</b>
TPCMD RUN	<b>Runs target application</b>
TPCMD MASSERASE F	<b>Issues masserase of Flash memory (F)</b>

## 10.88.2 Connections CSI



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 139: Example of Connection to a Renesas RH850 Target Device MCU through CSI

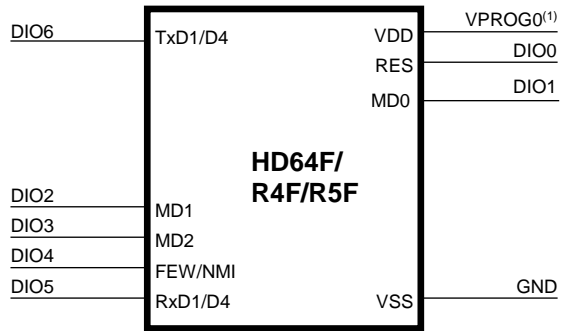
## 10.89 Renesas HD64F/R4F/R5F

### 10.89.1 Specific Commands

Table 96: Renesas HD64F/R4F/R5F Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE UART	Sets the communication mode
TCSETPAR SCLK <frequency Hz>	Sets the communication serial clock frequency (Hz)
TCSETPAR BAUDRATE <baudrate bps>	Sets the communication baud rate (bps)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR CLKOUT 5000000 12500000 6250000 3125000  562500 0	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.
TCSETPAR CK_MODE <mode>	Clock mode
TCSETPAR CKM <multiplier>	Main clock multiplier
TCSETPAR CKP <multiplier>	Peripheral clock multiplier
TCSETPAR DIO1 DIO2 DIO3 DIO4 0 1	Specifies FlashRunner DIOx lines to set the device’s mode pins to Serial Boot Mode
TCSETPAR KEY_CODE CONST <values>	Key code settings
TCSETPAR KEY_CODE SRC <src start addr>	Reads the key-code data from binary file
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD BLOCKERASE F <tgt start addr> <len>	Erases one or more Flash memory blocks
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	Verifies Flash memory. Reads back all written data (R, slow) or compares a checksum (S, fast).
TPCMD READ F <tgt start addr> <len>	Reads Flash memory
TPCMD RUN	Runs target application

## 10.89.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 140: Example of Connection to a HD64F/R4F/R5F Target Device (UART Comm. Mode)

## 10.90 Renesas M30

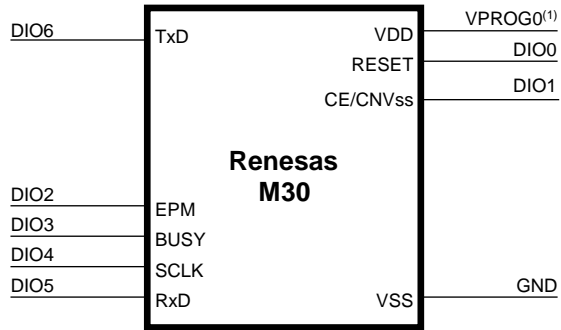
### 10.90.1 Specific Commands

Table 97: Renesas M30 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE USART	<b>Sets the communication mode</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDELAY <time ms>	<b>Sets the reset delay time which extnd both Reset up and Reset down time</b>
TCSETPAR CLKOUT 5000000 12500000 6250000 3125000  562500 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
TCSETPAR ID_CODE CONST <id1> ... <id7> TCSETPAR ID_CODE SRC <src offset>	<b>Specifies the ID Code necessary for performing programming operations. The ID Code can be passed to FlashRunner directly or read from the source specified by the TPSETSRC command.</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the frequency for communication mode (Hz)</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSErase F	<b>Mass erases Flash memory</b>
TPCMD BLOCKERASE F <tgt start addr> <len>	<b>Erases one or more Flash memory blocks</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Reads back all written data (R, slow) or compares a checksum (S, fast).</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads Flash memory</b>
TPCMD RUN	<b>Runs target application</b>



## 10.90.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 141: Example of Connection to a Renesas Target Device (USART Comm. Mode)

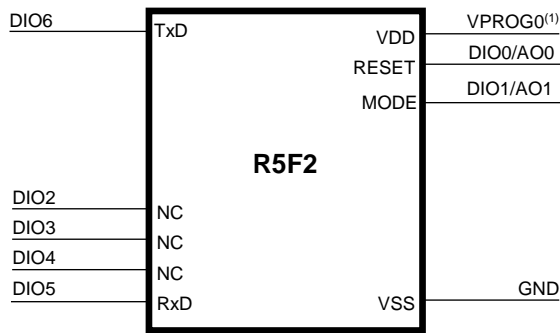
## 10.91 Renesas R5F2

### 10.91.1 Specific Commands

Table 98: Renesas R5F2 Specific Commands

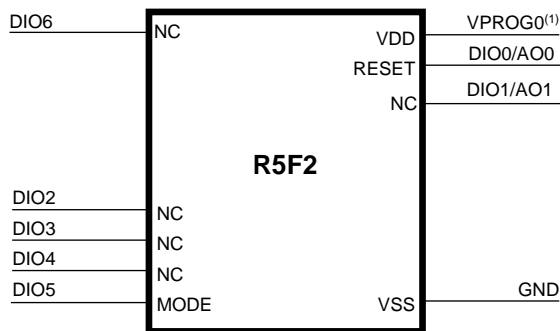
Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE UART SINGLE_WIRE	<b>Sets the communication mode</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the communication baud rate (bps).</b> For the UART mode, the baudrate values can be 9600, 19200, 38400, 57600 and 115200 bps For the SINGLE_WIRE mode, the baudrate values can be 9600, 250000 and 500000 bps
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDELAY <time ms>	<b>Sets the reset delay time which extnd both Reset up and Reset down time</b>
TCSETPAR CLKOUT 5000000 12500000 6250000 3125000  562500 0	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b> 2 1
TCSETPAR ID_CODE CONST <id1> ... <id7> TCSETPAR ID_CODE SRC <src offset>	<b>Specifies the ID Code necessary for performing programming operations. The ID Code can be passed to FlashRunner directly or read from the source specified by the TPSETSRC command.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD BLOCKERASE F <tgt start addr> <len>	<b>Erases one or more Flash memory blocks</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory</b>
TPCMD VERIFY F R S <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Reads back all written data (R, slow) or compares a checksum (S, fast).</b>
TPCMD READ F <tgt start addr> <len>	<b>Reads Flash memory</b>
TPCMD RUN	<b>Runs target application</b>

## 10.91.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 142: Example of Connection to a R5F2 Target Device (UART Comm. Mode)



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 143: Example of Connection to a R5F2 Target Device (Single-Wire Comm. Mode)

## 10.92 Renesas R5F1

### 10.92.1 Specific Commands

Table 99: Renesas R5F1 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE UART1 UART2	<b>Sets the communication mode</b>
TCSETPAR BAUDRATE <baudrate bps>	<b>Sets the baud rate (bps) for dual voltage devices in UART mode</b>
TCSETPAR SCLK <frequency Hz>	<b>Sets the communication serial clock frequency (Hz)</b>
TCSETPAR FOSC <frequency Hz>	<b>Sets the external oscillator frequency (Hz)</b>
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR PWUP <time ms>	<b>Sets the power up time (ms)</b>
TCSETPAR PWDOWN <time ms>	<b>Sets the power down time (ms)</b>
TCSETPAR RSTUP <time μs>	<b>Sets the Reset up time (μs)</b>
TCSETPAR RSTDOWN <time μs>	<b>Sets the Reset down time (μs)</b>
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	<b>Sets the Reset line driving mode</b>
TCSETPAR CLKOUT <frequency Hz>	<b>Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner's ISP connector. Useful for target devices with no external oscillator.</b>
<b>TPCMD Command</b>	
TPCMD BLANKCHECK C D <tgt start addr> <len>	<b>Blank checks Code Flash memory, Data flash memory.</b>
TPCMD MASSErase C D	<b>Mass erases Code Flash memory, Data flash memory.</b>
TPCMD BLOCKERASE C D <tgt start addr> <len>	<b>Erases one or more Code Flash memory, Data flash memory.</b>
TPCMD PROGRAM C D <src offset> <tgt start addr> <len>	<b>Programs Code Flash memory, Data flash memory.</b>
TPCMD VERIFY C D R S <src offset> <tgt start addr> <len>	<b>Verifies Code Flash memory, Data flash memory. Reads back all written data (R, slow) or compares a checksum (S, fast).</b>
TPCMD RUN	<b>Runs target application</b>

## 10.92.2 Connections (R5F1)

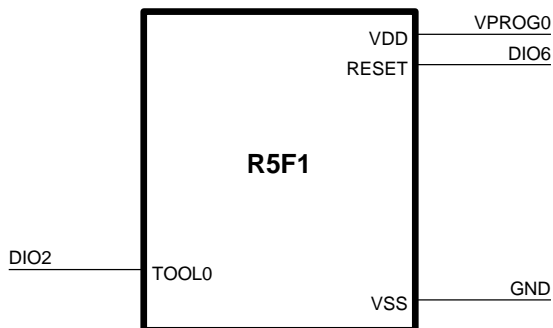


Figure 144: Example of Connection to R5F1 Target Device through UART1/UART2

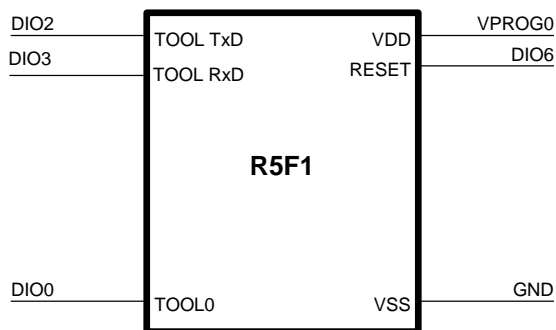


Figure 145: Example of Connection to R5F1 Target Device through UART1/UART2

## 10.93 Renesas R5F5

### 10.93.1 Specific Commands

Table 100: Renesas R5F5 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR CMODE UART	Sets the communication mode
TCSETPAR BAUDRATE <baudrate bps>	Sets the baud rate (bps) for dual voltage devices in UART mode
TCSETPAR SCLK <frequency Hz>	Sets the communication serial clock frequency (Hz)
TCSETPAR FOSC <frequency Hz>	Sets the external oscillator frequency (Hz)
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the “ISP” connector (mV)
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTUP <time μs>	Sets the Reset up time (μs)
TCSETPAR RSTDOWN <time μs>	Sets the Reset down time (μs)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR CLKOUT <frequency Hz>	Generates an auxiliary clock signal at the CLKOUT pin of FlashRunner’s ISP connector. Useful for target devices with no external oscillator.
TCSETPAR ID_CODE CONST <id1> ... <id7> TCSETPAR ID_CODE SRC <src offset>	Specifies the ID Code necessary for performing programming operations. The ID Code can be passed to FlashRunner directly or read from the source specified by the TPSETSRC command.
<b>TPCMD Command</b>	
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD MASSErase F B	Mass erases Flash memory
TPCMD BLOCKERASE F <tgt start addr> <len>	Erases one or more Flash memory blocks
TPCMD PROGRAM F E B L <src offset> <tgt start addr> <len>	Programs Flash memory (F), Eeprom memory (E), User boot area memory (B), lock bits (L).
TPCMD VERIFY F E B L R S <src offset> <tgt start addr> <len>	Verifies Flash memory (F), Eeprom memory (E), User boot area memory (B), lock bits (L). Reads back all written data (R, slow) or compares a checksum (S, fast).
TPCMD PROGRAM_OFS <src offset>	Programs OFS0 and OFS1. The size is fixed to 8 bytes
TPCMD PROGRAM_TM <src offset>	Programs TMINF and enable the TM function. The command writes 0 to TMEF. TMEF [2:0] bits. The size is fixed to 4 bytes.
TPCMD PROGRAM_ID <src offset>	Programs a different ID_code. The size is fixed

	to 16 bytes
TPCMD READ F <tgt start addr> <len>	Reads Flash memory
TPCMD RUN	Runs target application

### 10.93.2 Connections (R5F5)

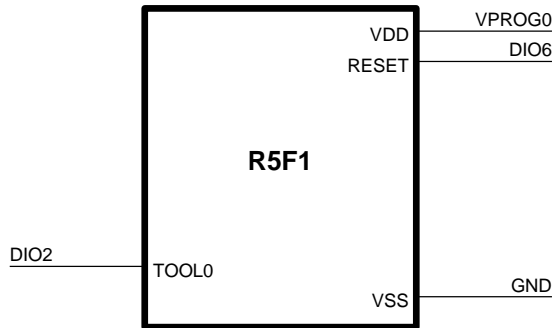


Figure 146: Example of Connection to R5F1 Target Device through UART

## 10.94 Texas Instruments BQ40/UCD3

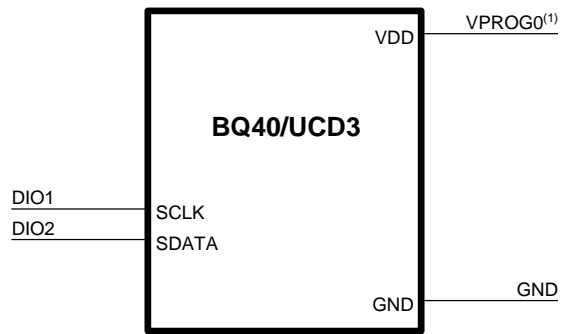
### 10.94.1 Specific Commands

Table 101: Texas Instruments BQ40/UCD3 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR RSTDOWN <time us>	Reset down time (uS)
TCSETPAR RSTUP <time us>	Reset up time (uS)
TCSETPAR RSTDRV	Reset driving mode
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR SCLK <frequency Hz>	Sets the communication serial clock frequency (Hz)
<b>TPCMD Command</b>	
TPCMD CONNECT	Power on the board and connect the programmer to it
TPCMD MASSErase C	Mass erase complete flash memory, Program memory, Data memory (C, F, E)
TPCMD ERASE F E <src offset> <len>	Erase program/Data page (F/E)
TPCMD BLANKCHECK F E <tgt start addr> <len>	Blank checks program/Data (F/E)
TPCMD PROGRAM F E <src offset> <tgt start addr> <len>	Programs Flash memory and EEPROM memory.
TPCMD VERIFY F E <src offset> <tgt start addr> <len>	Verify Flash memory and EEPROM memory.
TPCMD PROGRAM_DFI <src offset> <IF addr> <calibration>	Writes files in Data Flash memory.
TPCMD VERIFY_DFI R S <src offset> <IF addr> <calibration>	Verifies Data Flash memory.
TPCMD DISCONNECT	Disconnect the programmer to the board and power off it
TPCMD RUN	Runs target application



## 10.94.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device

Figure 147: Example of Connection to BQ40/UCD3 Target Device

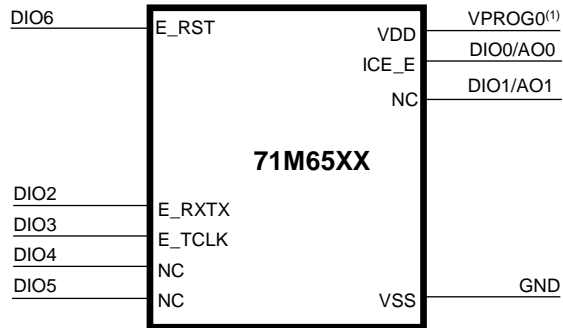
## 10.95 Maxim Teridian 71M65xx TRDN

### 10.95.1 Specific Commands

Table 102: Maxim teridain 71M65xx Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	Sets the VDD voltage (mV)
TCSETPAR VDD_AUX <voltage mV>	Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)
TCSETPAR RSTDOWN <time us>	Reset down time (uS)
TCSETPAR RSTUP <time us>	Reset up time (uS)
TCSETPAR RSTDRV	Reset driving mode
TCSETPAR PWUP <time ms>	Sets the power up time (ms)
TCSETPAR PWDOWN <time ms>	Sets the power down time (ms)
TCSETPAR RSTDRV PUSHPULL OPENDRAIN	Sets the Reset line driving mode
TCSETPAR RST_EXIT LOW TRISTATE	Sets the reset exit mode
<b>TPCMD Command</b>	
TPCMD CONNECT	Power on the board and connect the programmer to it
TPCMD MASSERASE F	Mass erases Flash memory
TPCMD BLANKCHECK F <tgt start addr> <len>	Blank checks Flash memory
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	Programs Flash memory. Addresses and length are expressed in 16-bit word.
TPCMD VERIFY F E R S <src offset> <tgt start addr> <len>	Verifies Flash memory. Addresses and length are expressed in 16-bit word.
TPCMD DISCONNECT	Disconnect the programmer to the board and power off it
TPCMD RUN	Runs target application

## 10.95.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 148: Example of Connection to Maxim Teridian 71M65xx Target Device

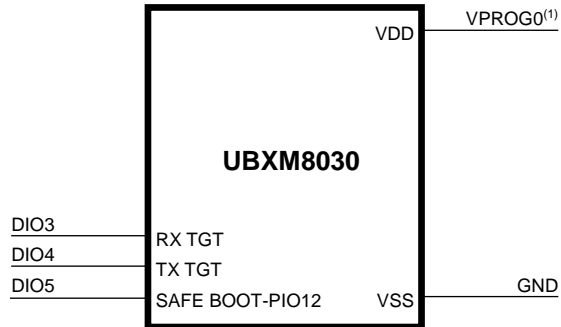
# 10.96 UBX UBXM8030

## 10.96.1 Specific Commands

Table 103: UBX UBXM8030 Specific Commands

Command Syntax	Description
<b>TCSETPAR Command</b>	
TCSETPAR VDD <voltage mV>	<b>Sets the VDD voltage (mV)</b>
TCSETPAR VDD_AUX <voltage mV>	<b>Sets an auxiliary voltage on the VPROG1 line of the "ISP" connector (mV)</b>
TCSETPAR CMODE UART	<b>Sets the communication mode</b>
<b>TPCMD Command</b>	
TPCMD CONNECT	<b>Power on the board and connect the programmer to it</b>
TPCMD MASSERASE F	<b>Mass erases Flash memory</b>
TPCMD BLANKCHECK F <tgt start addr> <len>	<b>Blank checks Flash memory</b>
TPCMD PROGRAM F <src offset> <tgt start addr> <len>	<b>Programs Flash memory. Addresses and length are expressed in 16-bit word.</b>
TPCMD VERIFY F E R S <src offset> <tgt start addr> <len>	<b>Verifies Flash memory. Addresses and length are expressed in 16-bit word.</b>
TPCMD PRINT_PROGRAM_ACK	<b>Print program acknowledge message</b>
TPCMD READ LLC NMEA	<b>Check low level configuration. Check the NMEA return</b>
TPCMD DISCONNECT	<b>Disconnect the programmer to the board and power off it</b>
TPCMD RUN	<b>Runs target application</b>

## 10.96.2 Connections



<sup>(1)</sup> Connect this line if you want FlashRunner to automatically power the target device.

Figure 149: Example of Connection to Maxim Teridian 71M65xx Target Device

# 11 Errors

## 11.1 Overview

FlashRunner signals system errors via its “STATUS” LED (basic hardware errors) or via answers to commands sent from the host system.

## 11.2 Status LED

FlashRunner features an automatic, self-diagnostic procedure that is executed every time the instrument is powered. If a hardware malfunction is detected, the “STATUS” LED blinks.

The “STATUS” LED also blinks when FlashRunner is busy updating its firmware, when no SD card is present, or when an invalid card is present. The number of blinks of the “STATUS” LED indicates which error has occurred. The following table describes the error corresponding to the number of blinks.

Table 104: Status LED

Blinks	Status Description
1	FlashRunner is busy updating the firmware
2	SD card not present
3	Invalid card contents (the card has not been formatted correctly or doesn't contain the required directory structure). Use the <code>ESFORMAT</code> command to reformat the card.
4	Hardware self-test error
5	SD card locked, invalid password

Each blink is separated by a delay of 200 ms, and each series of blinks is repeated with a 1.5 s delay.

## 11.3 Command Errors

The following table lists error codes that FlashRunner may send to the host system as response to commands sent from the host system.

## Errors

Table 105: Error Summary

Error #	Description
<b>General Errors</b>	
\$0010	LVDSET LPW <b>command: missing or invalid parameter</b>
\$0011	LVDSET LPT <b>command: missing or invalid parameter</b>
\$0012	LVDSET CH <b>command: missing parameter</b>
\$0013	LVDSET CH <b>command: invalid parameter</b>
\$0014	LVDSET ENABLE <b>command: the LVDSET CH command specified an invalid channel</b>
\$0015	LVDSET ENABLE <b>command: execution error</b>
\$0016	LVDSET ENABLE <b>command: the LVDSET LPT command specified an out-of-range value</b>
\$0017	LVDSET ENABLE <b>command: the LVDSET LPW command specified an out-of-range value</b>
\$0018	LVDSET DISABLE <b>command: execution error</b>
\$001D	LVDSET <b>command: missing command option</b>
\$001E	LVDSET <b>command: invalid command option</b>
\$001F	LVDSET <b>command: internal error</b>
\$0031	RLYGET <b>command: missing command option</b>
\$0032	RLYGET <b>command: invalid command option</b>
\$0033	RLYGET <b>command: internal error</b>
\$0034	RLYSET <b>command: missing command option</b>
\$0035	RLYSET <b>command: invalid command option</b>
\$0036	RLYSET <b>command: internal error</b>
\$0037	RLYSET OPEN <b>command: missing parameter</b>
\$0038	RLYSET OPEN <b>command: invalid parameter</b>
\$0039	RLYSET OPEN <b>command: invalid when the relay driving mode is set to hardware (RLYSET DRVMODE HW)</b>
\$003A	RLYSET OPEN <b>command: execution error</b>
\$003B	RLYSET CLOSE <b>command: missing parameter</b>
\$003C	RLYSET CLOSE <b>command: invalid parameter</b>
\$003D	RLYSET CLOSE <b>command: invalid when the relay driving mode is set to hardware (RLYSET DRVMODE HW)</b>
\$003E	RLYSET CLOSE <b>command: execution error</b>
\$003F	RLYSET DRVMODE <b>command: missing command option</b>
\$0040	RLYSET DRVMODE <b>command: invalid command option</b>
\$0041	RLYSET DRVMODE <b>command: execution error</b>
\$0045	RLYGET DRVMODE <b>command: execution error</b>
\$0046	RLYGET CYCLES <b>command: missing parameter</b>
\$0047	RLYGET CYCLES <b>command: invalid parameter</b>
\$0048	RLYGET CYCLES <b>command: execution error</b>
\$0050	OCDSET <b>command: missing command option</b>
\$0051	OCDSET <b>command: invalid command option</b>
\$0052	OCDSET <b>command: internal error</b>



Error #	Description
\$0056	OCDSET ENABLE <b>command: execution error</b>
\$0057	OCDSET DISABLE <b>command: execution error</b>
\$0060	TPCMD <b>command: execution error</b>
\$0061	TPCMD <b>command: low-voltage event detected</b>
\$0062	TPCMD <b>command: over-current event detected on VPROG0 line</b>
\$0063	TPCMD <b>command: over-current event detected on VPROG1 line</b>
\$0064	TPCMD <b>command: low-voltage detect capabilities not supported by the selected programming algorithm</b>
\$0065	TPSTART <b>command: execution error</b>
\$0066	TPSTART <b>command: low-voltage event detected</b>
\$0067	TPSTART <b>command: over-current event detected on VPROG0 line</b>
\$0068	TPSTART <b>command: over-current event detected on VPROG1 line</b>
\$0069	TPSTART <b>command: low-voltage detect capabilities not supported by the selected programming algorithm</b>
\$006A	TPEND <b>command: execution error</b>
\$006B	TPEND <b>command: low-voltage event detected</b>
\$006C	TPEND <b>command: over-current event detected on VPROG0 line</b>
\$006D	TPEND <b>command: over-current event detected on VPROG1 line</b>
\$006E	TPEND <b>command: low-voltage detect capabilities not supported by the selected programming algorithm</b>
\$0080	FSENDFILE <b>command: execution error</b>
\$0081	FSENDFILE <b>command: file and SD card passwords do not match, or SD card not encrypted</b>
\$0082	FSENDFILE <b>command: FlashRunner could not decrypt the file</b>
\$0088	ESGETFILE <b>command: execution error</b>
\$0089	ESGETFILE <b>command: operation not allowed</b>
\$008A	ESGETFILE <b>command: FlashRunner could not encrypt the file</b>
\$00FF	<b>Command not supported by this FlashRunner hardware model or configuration</b>
\$0100	<b>Invalid command</b>
\$0101	<b>Numeric parameter invalid or out of range</b>
\$0104	<b>Command not executable inside a script, or command execution not allowed to the active user</b>
\$0105	<b>Missing command</b>
\$0106	<b>Missing parameter</b>
\$0107	FSFORMAT <b>command: execution error</b>
\$0108	FSRD <b>command: missing or invalid directory parameter</b>
\$0109	FSRD <b>command: execution error</b>
\$010A	FSMD <b>command: missing or invalid directory parameter</b>
\$010B	FSMD <b>command: execution error</b>
\$010C	FSCD <b>command: missing or invalid directory parameter</b>
\$010D	FSCD <b>command: execution error</b>
\$010E	FSENDFILE <b>command: protocol parameter missing or protocol not supported</b>
\$010F	FSENDFILE <b>command: missing or invalid filename parameter</b>
\$0110	FSENDFILE <b>command: card write error</b>

## Errors

Error #	Description
\$0111	FSGETFILE <b>command: protocol parameter missing or protocol not supported</b>
\$0112	FSGETFILE <b>command: missing or invalid filename parameter</b>
\$0113	FSGETFILE <b>command: card read error</b>
\$0114	FSRF <b>command: missing or invalid filename parameter</b>
\$0115	FSRF <b>command: execution error</b>
\$0116	FSLS <b>command: card read error</b>
\$0117	FSMKLOG <b>command: missing or invalid filename parameter</b>
\$0118	FSMKLOG <b>command: card write error</b>
\$0119	FSCLRLOG <b>command: card write error</b>
\$011A	FSSETTIME <b>command: execution error</b>
\$011B	FSSETTIME <b>command: missing, invalid or out of range parameter</b>
\$011C	FSGETTIME <b>command: execution error</b>
\$011D	SSETBR <b>command: missing or invalid baud rate</b>
\$011E	SGETCRC <b>command: missing or invalid memory source parameter</b>
\$011F	SGETCRC DYNAMIC <b>command: missing, invalid or out of range parameter</b>
\$0120	SGETCRC FILE <b>command: missing, invalid or out of range parameter</b>
\$0121	SGETCRC FILE <b>command: card read error</b>
\$0122	DMSET <b>command: missing, invalid or out of range parameter</b>
\$0123	DMSET <b>command: missing data parameters</b>
\$0124	DMGET <b>command: missing, invalid or out of range parameter</b>
\$0125	TCSETDEV <b>command: missing or invalid device code</b>
\$0126	TCSETDEV <b>command: missing or invalid manufacturer code</b>
\$0127	TCSETDEV <b>command: missing or invalid algorithm code</b>
\$0128	TCSETDEV <b>command: specified algorithm not present on card</b>
\$0129	TCSETDEV <b>command: specified device not supported</b>
\$012A	TCSETDEV <b>command: missing or invalid algorithm license</b>
\$012B	TCSETDEV <b>command: algorithm loading error</b>
\$012C	TPSETSRC <b>command: missing or invalid source parameter</b>
\$012D	TPSETSRC FILE <b>command: missing or invalid filename parameter</b>
\$012E	TPSETSRC FILE <b>command: file reading error (CRC error)</b>
\$012F	TPSETSRC FILE <b>command: file open error</b>
\$0130	SGETERR <b>command: execution error</b>
\$0131	RUN <b>command: missing parameter</b>
\$0132	TPCMD <b>command: command not preceded by TPSTART command</b>
\$0133	TPCMD <b>command: command not preceded by TCSETDEV command</b>
\$0134	TCSETPAR <b>command: command not preceded by TCSETDEV command</b>
\$0135	TPSTART <b>command: command not preceded by TCSETDEV command</b>
\$0136	TPEND <b>command: command not preceded by TPSTART command</b>

Error #	Description
\$0137	<code>MXGET</code> <b>command: missing or invalid parameter</b>
\$0138	<code>MXGET</code> <b>command: missing or invalid parameter type</b>
\$0139	<code>MXGET RELAYCYCLES</code> <b>command: command execution error</b>
\$013A	<code>MXGET</code> <b>command: generic execution error</b>
\$013E	<code>MXOPEN</code> <b>command: command execution error</b>
\$013F	<code>MXCLOSE</code> <b>command: missing or invalid parameter</b>
\$0140	<code>MXCLOSE</code> <b>command: command execution error</b>
\$0141	<code>FSGETTIME</code> <b>command: command not supported by this FlashRunner model</b>
\$0142	<code>FSSETTIME</code> <b>command: command not supported by this FlashRunner model</b>
\$0143	<code>FSMKLOG</code> <b>command: command not supported by this FlashRunner model</b>
\$0144	<code>FSCLRLOG</code> <b>command: command not supported by this FlashRunner model</b>
\$0145	<code>SGETCRC DYNAMIC</code> <b>command: command not supported by this FlashRunner model</b>
\$0146	<code>DMSET DYNAMIC</code> <b>command: command not supported by this FlashRunner model</b>
\$0147	<code>DMGET DYNAMIC</code> <b>command: command not supported by this FlashRunner model</b>
\$0148	<code>TPSETSRC DYNAMIC</code> <b>command: command not supported by this FlashRunner model</b>
\$0149	<code>SGETSTS</code> <b>command: execution error</b>
\$0150	<code>SGETSTS</code> <b>command: invalid command option</b>
\$0151	<code>SGETCRC32</code> <b>command: missing or invalid source parameter</b>
\$0153	<code>SGETCRC32 DYNAMIC</code> <b>command: FlashRunner model doesn't feature Dynamic memory</b>
\$0154	<code>SGETCRC32 FILE</code> <b>command: missing or invalid parameter</b>
\$0155	<code>SGETCRC32 FILE</code> <b>command: file reading error</b>
\$0164	<code>ACHGPWD</code> <b>command: missing or invalid option/parameter</b>
\$0165	<code>ACHGPWD</code> <b>command: execution error</b>
\$0166	<code>ACHGUSER</code> <b>command: missing or invalid command option</b>
\$0167	<code>ACHGUSER</code> <b>command: execution error</b>
\$0168	<code>ASECURE</code> <b>command: execution error</b>
\$0169	<code>ASECURE</code> <b>command: Data Protection System license not found on SD card</b>
\$016A	<code>AUNSECURE</code> <b>command: execution error</b>
\$0181	<code>DLOCK</code> <b>command: missing or invalid password</b>
\$0182	<code>DLOCK</code> <b>command: execution error</b>
\$0183	<code>DLOCK</code> <b>command: data encryption not supported by SD card</b>
\$0184	<code>DUNLOCK</code> <b>command: missing or invalid password</b>
\$0185	<code>DUNLOCK</code> <b>command: execution error</b>
\$0191	<code>TPSETDST</code> <b>command: missing or invalid destination parameter</b>
\$0195	<code>TPSETDST DYNAMIC</code> <b>command: command not supported by this FlashRunner model</b>
\$01A0	<code>TST</code> <b>command: execution error</b>
\$01A1	<code>TST WRWORD</code> <b>command: TCSETPAR SOFTEC DEFAULT DEFAULT command not sent</b>
\$01A2	<code>TST WRWORD</code> <b>command: missing or invalid parameter</b>

## Errors

---

Error #	Description
\$01A3	TST WRWORD <b>command: missing or invalid parameter</b>
\$01A4	TST WRWORD <b>command: missing or invalid parameter</b>
\$01A5	TST WRWORD <b>command: execution error</b>
\$01A6	TST RDWORD <b>command: missing or invalid parameter</b>
\$01A7	TST RDWORD <b>command: missing or invalid parameter</b>
\$01A8	TST RDWORD <b>command: execution error</b>
<b>Script - Specific Errors</b>	
\$0200	<b>Script file not found</b>
\$0201	<b>Invalid script file</b>
\$0202	<b>Script number out of range</b>
\$0203	<b>Invalid script directive</b>
\$0204	#DELAY <b>directive: missing, invalid or out of range parameter</b>
\$0205	#WAIT <b>directive: missing or invalid parameter</b>
\$0206	#LOG ON <b>directive: missing, invalid or out of range parameter</b>
\$0207	#LOG ON <b>directive: execution error</b>
\$0208	#LOG OFF <b>directive: execution error</b>
\$0209	#HALT ON <b>directive: missing or invalid parameter</b>
\$020A	<b>Could not write to the log file</b>
\$020B	#PRINT <b>directive: missing, invalid or out of range parameter</b>
\$020C	#PRINT <b>directive: execution error</b>

Error #	Description
<b>FR04 - Specific errors</b>	
\$E100	<b>FR04ENG1 does not communicate</b>
\$E101	<b>FR04ENG2 does not communicate</b>
\$E102	<b>FR04ENG3 does not communicate</b>
\$E103	<b>FR04ENG4 does not communicate</b>
\$E104	<b>FR04ENG1 wrong answer</b>
\$E105	<b>FR04ENG2 wrong answer</b>
\$E106	<b>FR04ENG3 wrong answer</b>
\$E107	<b>FR04ENG4 wrong answer</b>
\$E108	<b>FR04ENG1 is busy</b>
\$E109	<b>FR04ENG2 is busy</b>
\$E10A	<b>FR04ENG3 is busy</b>
\$E10B	<b>FR04ENG4 is busy</b>
<b>Freescale HC08 - Specific Errors</b>	
\$1100	TCSETDEV <b>command: manufacturer not supported</b>
\$1101	TCSETDEV <b>command: algorithm not found on card</b>
\$1102	TCSETDEV <b>command: device not supported</b>
\$1103	TCSETDEV <b>command: corrupted algorithm file</b>
\$1104	TCSETDEV <b>command: internal hardware configuration error</b>
\$1105	TCSETPAR <b>command: parameter not supported</b>
\$1106	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$1107	TPSTART <b>command: command not preceded by TCSETPAR FOSC command</b>
\$1108	TCSETPAR FDIV <b>command: missing, invalid or out of range parameter</b>
\$1109	TPSTART <b>command: command not preceded by TCSETPAR FDIV command</b>
\$110A	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$110B	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$110C	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$110D	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$110E	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$110F	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$1110	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$1111	TPCMD <b>command: parameter not supported</b>
\$1112	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1113	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1114	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1115	TPCMD VERIFY <b>command: Flash verify error</b>
\$1116	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$1117	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1118	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>

## Errors

Error #	Description
\$1119	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$111A	TPCMD PROGRAM <b>command: Flash program error</b>
\$111B	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$111C	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$111D	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$111E	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$111F	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$1120	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1121	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1122	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$1123	TPCMD TRIM <b>command: missing, invalid or out of range parameter</b>
\$1124	TPCMD TRIM <b>command: trimming not supported by target device</b>
\$1125	TPCMD TRIM <b>command: execution error</b>
\$1126	TPCMD RUN <b>command: execution error</b>
\$1127	TPCMD SETPWD <b>command: missing, invalid or out of range parameter</b>
\$1128	TPCMD SETPWD FILE <b>command: file read error</b>
\$1129	TPCMD READ <b>command: missing, invalid or out of range parameter</b>
\$112A	TPCMD READ <b>command: memory read error</b>
\$112B	TPEND <b>command: execution command</b>
\$112C	TPSTART <b>command: the combination of TCSETPAR FOSC and TCSETPAR FDIV commands specify an invalid programming frequency</b>
\$112D	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$112E	TCSETPAR RSTDW <b>command: missing, invalid or out of range parameter</b>
\$112F	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1130	TCSETPAR HSCI_SPEED <b>command: missing, invalid or out of range parameter</b>
\$1131	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$1132	TCSETPAR IRQDRV <b>command: missing, invalid or out of range parameter</b>
\$1133	TPCMD TRIM <b>command: out of tolerance</b>
\$1134	TPCMD TRIM <b>command: invalid tolerance</b>
\$1135	TPCMD TRIM <b>command: invalid INT frequency</b>
<b>Atmel AVR8 - Specific Errors</b>	
\$1200	TCSETDEV <b>command: manufacturer not supported</b>
\$1201	TCSETDEV <b>command: algorithm not found on card</b>
\$1202	TCSETDEV <b>command: device not supported</b>
\$1203	TCSETDEV <b>command: internal hardware configuration error</b>
\$1204	TCSETPAR <b>command: parameter not supported</b>
\$1205	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$1206	TPSTART <b>command: command not preceded by TCSETPAR FCPU command</b>
\$1207	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$1208	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$1209	TPCMD <b>command: parameter not supported</b>
\$1210	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1211	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1212	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1213	TPCMD VERIFY <b>command: Flash verify error</b>
\$1214	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$1215	TPCMD VERIFY <b>command: Fuse Bits verify error</b>
\$1216	TPCMD VERIFY <b>command: Lock Bits verify error</b>
\$1217	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1218	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1219	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$121A	TPCMD PROGRAM <b>command: Flash program error</b>
\$121B	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$121C	TPCMD PROGRAM <b>command: Fuse Bits program error</b>
\$121D	TPCMD PROGRAM <b>command: Lock Bits program error</b>
\$121E	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$121F	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1220	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1221	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$1222	TPCMD MASSERASE <b>command: mass erase error</b>
\$1223	TPCMD SAVECALIBRVALUE <b>command: missing or invalid parameter</b>
\$1224	TPCMD SAVECALIBRVALUE <b>command: target address parameter or frequency out of range</b>
\$1225	TPCMD RUN <b>command: execution error</b>
\$1226	TPEND <b>command: execution error</b>
\$1227	TPSTART <b>command: execution error</b>
\$1228	<b>Can't establish communication with the target device</b>
\$1230	TPCMD READ <b>command: missing or invalid parameter</b>
\$1231	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$1232	TPCMD READ <b>command: read error</b>
\$1233	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1234	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1235	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1236	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1237	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$1238	TPCMD SETJTCLK <b>command: missing, invalid or out of range parameter</b>
\$1239	TCSETPAR CMODE <b>command: invalid protocol</b>
\$123A	TCSETPAR JTCLK <b>command: missing parameter</b>

## Errors

Error #	Description
\$123B	TPCMD SETDAISYCHAIN <b>command: missing, invalid or out of range parameter</b>
<b>STMicroelectronics ST7C and ST7F - Specific Errors</b>	
\$1300	TCSETDEV <b>command: manufacturer not supported</b>
\$1301	TCSETDEV <b>command: algorithm not found on card</b>
\$1302	TCSETDEV <b>command: device not supported</b>
\$1303	TCSETDEV <b>command: corrupted algorithm file</b>
\$1304	TCSETDEV <b>command: internal hardware configuration error</b>
\$1305	TCSETPAR <b>command: parameter not supported</b>
\$1306	TCSETPAR ICPMODE <b>command: missing or invalid parameter</b>
\$1307	TPSTART <b>command: command not preceded by TCSETPAR ICPMODE command</b>
\$1308	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1309	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$130A	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$130B	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$130C	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$130D	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$130E	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$130F	TPSTART <b>command: command not preceded by TCSETPAR CLKOUT command</b>
\$1310	TPCMD <b>command: parameter not supported</b>
\$1311	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1312	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1313	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1314	TPCMD VERIFY <b>command: Flash verify error</b>
\$1315	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$1316	TPCMD VERIFY <b>command: Option Byte(s) verify error</b>
\$1317	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1318	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1319	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$131A	TPCMD PROGRAM <b>command: Flash program error</b>
\$131B	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$131C	TPCMD PROGRAM <b>command: Option Byte(s) program error</b>
\$131D	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$131E	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$131F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1320	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$1321	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1322	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1323	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>



Error #	Description
\$1324	TPCMD UNPROTECT <b>command: execution error</b>
\$1325	TPCMD RESTORE_RC_CALIBRATION <b>command: execution error</b>
\$1326	TPCMD RUN <b>command: execution error</b>
\$1327	TPCMD READ <b>command: missing or invalid parameter</b>
\$1328	TPCMD READ <b>command: execution error</b>
\$1329	TPEND <b>command: execution command</b>
\$132A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$132B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$132C	TPCMD TRIM <b>command: missing, invalid or out of range parameter</b>
\$132D	TPCMD TRIM <b>command: trimming error</b>
\$132E	TPCMD TRIM <b>command: trimming not supported by the selected device</b>
\$132F	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$1330	TPCMD DUMP <b>command: missing, invalid or out of range parameter</b>
\$1331	TPCMD DUMP <b>command: missing, invalid or out of range parameter</b>
\$1332	TPCMD DUMP <b>command: dump not supported by the selected device</b>
\$1333	<b>dump execution error</b>
\$1334	TPCMD DUMP <b>command: dump execution error</b>
<b>Microchip PIC18 - Specific Errors</b>	
\$1400	TCSETDEV <b>command: manufacturer not supported</b>
\$1401	TCSETDEV <b>command: algorithm not found on card</b>
\$1402	TCSETDEV <b>command: device not supported</b>
\$1403	TCSETDEV <b>command: internal hardware configuration error</b>
\$1404	TCSETPAR <b>command: parameter not supported</b>
\$1405	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$1406	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$1407	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1408	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$1409	TCSETPAR VPP <b>command: missing, invalid or out of range parameter</b>
\$140A	TPSTART <b>command: command not preceded by TCSETPAR VPP command</b>
\$140B	TCSETPAR PROGMODE <b>command: missing or invalid parameter</b>
\$140C	TPSTART <b>command: command not preceded by TCSETPAR PROGMODE command</b>
\$140D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$140E	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$140F	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1410	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$1411	TPCMD <b>command: parameter not supported</b>
\$1412	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1413	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>

## Errors

Error #	Description
\$1414	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1415	TPCMD VERIFY <b>command: Flash verify error</b>
\$1416	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$1417	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1418	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1419	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$141A	TPCMD PROGRAM <b>command: Flash program error</b>
\$141B	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$141D	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$141E	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$141F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1420	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$1421	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1422	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1423	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$1424	TPCMD MASSERASE <b>command: chip erase error</b>
\$1425	TPCMD RUN <b>command: execution error</b>
\$1426	TPEND <b>command: execution error</b>
\$1427	TPSTART <b>command: execution error</b>
\$1428	<b>Can't establish communication with the target device</b>
\$1429	TPCMD MASSERASE <b>command: invalid VDD value</b>
\$1430	<b>Target device ID is different than that specified by the TCSETDEV command</b>
\$1440	TPCMD READ <b>command: missing or invalid parameter</b>
\$1441	TPCMD READ <b>command: Flash read error</b>
\$1442	TPCMD READ <b>command: EEPROM read error</b>
\$1431	TCSETPAR RTSDOWN <b>command: missing, invalid or out of range parameter</b>
\$1432	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
<b>Freescal S08 - Specific Errors</b>	
\$1500	TCSETDEV <b>command: manufacturer not supported</b>
\$1501	TCSETDEV <b>command: algorithm not found on card</b>
\$1502	TCSETDEV <b>command: device not supported</b>
\$1503	TCSETDEV <b>command: corrupted algorithm file</b>
\$1504	TCSETDEV <b>command: internal hardware configuration error</b>
\$1510	TCSETPAR <b>command: parameter not supported</b>
\$1511	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$1512	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1513	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$1514	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$1515	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$1516	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1517	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$1518	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1519	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$151A	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$151B	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$151C	TCSETPAR BDM_ENTRY_MODE <b>command: missing, invalid or out of range parameter</b>
\$151D	TPSTART <b>command: command not preceded by TCSETPAR BDM_ENTRY_MODE command</b>
\$151E	TCSETPAR FLL_OSC <b>command: missing, invalid or out of range parameter</b>
\$151F	TPSTART <b>command: command not preceded by TCSETPAR FLL_OSC command</b>
\$1520	TPCMD <b>command: parameter not supported</b>
\$1521	TPCMD RUN <b>command: execution error</b>
\$1522	TPCMD IS_DEVICE_SECURED <b>command: execution error</b>
\$1523	TPCMD TRIM <b>command: missing, invalid or out of range parameter</b>
\$1524	TPCMD TRIM <b>command: trimming not supported by target device</b>
\$1525	TPCMD TRIM <b>command: execution error</b>
\$1526	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1527	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1528	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1529	TPCMD VERIFY <b>command: Flash verify error</b>
\$152A	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$152B	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$152C	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$152D	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$152E	TPCMD PROGRAM <b>command: Flash program error</b>
\$152F	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$1530	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$1531	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1532	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1533	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$1534	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1535	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1536	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$1537	TPSTART <b>command: execution error</b>
\$1538	TPEND <b>command: execution error</b>
\$1539	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$153A	TPCMD VERIFY <b>command: device is protected</b>

## Errors

Error #	Description
\$153B	TPCMD PROGRAM <b>command: device is protected</b>
\$153C	TPCMD BLANKCHECK <b>command: device is protected</b>
\$153D	TPCMD READ <b>command: missing or invalid parameter</b>
\$153E	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$153F	TPCMD READ <b>command: device is protected</b>
\$1540	TPCMD READ <b>command: read error</b>
\$1541	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$1542	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$1543	TPCMD DUMP <b>command: destination offset parameter out of range</b>
\$1544	TPCMD DUMP <b>command: Flash dump error</b>
\$1545	TPCMD DUMP <b>command: EEPROM dump error</b>
\$1546	TPCMD DUMP <b>command: device is protected</b>
\$1547	TPCMD DUMP <b>command: requires FlashRunner firmware version 01.03.00.00 or newer</b>
\$1548	TPCMD RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$1549	TPCMD PAGEERASE <b>command: Flash page erase error</b>
\$154A	TPCMD PAGEERASE <b>command: missing, invalid or out of range parameter</b>
\$154B	TPCMD PAGEERASE <b>command: missing, invalid or out of range parameter</b>
\$154C	TPCMD PAGEERASE <b>command: target protected</b>
\$154D	<b>Trim detection ICSC1 execution error</b>
\$154E	<b>Trim detection ICSC2 execution error</b>
\$154F	<b>Trim detection ICSSC execution error</b>
<b>Freescal RS08 - Specific Errors</b>	
\$1600	TCSETDEV <b>command: manufacturer not supported</b>
\$1601	TCSETDEV <b>command: algorithm not found on card</b>
\$1602	TCSETDEV <b>command: device not supported</b>
\$1603	TCSETDEV <b>command: corrupted algorithm file</b>
\$1604	TCSETDEV <b>command: internal hardware configuration error</b>
\$1610	TCSETPAR <b>command: parameter not supported</b>
\$1611	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$1612	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1613	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$1614	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1615	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$1616	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1617	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$1618	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1619	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$161A	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$161B	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>

Error #	Description
\$161C	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$161D	TCSETPAR BDM_ENTRY_MODE <b>command: missing, invalid or out of range parameter</b>
\$161E	TPSTART <b>command: command not preceded by TCSETPAR BDM_ENTRY_MODE command</b>
\$1620	TPCMD <b>command: parameter not supported</b>
\$1621	TPCMD RUN <b>command: execution error</b>
\$1622	TPCMD IS_DEVICE_SECURED <b>command: execution error</b>
\$1623	TPCMD TRIM <b>command: missing, invalid or out of range parameter</b>
\$1624	TPCMD TRIM <b>command: trimming not supported by target device</b>
\$1625	TPCMD TRIM <b>command: execution error</b>
\$1626	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1627	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1628	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1629	TPCMD VERIFY <b>command: Flash verify error</b>
\$162A	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$162B	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$162C	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$162D	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$162E	TPCMD PROGRAM <b>command: Flash program error</b>
\$162F	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$1630	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$1631	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1632	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1633	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$1634	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1635	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1636	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$1637	TPSTART <b>command: execution error</b>
\$1638	TPEND <b>command: execution error</b>
\$1639	TPCMD VERIFY <b>command: device is protected</b>
\$163A	TPCMD PROGRAM <b>command: device is protected</b>
\$163B	TPCMD BLANKCHECK <b>command: device is protected</b>
<b>Freescale S12 - Specific Errors</b>	
\$1700	TCSETDEV <b>command: manufacturer not supported</b>
\$1701	TCSETDEV <b>command: algorithm not found on card</b>
\$1702	TCSETDEV <b>command: device not supported</b>
\$1703	TCSETDEV <b>command: corrupted algorithm file</b>
\$1704	TCSETDEV <b>command: internal hardware configuration error</b>
\$1710	TCSETPAR <b>command: parameter not supported</b>

## Errors

Error #	Description
\$1711	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$1712	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1713	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1714	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1715	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1716	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1717	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$1718	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$171A	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$171B	TCSETPAR REFDIV <b>command: missing, invalid or out of range parameter</b>
\$171C	TCSETPAR SYNR <b>command: missing, invalid or out of range parameter</b>
\$171D	TPCMD <b>command: missing parameter</b>
\$171E	TPCMD <b>command: parameter not supported</b>
\$171F	TPCMD RUN <b>command: execution error</b>
\$1720	TPCMD UNSECURE <b>command: execution error</b>
\$1721	TPCMD MASSERASE <b>command: missing parameter</b>
\$1722	TPCMD MASSERASE <b>command: invalid parameter</b>
\$1723	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1724	TPCMD MASSERASE <b>command: device is protected</b>
\$1725	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$1726	TPCMD MASSERASE <b>command: device is protected</b>
\$1727	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$1728	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$1729	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$172A	TPCMD BLANKCHECK <b>command: device is protected</b>
\$172B	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$172C	TPCMD BLANKCHECK <b>command: device is protected</b>
\$172D	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$172E	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$172F	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$1730	TPCMD PROGRAM <b>command: Flash program error</b>
\$1731	TPCMD PROGRAM <b>command: device is protected</b>
\$1732	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$1733	TPCMD PROGRAM <b>command: device is protected</b>
\$1734	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1735	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1736	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1737	TPCMD VERIFY <b>command: Flash verify error</b>

Error #	Description
\$1738	TPCMD VERIFY <b>command: device is protected</b>
\$1739	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$173A	TPCMD VERIFY <b>command: device is protected</b>
\$173B	TPSTART <b>command: execution error</b>
\$173C	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$173D	TPEND <b>command: execution error</b>
\$173E	TPCMD READ <b>command: missing or invalid parameter</b>
\$173F	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$1740	TPCMD READ F <b>command: Flash read error</b>
\$1741	TPCMD READ F <b>command: device is protected</b>
\$1742	TPCMD READ E <b>command: EEPROM read error</b>
\$1743	TPCMD READ E <b>command: device is protected</b>
\$1744	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$1746	TPCMD PROGRAM <b>command: system program error</b>
\$1747	TPCMD PROGRAM <b>command: Device system secured</b>
\$1748	TPCMD PROGRAM <b>command: Emulated Eeprom program error</b>
\$1749	TPCMD PROGRAM <b>command: Device emulated Eeprom secured</b>
\$174A	TPCMD VERIFY <b>command: Emulated Eeprom verify error</b>
\$174B	TPCMD VERIFY <b>command: Device emulated Eeprom secured</b>
\$174C	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$174D	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$174E	TPCMD ERASE <b>command: Device flash memory secured</b>
\$174F	TPCMD ERASE <b>command: Flash program error</b>
\$174F	TPCMD ERASE <b>command: Eeprom memory program error</b>
\$174F	TPCMD ERASE <b>command: device Eeprom memory secured</b>
<b>Microchip PC10, PIC12 and PIC16 - Specific Errors</b>	
\$1800	TCSETDEV <b>command: manufacturer not supported</b>
\$1801	TCSETDEV <b>command: algorithm not found on card</b>
\$1802	TCSETDEV <b>command: device not supported</b>
\$1803	TCSETDEV <b>command: internal hardware configuration error</b>
\$1804	TCSETPAR <b>command: parameter not supported</b>
\$1805	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$1806	TPSTART <b>command: command not preceded by</b> TCSETPAR SCLK <b>command</b>
\$1807	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1808	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$1809	TCSETPAR VPP <b>command: missing, invalid or out of range parameter</b>
\$180A	TPSTART <b>command: command not preceded by</b> TCSETPAR VPP <b>command</b>
\$180B	TCSETPAR PROGMODE <b>command: missing or invalid parameter</b>

## Errors

Error #	Description
\$180C	TFSTART <b>command: command not preceded by</b> TCSETPAR PROGMODE <b>command</b>
\$180D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$180E	TFSTART <b>command: command not preceded by</b> TCSETPAR PWDOWN <b>command</b>
\$180F	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1810	TFSTART <b>command: command not preceded by</b> TCSETPAR PWUP <b>command</b>
\$1811	TPCMD <b>command: parameter not supported</b>
\$1812	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1813	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1814	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1815	TPCMD VERIFY <b>command: Flash verify error</b>
\$1816	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$1817	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1818	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1819	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$181A	TPCMD PROGRAM <b>command: Flash program error</b>
\$181B	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$181D	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$181E	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$181F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1820	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$1821	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1822	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1823	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$1824	TPCMD MASSERASE <b>command: chip erase error</b>
\$1825	TPCMD RUN <b>command: execution error</b>
\$1826	TFEND <b>command: execution error</b>
\$1827	TFSTART <b>command: execution error</b>
\$1828	TPCMD PROGRAM <b>command: invalid VDD parameter</b>
\$1829	TPCMD MASSERASE <b>command: invalid VDD parameter</b>
\$182A	TPCMD READ <b>command: missing or invalid parameter</b>
\$182B	TPCMD READ <b>command: Flash read error</b>
\$182C	TPCMD READ <b>command: EEPROM read error</b>
\$182D	TPCMD READ <b>command: OSCCAL read error</b>
\$182E	TPCMD READ <b>command: memory read error</b>
\$182F	TPCMD VERIFY <b>command: OSCCAL read error</b>
\$1830	TCSETPAR ISP_PGD <b>command: missing, invalid or out of range parameter</b>
\$1831	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>

**Atmel ARM7 - Specific Errors**



Error #	Description
\$1900	TCSETDEV <b>command: manufacturer not supported</b>
\$1901	TCSETDEV <b>command: algorithm not found on card</b>
\$1902	TCSETDEV <b>command: device not supported</b>
\$1903	TCSETDEV <b>command: corrupted algorithm file</b>
\$1904	TCSETDEV <b>command: corrupted algorithm file</b>
\$1905	TCSETDEV <b>command: internal hardware configuration error</b>
\$1906	TCSETPAR <b>command: parameter not supported</b>
\$1907	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1908	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$1909	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$190A	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$190B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$190C	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$190D	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$190E	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$190F	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1910	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$1912	TCSETPAR RSTDELAY <b>command: missing, invalid or out of range parameter</b>
\$1913	TPSTART <b>command: command not preceded by TCSETPAR RSTDELAY command</b>
\$1914	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$1916	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$191C	TPCMD <b>command: parameter not supported</b>
\$191D	TPCMD <b>command: command not preceded by TPCMD JTCLK command</b>
\$191E	TPCMD JTCLK <b>command: execution error</b>
\$191F	TPCMD JTCLK <b>command: missing or invalid parameter</b>
\$1920	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1921	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1922	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1923	TPCMD VERIFY <b>command: Flash verify error</b>
\$1927	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1928	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1929	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$192A	TPCMD PROGRAM <b>command: Flash program error</b>
\$192E	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$192F	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1930	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$1931	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1932	TPCMD MASSERASE <b>command: Flash masserase error</b>

## Errors

Error #	Description
\$1933	TPCMD SECTORERASE <b>command: missing or invalid parameter</b>
\$1934	TPCMD SECTORERASE <b>command: target start address parameter or length parameter out of range</b>
\$1935	TPCMD SECTORERASE <b>command: sector erase error</b>
\$1936	TPCMD RUN <b>command: execution error</b>
\$1937	TPCMD WRITE32 <b>command: missing or invalid parameter</b>
\$1938	TPCMD WRITE32 <b>command: execution error</b>
\$1939	TPCMD READ32 <b>command: missing or invalid parameter</b>
\$193A	TPCMD READ32 <b>command: execution error</b>
\$193C	TPSTART <b>command: execution error</b>
\$193D	TPCMD <b>command: execution error (could not communicate with target device)</b>
\$193E	TPEND <b>command: execution error</b>
\$1940	TPCMD WRITE16 <b>command: missing or invalid parameter</b>
\$1941	TPCMD WRITE16 <b>command: execution error</b>
\$1942	TPCMD READ16 <b>command: missing or invalid parameter</b>
\$1943	TPCMD READ16 <b>command: execution error</b>
\$1944	<b>Can't establish communication with the target device</b>
\$1945	TPCMD LOCK <b>command: missing or invalid parameter</b>
\$1946	TPCMD LOCK <b>command: target start address parameter or length parameter out of range</b>
\$1947	TPCMD LOCK <b>command: Lock flash memory error</b>
\$1948	TPCMD UNLOCK <b>command: missing or invalid parameter</b>
\$1949	TPCMD UNLOCK <b>command: target start address parameter or length parameter out of range</b>
\$194A	TPCMD UNLOCK <b>command: Unlock flash memory error</b>
<b>Nec 78K - Specific Errors</b>	
\$1A00	TCSETDEV <b>command: manufacturer not supported</b>
\$1A01	TCSETDEV <b>command: algorithm not found on card</b>
\$1A02	TCSETDEV <b>command: device not supported</b>
\$1A03	TCSETDEV <b>command: internal hardware configuration error</b>
\$1A04	TCSETPAR <b>command: parameter not supported</b>
\$1A05	TCSETPAR CMODE <b>command: missing parameter</b>
\$1A06	TCSETPAR CMODE <b>command: communication mode not supported by target device</b>
\$1A07	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>
\$1A08	TPSTART CMODE <b>command: invalid parameter</b>
\$1A09	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$1A0A	TCSETPAR CLKOUT <b>command: invalid frequency</b>
\$1A0B	TPSTART <b>command: command not preceded by TCSETPAR CLKOUT command</b>
\$1A0C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$1A0D	TPSTART <b>command: command not preceded by TCSETPAR FOSC command</b>
\$1A0E	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$1A0F	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$1A10	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1A11	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$1A12	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$1A13	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$1A14	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1A15	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$1A16	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1A17	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$1A18	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1A19	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$1A1A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1A1B	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$1A1C	TPCMD <b>command: parameter not supported</b>
\$1A1D	<b>Can't establish communication with the target device</b>
\$1A1E	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1A1F	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1A20	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1A21	TPCMD VERIFY <b>command: Flash verify error</b>
\$1A22	TPCMD VERIFY <b>command: operation not supported by target device</b>
\$1A23	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1A24	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1A25	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$1A26	TPCMD PROGRAM <b>command: Flash program error</b>
\$1A27	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$1A28	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1A29	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1A2A	TPCMD BLANKCHECK <b>command: operation not supported by target device</b>
\$1A2B	TPCMD MASSERASE <b>command: chip erase error</b>
\$1A2C	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$1A2D	TPCMD PROTECT <b>command: chip protection error</b>
\$1A2E	TPCMD PROTECT <b>command: operation not supported by target device</b>
\$1A2F	TPCMD BLOCKERASE <b>command: missing or invalid parameter</b>
\$1A30	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$1A31	TPCMD BLOCKERASE <b>command: block erase error</b>
\$1A32	TPCMD BLOCKERASE <b>command: operation not supported by target device</b>
\$1A33	TPCMD RUN <b>command: execution error</b>
\$1A34	TPSTART <b>command: execution error</b>

## Errors

Error #	Description
\$1A35	TPEND <b>command: could not power target off</b>
\$1A36	TPEND <b>command: execution error</b>
\$1A37	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$1A38	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$1A39	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$1A3A	TCSETPAR SIFU <b>command: serial input Pull-Up error</b>
\$1A3B	TCSETPAR SOPU <b>command: serial output Pull-Up error</b>
\$1A40	TCSETPAR VDD_AUX <b>command: command not allowed</b>
\$1A41	TCSETDEV <b>command: a newer version of FlashRunner firmware is required</b>
\$1A42	TPCMD GETSIGNATURE <b>command: get signature not supported</b>
\$1A43	TPCMD GETSIGNATURE <b>command: signature not OK</b>
\$1A44	TPCMD PROGRAMBYTE <b>command: execution error</b>
\$1A45	TPCMD PROGRAMBYTE <b>command: program byte not supported</b>
<b>Renesas H8S/H8SX/SuperH (Serial Boot Mode) - Specific Errors</b>	
\$1B00	TCSETDEV <b>command: manufacturer not supported</b>
\$1B01	TCSETDEV <b>command: algorithm not found on card</b>
\$1B02	TCSETDEV <b>command: device not supported</b>
\$1B03	TCSETDEV <b>command: corrupted algorithm file</b>
\$1B04	TCSETPAR <b>command: parameter not supported</b>
\$1B05	TCSETPAR CLKOUT <b>command: out of range parameter</b>
\$1B06	TCSETPAR CLKOUT <b>command: missing or invalid parameter</b>
\$1B07	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1B08	TPSTART <b>command: command not preceded by</b> TCSETPAR PWDOWN <b>command</b>
\$1B09	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1B0A	TPSTART <b>command: command not preceded by</b> TCSETPAR PWUP <b>command</b>
\$1B0B	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1B0C	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTDOWN <b>command</b>
\$1B0D	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1B0E	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTUP <b>command</b>
\$1B0F	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1B10	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$1B11	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$1B12	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD_AUX <b>command</b>
\$1B13	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$1B14	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$1B15	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$1B16	TPSTART <b>command: command not preceded by</b> TCSETPAR FOSC <b>command</b>
\$1B17	TCSETPAR CK_MODE <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$1B18	TPSTART <b>command: command not preceded by TCSETPAR CK_MODE command</b>
\$1B19	TCSETPAR CKM <b>command: missing, invalid or out of range parameter</b>
\$1B1A	TPSTART <b>command: command not preceded by TCSETPAR CKM command</b>
\$1B1B	TCSETPAR CKP <b>command: missing, invalid or out of range parameter</b>
\$1B1C	TPSTART <b>command: command not preceded by TCSETPAR CKP command</b>
\$1B1D	TCSETPAR CMODE <b>command: missing or invalid parameter</b>
\$1B1E	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>
\$1B1F	TCSETPAR DIO1 <b>command: missing or invalid parameter</b>
\$1B20	TPSTART <b>command: command not preceded by TCSETPAR DIO1 command</b>
\$1B21	TCSETPAR DIO2 <b>command: missing or invalid parameter</b>
\$1B22	TPSTART <b>command: command not preceded by TCSETPAR DIO2 command</b>
\$1B23	TCSETPAR DIO3 <b>command: missing or invalid parameter</b>
\$1B24	TPSTART <b>command: command not preceded by TCSETPAR DIO3 command</b>
\$1B25	TCSETPAR DIO4 <b>command: missing or invalid parameter</b>
\$1B26	TPSTART <b>command: command not preceded by TCSETPAR DIO4 command</b>
\$1B27	TPCMD <b>command: parameter not supported</b>
\$1B28	TPCMD RUN <b>command: execution error</b>
\$1B2A	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1B2C	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1B2D	TPCMD BLOCKERASE <b>command: missing parameter</b>
\$1B2E	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$1B2F	TPCMD BLOCKERASE <b>command: Flash block erase error</b>
\$1B30	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$1B31	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1B32	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1B33	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1B34	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1B35	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$1B36	TPCMD PROGRAM <b>command: Flash program error</b>
\$1B37	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1B38	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1B39	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1B3A	TPCMD VERIFY <b>command: Flash verify error</b>
\$2A3B	TPCMD READ <b>command: missing or invalid parameter</b>
\$2A3C	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2A3D	TPCMD READ <b>command: Flash read error</b>
\$1B3E	TPSTART <b>command: execution error</b>
\$1B3F	TPEND <b>command: execution error</b>

## Errors

Error #	Description
\$1B40	TPCMD READ_INFO <b>command: Information read error</b>
\$1B41	TCSETDEV <b>command: Algorithm not supported</b>
\$1B42	TCSETPAR TXDDRV <b>command: missing, invalid or out of range parameter</b>
\$1B44	TPSTART <b>command: command not preceded by</b> TCSETPAR KEY_CODE <b>command</b>
\$1B45	TCSETPAR KEY_CODE <b>command: missing, invalid or out of range parameter</b>
\$1B46	TPCMD ENABLE <b>command: missing, invalid or out of range parameter</b>
\$1B47	TPCMD DISABLE <b>command: missing, invalid or out of range parameter</b>
\$1B48	TPCMD ENABLE <b>command: enable lock bits error</b>
\$1B49	TPCMD DISABLE <b>command: disable lock bits error</b>
\$1B4A	TPCMD ENABLE <b>command: enable transmit error</b>
\$1B4B	TPCMD DISABLE <b>command: disable transmit error</b>
\$1B4C	TPCMD ENABLE <b>command: enable acknowledge error</b>
\$1B4E	TPCMD DISABLE <b>command: disable acknowledge error</b>
\$1B4F	TPCMD DUMP <b>command: flash dump error</b>
<b>Serial Memories SERMEM - Specific Errors</b>	
\$1C00	TCSETDEV <b>command: manufacturer not supported</b>
\$1C01	TCSETDEV <b>command: algorithm not found on card</b>
\$1C02	TCSETDEV <b>command: device not supported</b>
\$1C03	TCSETDEV <b>command: internal hardware configuration error</b>
\$1C04	TCSETPAR <b>command: parameter not supported</b>
\$1C05	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$1C06	TPSTART <b>command: command not preceded by</b> TCSETPAR SCLK <b>command</b>
\$1C07	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1C08	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$1C09	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$1C0B	TCSETPAR DATA_WIDTH <b>command: missing, invalid or out of range parameter</b>
\$1C0C	TPSTART <b>command: command not preceded by</b> TCSETPAR DATA_WIDTH <b>command</b>
\$1C0D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1C0E	TPSTART <b>command: command not preceded by</b> TCSETPAR PWDOWN <b>command</b>
\$1C0F	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1C10	TPSTART <b>command: command not preceded by</b> TCSETPAR PWUP <b>command</b>
\$1C11	TPCMD <b>command: parameter not supported</b>
\$1C12	<b>Can't establish communication with the target device</b>
\$1C13	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1C14	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1C15	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1C16	TPCMD VERIFY <b>command: memory data verify error</b>
\$1C17	TPCMD VERIFY <b>command: status register verify error</b>

Error #	Description
\$1C18	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1C19	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1C1A	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$1C1B	TPCMD PROGRAM <b>command: memory data program error</b>
\$1C1C	TPCMD PROGRAM <b>command: status register program error</b>
\$1C1D	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$1C1E	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1C1F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1C20	TPCMD READ <b>command: missing or invalid parameter</b>
\$1C21	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$1C22	TPCMD READ <b>command: memory data read error</b>
\$1C23	TPCMD READ <b>command: status register read error</b>
\$1C24	TPCMD CHIP_ERASE <b>command: chip erase error</b>
\$1C25	TPCMD CHIP_ERASE <b>command: command not supported</b>
\$1C26	TPCMD SECTOR_ERASE <b>command: missing or invalid parameter</b>
\$1C27	TPCMD SECTOR_ERASE <b>command: sector erase error</b>
\$1C28	TPCMD SECTOR_ERASE <b>command: command not supported</b>
\$1C29	TPCMD BLOCK_ERASE <b>command: missing or invalid parameter</b>
\$1C2A	TPCMD BLOCK_ERASE <b>command: chip erase error</b>
\$1C31	TPCMD START <b>command: execution error</b>
\$1C33	TPCMD END <b>command: execution error</b>
\$1C34	TCSETPAR DEVICE_ADDR <b>command: missing, invalid or out of range parameter</b>
\$1C35	TPSTART <b>command: command not preceded by TCSETPAR DEVICE_ADDR command</b>
\$1C36	TPCMD DATA_WIDTH <b>command: command not supported by the selected device</b>
\$1C37	TPCMD DEVICE_ADDR <b>command: command not supported by the selected device</b>
\$1C38	TPCMD ENABLE <b>command: missing or invalid parameter</b>
\$1C39	TPCMD ENABLE <b>command: execution error</b>
\$1C3A	TPCMD ENABLE <b>command: command not supported by the selected device</b>
\$1C3B	TPCMD DISABLE <b>command: missing or invalid parameter</b>
\$1C3C	TPCMD DISABLE <b>command: execution error</b>
\$1C3D	TPCMD DISABLE <b>command: command not supported by the selected device</b>
\$1C3E	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$1C3F	TPCMD ERASE <b>command: execution error</b>
\$1C40	TPCMD ERASE <b>command: command not supported by the selected device</b>
\$1C41	TCSETPAR PAGESIZE <b>command: missing, invalid or out of range parameter</b>
\$1C42	TCSETPAR PAGESIZE <b>command: command not supported by the selected device</b>
\$1C43	TPSTART <b>command: command not preceded by TCSETPAR PAGESIZE command</b>
\$1C44	TPCMD BLOCKERASE <b>command: command not supported by the selected device</b>

## Errors

Error #	Description
\$1C45	<b>TPCMD PROTECT</b> command: missing or invalid parameter
\$1C46	<b>TPCMD PROTECT</b> command: execution error
\$1C47	<b>TPCMD PROTECT</b> command: command not supported by the selected device
\$1C48	<b>TPCMD UNPROTECT</b> command: missing or invalid parameter
\$1C49	<b>TPCMD UNPROTECT</b> command: execution error
\$1C4A	<b>TPCMD UNPROTECT</b> command: command not supported by the selected device
\$1C4B	<b>TPCMD PROGRAM Info Page</b> command: execution error
\$1C4C	<b>TPCMD READ Lock Register</b> command: execution error
\$1C4D	<b>TPCMD PROGRAM Lock Register</b> command: execution error
\$1C4E	<b>TCSETPAR SKIP_IDCODE_CHECK</b> : missing or invalid parameter
\$1C4F	<b>TCSETDEV</b> command: the current FlashRunner firmware version doesn't support the specified Programming Algorithm version
\$2379	<b>TPCMD SER_NUMBERING</b> command: missing or invalid parameter
\$237A	<b>TPCMD SER_NUMBERING</b> command: execution error
\$23A0	<b>TPCMD DUMP</b> command: algorithm not supported
\$23A1	<b>TPCMD DUMP</b> command: missing or invalid parameter
\$23A2	<b>TPCMD DUMP</b> command: destination missing or invalid parameter
\$23A3	<b>TPCMD DUMP</b> command: missing, invalid or out of range parameter
\$23A4	<b>TPCMD DUMP</b> command: execution error
\$23A5	<b>TPCMD DUMP</b> command: binary file write byte error
\$23A6	<b>TPCMD DUMP</b> command: binary file write word error
<b>Freescale MCF52 MCF_A - Specific Errors</b>	
\$1D00	<b>TCSETDEV</b> command: manufacturer not supported
\$1D01	<b>TCSETDEV</b> command: algorithm not found on card
\$1D02	<b>TCSETDEV</b> command: device not supported
\$1D03	<b>TCSETDEV</b> command: internal hardware configuration error
\$1D04	<b>TCSETDEV</b> command: corrupted algorithm file
\$1D05	<b>TCSETPAR</b> command: parameter not supported
\$1D06	<b>TCSETPAR CLKOUT</b> command: missing, invalid or out of range parameter
\$1D07	<b>TCSETPAR RSTDOWN</b> command: missing, invalid or out of range parameter
\$1D08	<b>TCSETPAR RSTUP</b> command: missing, invalid or out of range parameter
\$1D09	<b>TCSETPAR PWDOWN</b> command: missing, invalid or out of range parameter
\$1D0A	<b>TCSETPAR PWUP</b> command: missing, invalid or out of range parameter
\$1D0B	<b>TCSETPAR VDD</b> command: missing, invalid or out of range parameter
\$1D0C	<b>TCSETPAR VDD_AUX</b> command: missing, invalid or out of range parameter
\$1D0D	<b>TCSETPAR FOSC</b> command: missing, invalid or out of range parameter
\$1D0E	<b>TCSETPAR PLLFREQ</b> command: missing, invalid or out of range parameter
\$1D0F	<b>TCSETPAR SCLK</b> command: missing, invalid or out of range parameter
\$1D10	<b>TCSETPAR SCLK</b> command: FOSC, PLLFREQ or PROG_MODE parameters have not been set



Error #	Description
\$1D11	TCSETPAR PROG_MODE <b>command: missing, invalid or out of range parameter</b>
\$1D12	TPCMD <b>command: missing parameter</b>
\$1D13	TPCMD <b>command: parameter not supported</b>
\$1D14	TPCMD RUN <b>command: execution error</b>
\$1D15	TPCMD UNSECURE <b>command: execution error</b>
\$1D16	TPCMD MASSERASE <b>command: missing parameter</b>
\$1D17	TPCMD MASSERASE <b>command: invalid parameter</b>
\$1D18	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1D19	TPCMD MASSERASE <b>command: device is protected</b>
\$1D1A	TPCMD PAGEERASE <b>command: missing parameter</b>
\$1D1B	TPCMD PAGEERASE <b>command: invalid parameter</b>
\$1D1C	TPCMD PAGEERASE <b>command: Flash page erase error</b>
\$1D1D	TPCMD PAGEERASE <b>command: device is protected</b>
\$1D1E	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$1D1F	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$1D20	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1D21	TPCMD BLANKCHECK <b>command: device is protected</b>
\$1D22	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1D23	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1D24	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$1D25	TPCMD PROGRAM <b>command: Flash program error</b>
\$1D26	TPCMD PROGRAM <b>command: device is protected</b>
\$1D27	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1D28	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1D29	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1D2A	TPCMD VERIFY <b>command: Flash verify error</b>
\$1D2B	TPCMD VERIFY <b>command: device is protected</b>
\$1D2C	TPSTART <b>command: execution error</b>
\$1D2D	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$1D2E	TPSTART <b>command: algorithm not found on card</b>
\$1D2F	TPSTART <b>command: internal hardware configuration error</b>
\$1D30	TPSTART <b>command: corrupted algorithm file</b>
\$1D31	TPEND <b>command: execution error</b>
<b>Fujitsu MB90 FJTS_A - Specific Errors</b>	
\$1E00	TCSETDEV <b>command: manufacturer not supported</b>
\$1E01	TCSETDEV <b>command: algorithm not found on card</b>
\$1E02	TCSETDEV <b>command: device not supported</b>
\$1E03	TCSETDEV <b>command: corrupted algorithm file</b>

## Errors

Error #	Description
\$1E04	TCSETDEV <b>command: internal hardware configuration error</b>
\$1E05	TCSETPAR <b>command: parameter not supported</b>
\$1E06	TCSETPAR CLKOUT <b>command: missing or invalid parameter</b>
\$1E07	TCSETPAR CLKOUT <b>command: out of range parameter</b>
\$1E08	TPSTART <b>command: command not preceded by</b> TCSETPAR CLKOUT <b>command</b>
\$1E09	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$1E0A	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$1E0B	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$1E0C	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD_AUX <b>command</b>
\$1E0D	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1E0E	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTDOWN <b>command</b>
\$1E0F	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1E10	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTUP <b>command</b>
\$1E11	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1E12	TPSTART <b>command: command not preceded by</b> TCSETPAR PWDOWN <b>command</b>
\$1E13	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1E14	TPSTART <b>command: command not preceded by</b> TCSETPAR PWUP <b>command</b>
\$1E15	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$1E16	TPSTART <b>command: command not preceded by</b> TCSETPAR FOSC <b>command</b>
\$1E17	TCSETPAR CMODE <b>command: missing parameter</b>
\$1E18	TCSETPAR CMODE <b>command: invalid parameter</b>
\$1E19	TPSTART <b>command: command not preceded by</b> TCSETPAR CMODE <b>command</b>
\$1E1A	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$1E1B	TPSTART <b>command: command not preceded by</b> TCSETPAR SCLK <b>command</b>
\$1E1C	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$1E1D	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$1E1E	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$1E1F	TPSTART <b>command: command not preceded by</b> TCSETPAR PLLFREQ <b>command</b>
\$1E20	TCSETPAR PLLFREQ <b>command: command must precede the</b> TCSETPAR BAUDRATE <b>command</b>
\$1E21	TCSETPAR PLLFREQ <b>command: command not preceded by the</b> TCSETPAR FOSC <b>command</b>
\$1E22	TCSETPAR BAUDRATE <b>command: baud rate not valid given the FOSC/PLLFREQ combination, or FlashRunner can't generate the specified baud rate within a working range</b>
\$1E23	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$1E24	TCSETPAR <b>command: invalid device family</b>
\$1E30	TPCMD <b>command: missing or invalid range parameter</b>
\$1E00	TPCMD <b>command: missing parameter</b>
\$1E01	TPCMD <b>command: invalid command</b>
\$1E02	TPCMD <b>command: entry error</b>
\$1E03	TPCMD MASSERASE <b>command: missing parameter</b>

Error #	Description
\$1E04	TPCMD MASSERASE <b>command: invalid parameter</b>
\$1E05	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$1E09	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$1E0A	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$1E0B	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$1E0F	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1E10	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1E11	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$1E12	TPCMD PROGRAM <b>command: Flash program error</b>
\$1E16	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1E17	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1E18	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1E19	TPCMD VERIFY <b>command: Flash verify error</b>
\$1E1A	TPCMD READ <b>command: missing or invalid parameter</b>
\$1E1B	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$1E1C	TPCMD READ <b>command: source offset parameter out of range</b>
\$1E1D	TPCMD READ <b>command: Flash read error</b>
\$1E20	TPCMD RUN <b>command: execution error</b>
\$1E28	TPSTART <b>command: execution error</b>
\$1E29	TPEND <b>command: execution error</b>
\$1E2F	TCSETDEV <b>command: old OS verision</b>
<b>Renesas M16C/M32C/R8C/R32C RNSS_B - Specific Errors</b>	
\$1F00	TCSETDEV <b>command: manufacturer not supported</b>
\$1F01	TCSETDEV <b>command: algorithm not found on card</b>
\$1F02	TCSETDEV <b>command: device not supported</b>
\$1F03	TCSETDEV <b>command: corrupted algorithm file</b>
\$1F04	TCSETPAR <b>command: parameter not supported</b>
\$1F05	TCSETPAR CLKOUT <b>command: out of range parameter</b>
\$1F06	TCSETPAR CLKOUT <b>command: missing or invalid parameter</b>
\$1F07	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$1F08	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$1F09	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$1FOA	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$1FOB	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$1FOC	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$1FOD	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$1FOE	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$1FOF	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>

## Errors

Error #	Description
\$1F10	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$1F11	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD_AUX <b>command</b>
\$1F12	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$1F17	TPSTART <b>command: command not preceded by</b> TCSETPAR ID_CODE <b>command</b>
\$1F18	TCSETPAR ID_CODE <b>command: missing, invalid or out of range parameter</b>
\$1F19	TPSTART <b>command: command not preceded by</b> TCSETPAR CMODE <b>command</b>
\$1F1A	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$1F1B	TPSTART <b>command: command not preceded by</b> TCSETPAR SCLK <b>command</b>
\$1F1C	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$1F1D	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$1F1E	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$1F1F	TPCMD <b>command: missing parameter</b>
\$1F20	TPCMD <b>command: parameter not supported</b>
\$1F21	TPCMD RUN <b>command: execution error</b>
\$1F27	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$1F28	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$1F29	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$1F2A	TPCMD VERIFY <b>command: User Flash verify error</b>
\$1F2B	TPCMD VERIFY <b>command: Data Flash verify error</b>
\$1F2C	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$1F2D	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$1F2E	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$1F2F	TPCMD PROGRAM <b>command: User Flash program error</b>
\$1F30	TPCMD PROGRAM <b>command: Data Flash program error</b>
\$1F31	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$1F32	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$1F33	TPCMD BLANKCHECK <b>command: User Flash blank check error</b>
\$1F34	TPCMD BLANKCHECK <b>command: Data Flash blank check error</b>
\$1F39	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$1F3A	TPCMD MASSERASE <b>command: User Flash mass erase error</b>
\$1F3B	TPCMD MASSERASE <b>command: Data Flash mass erase error</b>
\$1F3C	TPCMD BLOCKERASE <b>command: missing parameter</b>
\$1F3D	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$1F3E	TPCMD BLOCKERASE <b>command: User Flash block erase error</b>
\$1F3F	TPCMD BLOCKERASE <b>command: Data Flash block erase error</b>
\$1F40	TPSTART <b>command: execution error</b>
\$1F41	TPEND <b>command: execution error</b>
\$1F42	TPCMD LOCK <b>command: missing parameter</b>

Error #	Description
\$1F43	TPCMD LOCK <b>command: target start address parameter or length parameter out of range</b>
\$1F44	TPCMD LOCK <b>command: lock error</b>
\$1F45	TPCMD UNLOCK <b>command: unlock error</b>
\$1F49	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$1F4A	TCSETPAR E2DATA_ECC <b>command: missing, invalid or out of range parameter</b>
\$1F4B	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$1F4C	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$1F4D	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$1F4E	TPCMD DUMP <b>command: Flash dump error</b>
\$1F4F	TPCMD DUMP <b>command: Eeprom Flash dump error</b>
<b>Nec V850 - Specific Errors</b>	
\$2000	TCSETDEV <b>command: manufacturer not supported</b>
\$2001	TCSETDEV <b>command: algorithm not found on card</b>
\$2002	TCSETDEV <b>command: device not supported</b>
\$2003	TCSETDEV <b>command: internal hardware configuration error</b>
\$2004	TCSETPAR <b>command: parameter not supported</b>
\$2005	TCSETPAR CMODE <b>command: missing parameter</b>
\$2006	TCSETPAR CMODE <b>command: communication mode not supported by target device</b>
\$2007	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>
\$2008	TPSTART CMODE <b>command: invalid parameter</b>
\$2009	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$200A	TCSETPAR CLKOUT <b>command: invalid frequency</b>
\$200B	TPSTART <b>command: command not preceded by TCSETPAR CLKOUT command</b>
\$200C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$200D	TPSTART <b>command: command not preceded by TCSETPAR FOSC command</b>
\$200E	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$200F	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$2010	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2011	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2012	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2013	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$2014	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2015	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$2016	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2017	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$2018	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2019	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$201A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>

## Errors

Error #	Description
\$201B	TPSTART <b>command: command not preceded by</b> TCSETPAR FWUP <b>command</b>
\$201C	TPCMD <b>command: parameter not supported</b>
\$201D	<b>Can't establish communication with the target device</b>
\$201E	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$201F	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2020	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2021	TPCMD VERIFY <b>command: Flash verify error</b>
\$2023	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2024	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2025	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2026	TPCMD PROGRAM <b>command: Flash program error</b>
\$2027	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$2028	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$2029	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$202A	TPCMD BLANKCHECK <b>command: operation not supported by target device</b>
\$202B	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$202C	TPCMD PROTECT <b>command: missing, invalid or out of range parameter(s)</b>
\$202D	TPCMD PROTECT <b>command: chip protection error</b>
\$202E	TPCMD PROTECT <b>command: operation not supported by target device</b>
\$202F	TPCMD BLOCKERASE <b>command: missing or invalid parameter</b>
\$2030	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$2031	TPCMD BLOCKERASE <b>command: Flash block erase error</b>
\$2032	TPCMD BLOCKERASE <b>command: operation not supported by target device</b>
\$2033	TPCMD RUN <b>command: execution error</b>
\$2034	TPSTART <b>command: execution error</b>
\$2035	TPEND <b>command: execution error</b>
\$2036	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$2037	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$2038	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$2039	TPCMD READ <b>command: missing or invalid parameter</b>
\$203A	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$203B	TPCMD READ <b>command: operation not supported by target device</b>
\$203C	TPCMD READ <b>command: read error</b>
\$203D	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$203E	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$203F	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$2040	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$2041	TPCMD MASSERASE <b>command: missing or invalid parameter</b>

Error #	Description
\$2042	TPCMD BLOCKERASE <b>command: EEPROM block erase error</b>
\$2043	TPCMD READ <b>command: EEPROM read error</b>
\$2044	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$2045	TCSETPAR SIPU <b>command: missing or invalid parameter</b>
\$2046	TCSETPAR SOPU <b>command: missing or invalid parameter</b>
\$204F	TCSETDEV <b>command: old OS version</b>
<b>STMicroelectronics ST10, Infineon C166 and Infineon XC166 STM_IN_A - Specific Errors</b>	
\$2101	TCSETDEV <b>command: algorithm not found on card</b>
\$2102	TCSETDEV <b>command: device not supported</b>
\$2103	TCSETDEV <b>command: internal hardware configuration error</b>
\$2104	TCSETDEV <b>command: corrupted algorithm file</b>
\$2105	TCSETPAR <b>command: parameter not supported</b>
\$2106	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2107	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2108	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2109	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$210A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$210B	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$210C	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$210D	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$210E	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$210F	TCSETPAR BAUDRATE <b>command: FCPU parameter has not been set</b>
\$2110	TPCMD <b>command: missing parameter</b>
\$2111	TPCMD <b>command: parameter not supported</b>
\$2112	TPCMD RUN <b>command: execution error</b>
\$2113	TPCMD MASSERASE <b>command: missing parameter</b>
\$2114	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2115	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$2116	TPCMD MASSERASE <b>command: device is protected</b>
\$2117	TPCMD SECTORERASE <b>command: missing parameter</b>
\$2118	TPCMD SECTORERASE <b>command: invalid parameter</b>
\$2119	TPCMD SECTORERASE <b>command: Flash sector erase error</b>
\$211A	TPCMD SECTORERASE <b>command: device is protected</b>
\$211B	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$211C	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$211D	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$211E	TPCMD BLANKCHECK <b>command: device is protected</b>
\$211F	TPCMD PROGRAM <b>command: missing or invalid parameter</b>

## Errors

Error #	Description
\$2120	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2121	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2122	TPCMD PROGRAM <b>command: Flash program error</b>
\$2123	TPCMD PROGRAM <b>command: device is protected</b>
\$2124	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2125	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2126	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2127	TPCMD VERIFY <b>command: Flash verify error</b>
\$2128	TPCMD VERIFY <b>command: device is protected</b>
\$2129	TPSTART <b>command: execution error</b>
\$212A	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$212B	TPEND <b>command: execution error</b>
\$212C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$212D	TCSETPAR PLLMUL <b>command: missing, invalid or out of range parameter</b>
\$212E	TCSETPAR PLLIDIV <b>command: missing, invalid or out of range parameter</b>
\$212F	TCSETPAR PLODIV <b>command: missing, invalid or out of range parameter</b>
\$2130	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$2131	TCSETPAR PLLFREQ <b>command: FCPU, FOSC, PLLMUL, PLLIDIV or PLODIV parameters have not been set</b>
\$2132	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$2133	TPCMD PROTECT <b>command: execution error</b>
\$2134	TPCMD UNPROTECT <b>command: missing or invalid parameter</b>
\$2135	TPCMD UNPROTECT <b>command: execution error</b>
\$2136	TCSETPAR RSTDRV <b>command: missing or invalid command option</b>
\$2137	TCSETPAR CMODE <b>command: missing or invalid command option</b>
\$2138	TCSETPAR BAUDRATE <b>command: UART baud rate parameter not used in JTAG communication mode</b>
\$2139	TCSETPAR JTCLK <b>command: missing or invalid command option</b>
\$213A	TCSETPAR JTCLK <b>command: JTAG clock frequency parameter not used in UART communication mode</b>
\$213B	TCSETPAR JTCLK <b>command: FCPU and CMODE parameters have not been set</b>
\$213C	TPSTART <b>command: internal hardware configuration error</b>
<b>Texas Instruments MSP430 TL_A -Specific Errors</b>	
\$2200	TCSETDEV <b>command: manufacturer not supported</b>
\$2201	TCSETDEV <b>command: algorithm not found on card</b>
\$2202	TCSETDEV <b>command: device not supported</b>
\$2203	TCSETDEV <b>command: corrupted algorithm file</b>
\$2204	TCSETPAR <b>command: parameter not supported</b>
\$2205	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2206	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2207	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>



Error #	Description
\$2208	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$2209	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$220A	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$220B	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$220C	TPSTART <b>command: command not preceded by TCSETPAR JTCLK command</b>
\$2220	TPCMD <b>command: parameter not supported</b>
\$2221	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2222	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2223	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2224	TPCMD VERIFY <b>command: Flash verify error</b>
\$2225	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2226	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2227	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2228	TPCMD PROGRAM <b>command: Flash program error</b>
\$2229	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$222A	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$222B	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$222C	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$222D	TPCMD MASSERASE <b>command: masserase error</b>
\$222E	TPCMD MASSERASE <b>command: main memory masserase error</b>
\$222F	TPCMD SEGMENTERASE <b>command: missing or invalid parameter</b>
\$2230	TPCMD SEGMENTERASE <b>command: segment erase error</b>
\$2231	TPCMD SEGMENTERASE <b>command: target start address parameter or length parameter out of range</b>
\$2232	TPCMD <b>command: invalid VDD parameter during program or erase</b>
\$2233	TPCMD READ <b>command: missing or invalid parameter or command option</b>
\$2234	TPCMD READ F <b>command: target start address parameter or length parameter out of range</b>
\$2235	TPCMD READ F <b>command: Flash memory read error</b>
\$2236	TPCMD RUN <b>command: execution error</b>
\$2237	TPCMD WRITE8 <b>command: missing or invalid parameter</b>
\$2238	TPCMD WRITE8 <b>command: execution error</b>
\$2239	TPCMD READ8 <b>command: missing or invalid parameter</b>
\$223A	TPCMD READ8 <b>command: execution error</b>
\$223B	TPCMD WRITE16 <b>command: missing or invalid parameter</b>
\$223C	TPCMD WRITE16 <b>command: execution error</b>
\$223D	TPCMD READ16 <b>command: missing or invalid parameter</b>
\$223E	TPCMD READ16 <b>command: execution error</b>
\$223F	TPCMD BLOWFUSE <b>command: execution error</b>
\$2240	TPSTART <b>command: execution error</b>

## Errors

Error #	Description
\$2241	TFEND <b>command: execution error</b>
\$2244	<b>Can't establish communication with the target device</b>
\$2245	<b>Can't establish communication with the target device</b>
<b>Silicon Labs C8051F SLB_A - Specific Errors</b>	
\$2400	TCSETDEV <b>command: manufacturer not supported</b>
\$2401	TCSETDEV <b>command: algorithm not found on card</b>
\$2402	TCSETDEV <b>command: device not supported</b>
\$2403	TCSETDEV <b>command: corrupted algorithm file</b>
\$2404	TCSETPAR <b>command: parameter not supported</b>
\$2405	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2406	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2407	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2408	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$2409	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$240A	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$240B	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$240C	TPSTART <b>command: command not preceded by TCSETPAR JTCLK command</b>
\$240D	TCSETPAR FCU <b>command: missing, invalid or out of range parameter</b>
\$240E	TPSTART <b>command: command not preceded by TCSETPAR FCPU command</b>
\$240F	TCSETPAR RSTDRV <b>command: missing or invalid command option</b>
\$2410	TCSETPAR C2CLK <b>command: missing, invalid or out of range parameter</b>
\$2411	TPSTART <b>command: command not preceded by TCSETPAR C2CLK command</b>
\$2412	TCSETPAR CMODE <b>command: missing or invalid command option</b>
\$2413	TCSETDEV <b>command: programming algorithm requires a newer FlashRunner firmware</b>
\$2414	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2415	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2416	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2417	TPCMD WRITE8 <b>command: missing or invalid parameter</b>
\$2418	TPCMD READ8 <b>command: missing or invalid parameter</b>
\$2420	TPCMD <b>command: parameter not supported</b>
\$2421	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2422	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2423	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2424	TPCMD VERIFY <b>command: verify error</b>
\$2425	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2426	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2427	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2428	TPCMD PROGRAM <b>command: program error</b>
\$2429	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>

Error #	Description
\$242A	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$242B	TPCMD BLANKCHECK <b>command: blankcheck error</b>
\$242C	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$242D	TPCMD MASSERASE <b>command: masserase error</b>
\$242E	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$242F	TPCMD ERASE <b>command: erase error</b>
\$2430	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$2431	TPCMD RUN <b>command: execution error</b>
\$2432	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$2433	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$2434	TPCMD ERASE <b>command: erase error</b>
\$2440	TPSTART <b>command: execution error</b>
\$2441	TPEND <b>command: execution error</b>
\$2442	<b>Can't establish communication with the target device</b>
<b>Renesas H8/300H RNSS_C - Specific Errors</b>	
\$2500	TCSETDEV <b>command: manufacturer not supported</b>
\$2501	TCSETDEV <b>command: algorithm not found on card</b>
\$2502	TCSETDEV <b>command: device not supported</b>
\$2503	TCSETDEV <b>command: corrupted algorithm file</b>
\$2504	TCSETPAR <b>command: parameter not supported</b>
\$2505	TCSETPAR CLKOUT <b>command: out of range parameter</b>
\$2506	TCSETPAR CLKOUT <b>command: missing or invalid parameter</b>
\$2507	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2508	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$2509	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$250A	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$250B	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$250C	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$250D	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$250E	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$250F	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2510	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2511	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$2512	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2513	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$2514	TPSTART <b>command: command not preceded by TCSETPAR FOSC command</b>
\$2515	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>
\$2516	TCSETPAR CMODE <b>command: missing or invalid parameter</b>

## Errors

Error #	Description
\$2517	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$2518	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$2519	TPCMD <b>command: missing parameter</b>
\$251A	TPCMD <b>command: parameter not supported</b>
\$251B	TPCMD RUN <b>command: execution error</b>
\$251C	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$251D	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$251E	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$251F	TPCMD VERIFY <b>command: Flash verify error</b>
\$2520	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$2521	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2522	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2523	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2524	TPCMD PROGRAM <b>command: Flash program error</b>
\$2525	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$2526	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$2527	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$2528	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$2529	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$252A	TPCMD MASSErase <b>command: missing or invalid parameter</b>
\$252B	TPCMD MASSErase <b>command: Flash mass erase error</b>
\$252C	TPCMD MASSErase <b>command: EEPROM mass erase error</b>
\$252D	TPSTART <b>command: execution error</b>
\$252E	TPEND <b>command: execution error</b>
\$252F	TPCMD READ <b>command: missing or invalid parameter</b>
\$2530	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2531	TPCMD READ <b>command: Flash read error</b>
\$2532	TPCMD READ <b>command: Eeprom read error</b>
\$2533	TCSETPAR CKM <b>command: missing, invalid or out of range parameter</b>
\$2534	TCSETPAR FOSC_SOURCE <b>command: missing, invalid or out of range parameter</b>
\$2535	TCSETDEV <b>command: Algorithm not supported</b>
\$2536	TCSETPAR RXDDRV <b>command: missing, invalid or out of range parameter</b>
\$2537	TPCMD <b>command: invalid Command</b>
<b>Microchip PIC24, dsPIC30 and dsPIC33 MCHP_A - Specific Errors</b>	
\$2600	TCSETDEV <b>command: manufacturer not supported</b>
\$2601	TCSETDEV <b>command: algorithm not found on card</b>
\$2602	TCSETDEV <b>command: device not supported</b>
\$2603	TCSETDEV <b>command: internal hardware configuration error</b>

Error #	Description
\$2604	TCSETPAR <b>command: parameter not supported</b>
\$2605	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$2606	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$2607	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2608	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2609	TCSETPAR VFP <b>command: missing, invalid or out of range parameter</b>
\$260A	TPSTART <b>command: command not preceded by TCSETPAR VFP command</b>
\$260D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$260E	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$260F	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2610	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$2611	TPCMD <b>command: parameter not supported</b>
\$2612	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$2613	TPCMD MASSERASE <b>command: invalid VDD value</b>
\$2614	TPCMD MASSERASE <b>command: chip erase error</b>
\$2615	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2616	TPCMD PROGRAM F <b>command: target start address parameter or length parameter out of range</b>
\$2617	TPCMD PROGRAM E <b>command: target start address parameter or length parameter out of range</b>
\$2618	TPCMD PROGRAM F <b>command: source offset parameter out of range</b>
\$2619	TPCMD PROGRAM E <b>command: source offset parameter out of range</b>
\$261A	TPCMD PROGRAM F <b>command: Flash program error</b>
\$261B	TPCMD PROGRAM E <b>command: EEPROM program error</b>
\$261C	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$261D	TPCMD VERIFY F <b>command: target start address parameter or length parameter out of range</b>
\$261E	TPCMD VERIFY E <b>command: target start address parameter or length parameter out of range</b>
\$261F	TPCMD VERIFY F <b>command: source offset parameter out of range</b>
\$2620	TPCMD VERIFY E <b>command: source offset parameter out of range</b>
\$2621	TPCMD VERIFY F <b>command: Flash verify error</b>
\$2622	TPCMD VERIFY E <b>command: EEPROM verify error</b>
\$2623	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$2624	TPCMD BLANKCHECK F <b>command: target start address parameter or length parameter out of range</b>
\$2625	TPCMD BLANKCHECK E <b>command: target start address parameter or length parameter out of range</b>
\$2626	TPCMD BLANKCHECK F <b>command: Flash blank check error</b>
\$2627	TPCMD BLANKCHECK E <b>command: EEPROM blank check error</b>
\$2628	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$2629	TPCMD ERASE F <b>command: target start address parameter or length parameter out of range</b>
\$262A	TPCMD ERASE F <b>command: Flash erase error</b>
\$262B	TPCMD ERASE E <b>command: target start address parameter or length parameter out of range</b>

## Errors

Error #	Description
\$262C	TPCMD ERASE E <b>command: EEPROM erase error</b>
\$2630	TPCMD READ <b>command: missing or invalid parameter</b>
\$2631	TPCMD READ <b>command: Flash read error</b>
\$2632	TPCMD READ <b>command: EEPROM read error</b>
\$2633	TPCMD RUN <b>command: execution error</b>
\$2634	TPEND <b>command: execution error</b>
\$2635	TPSTART <b>command: execution error</b>
\$2636	<b>Can't establish communication with the target device</b>
\$2637	<b>Target device ID is different than that specified by the TCSETDEV command</b>
<b>Infineon XC800 INF_A - Specific Errors</b>	
\$2700	TCSETDEV <b>command: manufacturer not supported</b>
\$2701	TCSETDEV <b>command: algorithm not found on card</b>
\$2702	TCSETDEV <b>command: device not supported</b>
\$2703	TCSETDEV <b>command: internal hardware configuration error</b>
\$2704	TCSETDEV <b>command: corrupted algorithm file</b>
\$2705	TCSETPAR <b>command: parameter not supported</b>
\$2706	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2707	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2708	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2709	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$270A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$270B	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$270C	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$270D	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$270E	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$270F	TPCMD <b>command: missing parameter</b>
\$2710	TPCMD <b>command: parameter not supported</b>
\$2711	TPCMD RUN <b>command: execution error</b>
\$2712	TPCMD MASSERASE <b>command: missing parameter</b>
\$2713	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2714	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$2715	TPCMD MASSERASE <b>command: device is protected</b>
\$2716	TPCMD SECTORERASE <b>command: missing parameter</b>
\$2717	TPCMD SECTORERASE <b>command: invalid parameter</b>
\$2718	TPCMD SECTORERASE <b>command: Flash sector erase error</b>
\$2719	TPCMD SECTORERASE <b>command: device is protected</b>
\$271A	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$271B	TPCMD BLANKCHECK <b>command: invalid parameter</b>

Error #	Description
\$271C	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$271D	TPCMD BLANKCHECK <b>command: device is protected</b>
\$271E	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$271F	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2720	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2721	TPCMD PROGRAM <b>command: Flash program error</b>
\$2722	TPCMD PROGRAM <b>command: device is protected</b>
\$2723	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2724	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2725	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2726	TPCMD VERIFY <b>command: Flash verify error</b>
\$2727	TPCMD VERIFY <b>command: device is protected</b>
\$2728	TPSTART <b>command: execution error</b>
\$2729	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$272A	TPEND <b>command: execution error</b>
\$272B	TPCMD READ <b>command: missing or invalid parameter</b>
\$272C	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$272D	TPCMD READ <b>command: read error</b>
<b>Freescale MPC500 FSL_A - Specific Errors</b>	
\$2800	TCSETDEV <b>command: manufacturer not supported</b>
\$2801	TCSETDEV <b>command: algorithm not found on card</b>
\$2802	TCSETDEV <b>command: device not supported</b>
\$2803	TCSETDEV <b>command: internal hardware configuration error</b>
\$2804	TCSETDEV <b>command: corrupted algorithm file</b>
\$2805	TCSETPAR <b>command: parameter not supported</b>
\$2806	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2807	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2808	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2809	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$280A	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$280B	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$280C	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$280D	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$280E	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$280F	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$2810	TCSETPAR PLLFREQ <b>command: FOSC parameter has not been set</b>
\$2811	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$2812	TCSETPAR SCLK <b>command: FOSC or PLLFREQ parameters have not been set</b>

## Errors

Error #	Description
\$2813	TPCMD <b>command: missing parameter</b>
\$2814	TPCMD <b>command: parameter not supported</b>
\$2815	TPCMD RUN <b>command: execution error</b>
\$2816	TPCMD MASSERASE <b>command: missing parameter</b>
\$2817	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2818	TPCMD MASSERASE <b>command: mass erase error</b>
\$2819	TPCMD MASSERASE <b>command: device is protected</b>
\$281A	TPCMD ERASE <b>command: missing parameter</b>
\$281B	TPCMD ERASE <b>command: invalid parameter</b>
\$281C	TPCMD ERASE <b>command: erase error</b>
\$281D	TPCMD ERASE <b>command: device is protected</b>
\$281E	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$281F	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$2820	TPCMD BLANKCHECK <b>command: blank check error</b>
\$2821	TPCMD BLANKCHECK <b>command: device is protected</b>
\$2822	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2823	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2824	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2825	TPCMD PROGRAM <b>command: program error</b>
\$2826	TPCMD PROGRAM <b>command: device is protected</b>
\$2827	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2828	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2829	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$282A	TPCMD VERIFY <b>command: verify error</b>
\$282B	TPCMD VERIFY <b>command: device is protected</b>
\$282C	TPCMD CHANGESENSOR <b>command: missing or invalid parameter</b>
\$282D	TPCMD CHANGESENSOR <b>command: execution error</b>
\$282E	TFSTART <b>command: execution error</b>
\$282F	TFSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$2830	TFEND <b>command: execution error</b>
\$2831	TPCMD READ32 <b>command: missing or invalid parameter</b>
\$2832	TPCMD READ32 <b>command: execution error</b>
\$2833	TPCMD WRITE32 <b>command: missing or invalid parameter</b>
\$2834	TPCMD WRITE32 <b>command: execution error</b>
\$2835	TPCMD READ <b>command: missing or invalid parameter</b>
\$2836	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2837	TPCMD READ <b>command: device is protected</b>
\$2838	TPCMD READ <b>command: read error</b>



Error #	Description
\$2839	TPCMD READ16 <b>command: missing or invalid parameter</b>
\$283A	TPCMD READ16 <b>command: execution error</b>
\$283B	TPCMD WRITE16 <b>command: missing or invalid parameter</b>
\$283C	TPCMD WRITE16 <b>command: execution error</b>
\$283D	TPCMD READ8 <b>command: missing or invalid parameter</b>
\$283E	TPCMD READ8 <b>command: execution error</b>
\$283F	TPCMD WRITE8 <b>command: missing or invalid parameter</b>
\$2840	TPCMD WRITE8 <b>command: execution error</b>
\$2841	<b>This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$2842	TPCMD <b>command: not supported</b>
<b>ARM9 - Specific Errors</b>	
\$2900	TCSETDEV <b>command: manufacturer not supported</b>
\$2901	TCSETDEV <b>command: algorithm not found on card</b>
\$2902	TCSETDEV <b>command: device not supported</b>
\$2903	TCSETDEV <b>command: corrupted algorithm file</b>
\$2904	TCSETDEV <b>command: corrupted algorithm file</b>
\$2905	TCSETDEV <b>command: internal hardware configuration error</b>
\$2906	TCSETPAR <b>command: parameter not supported</b>
\$2907	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2908	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2909	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$290A	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$290B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$290C	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$290D	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$290E	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$290F	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2910	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$2912	TCSETPAR RSTDELAY <b>command: missing, invalid or out of range parameter</b>
\$2913	TPSTART <b>command: command not preceded by TCSETPAR FCPU command</b>
\$2914	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$2916	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$2917	TPCMD <b>command: parameter not supported</b>
\$2918	TPCMD <b>command: command not preceded by TPCMD JTCLK command</b>
\$2919	TPCMD JTCLK <b>command: execution error</b>
\$291A	TPCMD JTCLK <b>command: missing or invalid parameter</b>
\$291B	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$291C	TPCMD MASSERASE <b>command: Flash masserase error</b>

## Errors

Error #	Description
\$291D	TPCMD MASSERASE <b>command: Configuration locations masserase error</b>
\$291E	TPCMD MASSERASE <b>command: User Code locations masserase error</b>
\$2920	TPCMD SECTORERASE <b>command: missing or invalid parameter</b>
\$2921	TPCMD SECTORERASE <b>command: target start address parameter or length parameter out of range</b>
\$2922	TPCMD SECTORERASE <b>command: sector erase error</b>
\$2923	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2924	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2925	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2926	TPCMD PROGRAM <b>command: Flash program error</b>
\$2927	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2928	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2929	TPCMD PROGRAM <b>command: OTP memory program error</b>
\$292A	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$292B	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$292C	TPCMD PROGRAM <b>command: User Code locations program error</b>
\$292D	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$292E	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$292F	TPCMD PROGRAM <b>command: Configuration locations program error</b>
\$2930	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2931	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2932	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2933	TPCMD VERIFY <b>command: Flash verify error</b>
\$2934	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2935	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2936	TPCMD VERIFY <b>command: OTP memory verify error</b>
\$2937	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2938	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2939	TPCMD VERIFY <b>command: User Code locations verify error</b>
\$293A	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$293B	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$293C	TPCMD VERIFY <b>command: Configuration locations verify error</b>
\$2940	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$2941	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$2942	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$2943	TPCMD READ <b>command: missing or invalid parameter</b>
\$2944	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2945	TPCMD READ <b>command: Flash read error</b>
\$2946	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>

Error #	Description
\$2947	TPCMD READ <b>command: OPT memory read error</b>
\$2948	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2949	TPCMD READ <b>command: User Code locations read error</b>
\$294A	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$294B	TPCMD READ <b>command: Configuration locations read error</b>
\$294C	TPCMD WRITE32 <b>command: missing or invalid parameter</b>
\$294D	TPCMD WRITE32 <b>command: execution error</b>
\$294E	TPCMD READ32 <b>command: missing or invalid parameter</b>
\$294F	TPCMD READ32 <b>command: execution error</b>
\$2950	TPCMD WRITE16 <b>command: missing or invalid parameter</b>
\$2951	TPCMD WRITE16 <b>command: execution error</b>
\$2952	TPCMD READ16 <b>command: missing or invalid parameter</b>
\$2953	TPCMD READ16 <b>command: execution error</b>
\$2954	TPCMD LOCK <b>command: missing or invalid parameter</b>
\$2955	TPCMD LOCK <b>command: execution error</b>
\$2956	TPCMD SECURE <b>command: execution error</b>
\$2957	TPCMD UNSECURE <b>command: execution error</b>
\$2958	TPCMD RUN <b>command: execution error</b>
\$295C	TPSTART <b>command: execution error</b>
\$295D	TPEND <b>command: execution error</b>
\$295E	TPCMD <b>command: execution error (could not communicate with target device)</b>
\$295F	<b>Can't establish communication with the target device</b>
<b>Renesas SuperH (JTAG) RNSS_D - Specific Errors</b>	
\$2A00	TCSETDEV <b>command: manufacturer not supported</b>
\$2A01	TCSETDEV <b>command: algorithm not found on card</b>
\$2A02	TCSETDEV <b>command: device not supported</b>
\$2A03	TCSETDEV <b>command: corrupted algorithm file</b>
\$2A04	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$2A05	TCSETDEV <b>command: internal hardware configuration error</b>
\$2A06	TCSETPAR <b>command: parameter not supported</b>
\$2A07	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2A08	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2A09	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2A0A	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$2A0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2A0C	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$2A0D	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2A0E	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>

## Errors

Error #	Description
\$2A0F	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2A10	TFSTART <b>command: command not preceded by</b> TCSETPAR RSTDOWN <b>command</b>
\$2A12	TCSETPAR RSTDELAY <b>command: missing, invalid or out of range parameter</b>
\$2A13	TCSETPAR OSC <b>command: missing, invalid or out of range parameter</b>
\$2A14	TFSTART <b>command: command not preceded by</b> TCSETPAR FOSC <b>command</b>
\$2A15	TCSETPAR KEY_CODE <b>command: missing, invalid or out of range parameter</b>
\$2A16	TFSTART <b>command: command not preceded by</b> TCSETPAR KEY_CODE <b>command</b>
\$2A17	TPCMD <b>command: parameter not supported</b>
\$2A18	TPCMD <b>command: command not preceded by</b> TPCMD JTCLK <b>command</b>
\$2A19	TPCMD JTCLK <b>command: execution error</b>
\$2A1A	TPCMD JTCLK <b>command: missing or invalid parameter</b>
\$2A1B	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2A1C	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2A1D	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2A1E	TPCMD VERIFY <b>command: Flash verify error</b>
\$2A1F	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$2A20	TPCMD VERIFY <b>command: IDCODE verify error</b>
\$2A21	TPCMD READ <b>command: missing or invalid parameter</b>
\$2A22	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2A23	TPCMD READ <b>command: Flash read error</b>
\$2A24	TPCMD READ <b>command: EEPROM read error</b>
\$2A25	TPCMD READ <b>command: IDCODE read error</b>
\$2A26	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2A27	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2A28	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2A29	TPCMD PROGRAM <b>command: Flash program error</b>
\$2A2A	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$2A2B	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$2A2C	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$2A2D	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$2A2E	TPCMD BLANKCHECK <b>command: EEPROM blankcheck error</b>
\$2A2F	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$2A30	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$2A31	TPCMD MASSERASE <b>command: EEPROM masserase error</b>
\$2A32	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$2A33	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$2A34	TPCMD ERASE <b>command: Flash erase error</b>
\$2A35	TPCMD ERASE <b>command: EEPROM erase error</b>

Error #	Description
\$2A36	TPSTART <b>command: execution error</b>
\$2A37	TPEND <b>command: execution error</b>
\$2A38	TPCMD RUN <b>command: execution error</b>
\$2A39	TPCMD WRITE32 <b>command: missing or invalid parameter</b>
\$2A3A	TPCMD WRITE32 <b>command: H-UDI write error</b>
\$2A3B	TPCMD WRITE16 <b>command: missing or invalid parameter</b>
\$2A3C	TPCMD WRITE16 <b>command: H-UDI write error</b>
\$2A3D	TPCMD WRITE8 <b>command: missing or invalid parameter</b>
\$2A3E	TPCMD WRITE8 <b>command: H-UDI write error</b>
\$2A40	TPCMD READ32 <b>command: missing or invalid parameter</b>
\$2A41	TPCMD READ32 <b>command: H-UDI read error</b>
\$2A42	TPCMD READ16 <b>command: missing or invalid parameter</b>
\$2A43	TPCMD READ16 <b>command: H-UDI read error</b>
\$2A44	TPCMD READ8 <b>command: missing or invalid parameter</b>
\$2A45	TPCMD READ8 <b>command: H-UDI read error</b>
\$2A46	TPCMD READ <b>command: Product code read error</b>
\$2A47	TPCMD VERIFY <b>command: product code verify error</b>
\$2A48	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2A49	TPCMD PROGRAM <b>command: NVMB program error</b>
\$2A4A	TPCMD PROGRAM <b>command: NVMB program not supported</b>
\$2A4E	TPCMD UNLOCK <b>command: missing or invalid parameter</b>
\$2A4F	TPCMD UNLOCK <b>command: unlock device error</b>
\$2A50	TPCMD UNLOCK <b>command: unlock device not supported</b>
<b>Freescale MPC500 FSL_B - Specific Errors</b>	
\$2B00	TCSETDEV <b>command: manufacturer not supported</b>
\$2B01	TCSETDEV <b>command: algorithm not found on card</b>
\$2B02	TCSETDEV <b>command: device not supported</b>
\$2B03	TCSETDEV <b>command: internal hardware configuration error</b>
\$2B04	TCSETDEV <b>command: corrupted algorithm file</b>
\$2B05	<b>This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$2B06	TCSETPAR <b>command: parameter not supported</b>
\$2B07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2B08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2B09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2B0A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$2B0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2B0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2B0D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>

## Errors

Error #	Description
\$2B0E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2B0F	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$2B10	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$2B11	TCSETPAR PLLFREQ <b>command: FOSC parameter has not been set</b>
\$2B12	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$2B13	TPCMD <b>command: missing parameter</b>
\$2B14	TPCMD <b>command: parameter not supported</b>
\$2B15	TPCMD RUN <b>command: execution error</b>
\$2B16	TPCMD MASSERASE <b>command: missing parameter</b>
\$2B17	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2B18	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$2B19	TPCMD MASSERASE <b>command: Shadow memory mass erase error</b>
\$2B1A	TPCMD MASSERASE <b>command: device is protected</b>
\$2B1B	TPCMD ERASE <b>command: missing parameter</b>
\$2B1C	TPCMD ERASE <b>command: invalid parameter</b>
\$2B1D	TPCMD ERASE <b>command: Flash memory erase error</b>
\$2B1E	TPCMD ERASE <b>command: Shadow memory erase error</b>
\$2B1F	TPCMD ERASE <b>command: device is protected</b>
\$2B20	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$2B21	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$2B22	TPCMD BLANKCHECK <b>command: Flash memory blank check error</b>
\$2B23	TPCMD BLANKCHECK <b>command: Shadow memory blank check error</b>
\$2B24	TPCMD BLANKCHECK <b>command: device is protected</b>
\$2B25	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2B26	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2B27	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2B28	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$2B29	TPCMD PROGRAM <b>command: Shadow memory program error</b>
\$2B2A	TPCMD PROGRAM <b>command: device is protected</b>
\$2B2B	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2B2C	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2B2D	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2B2E	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$2B2F	TPCMD VERIFY <b>command: Shadow memory verify error</b>
\$2B30	TPCMD VERIFY <b>command: device is protected</b>
\$2B31	TPSTART <b>command: execution error</b>
\$2B32	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$2B33	TPEND <b>command: execution error</b>

Error #	Description
\$2B34	TPCMD MASSERASE <b>command: Data Flash memory mass erase error</b>
\$2B35	TPCMD ERASE <b>command: Data Flash memory erase error</b>
\$2B36	TPCMD BLANKCHECK <b>command: Data Flash memory blankcheck error</b>
\$2B37	TPCMD PROGRAM <b>command: Data Flash memory program error</b>
\$2B38	TPCMD VERIFY <b>command: Data Flash memory verify error</b>
\$2B39	TPCMD ERASE <b>command: invalid parameter</b>
\$2B3A	TPCMD BLANKCHECK <b>command: Test Flash memory blankcheck error</b>
\$2B3B	TPCMD BLANKCHECK <b>command: Data Test Flash memory blankcheck error</b>
\$2B3C	TPCMD PROGRAM <b>command: Test Flash memory program error</b>
\$2B3D	TPCMD PROGRAM <b>command: Data Test Flash memory program error</b>
\$2B3E	TPCMD VERIFY <b>command: Test Flash memory verify error</b>
\$2B3F	TPCMD VERIFY <b>command: Data Test Flash memory verify error</b>
\$2B40	TPCMD READ <b>command: missing or invalid parameter</b>
\$2B41	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2B42	TPCMD READ <b>command: Flash read error</b>
\$2B43	TPCMD READ <b>command: Shadow Flash read error</b>
\$2B44	TPCMD READ <b>command: Device flash secured</b>
\$2B45	TPCMD DUMP <b>command: algorithm not supported</b>
\$2B46	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$2B47	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$2B48	TPCMD DUMP <b>command: dump error</b>
\$2B49	TPCMD JTAGID <b>command: invalid JTAGID</b>
\$2B4A	TCSETPAR RSTEXIT <b>command: missing or invalid parameter</b>

## Errors

Error #	Description
<b>STMicroelectronics STM32 CORTEX - Specific Errors</b>	
\$2C00	TCSETDEV <b>command: manufacturer not supported</b>
\$2C01	TCSETDEV <b>command: algorithm not found on card</b>
\$2C02	TCSETDEV <b>command: device not supported</b>
\$2C03	TCSETDEV <b>command: internal hardware configuration error</b>
\$2C04	TCSETDEV <b>command: corrupted algorithm file</b>
\$2C05	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$2C06	TCSETPAR <b>command: parameter not supported</b>
\$2C07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2C08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2C09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2C0A	TCSETPAR RSTDRV <b>command: missing or invalid command option</b>
\$2C0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2C0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2C0D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2C0E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2C0F	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$2C10	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$2C11	TPCMD <b>command: missing parameter</b>
\$2C12	TPCMD <b>command: parameter not supported</b>
\$2C13	TPCMD RUN <b>command: execution error</b>
\$2C14	TPCMD MASSERASE <b>command: missing parameter</b>
\$2C15	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2C16	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$2C17	TPCMD MASSERASE <b>command: Option Bytes masserase error</b>
\$2C18	TPCMD ERASE <b>command: missing parameter</b>
\$2C19	TPCMD ERASE <b>command: invalid parameter</b>
\$2C1A	TPCMD ERASE <b>command: Flash erase error</b>
\$2C1C	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$2C1D	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$2C1E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$2C20	TPCMD PROGRAM <b>command: missing parameter</b>
\$2C21	TPCMD PROGRAM <b>command: invalid parameter</b>
\$2C22	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2C23	TPCMD PROGRAM <b>command: Flash program error</b>
\$2C24	TPCMD PROGRAM <b>command: Option Bytes program error</b>
\$2C25	TPCMD VERIFY <b>command: missing parameter</b>
\$2C26	TPCMD VERIFY <b>command: invalid parameter</b>



Error #	Description
\$2C27	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2C28	TPCMD VERIFY <b>command: Flash verify error</b>
\$2C29	TPCMD VERIFY <b>command: Option Bytes verify error</b>
\$2C2F	TPSTART <b>command: execution error</b>
\$2C30	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$2C31	TPEND <b>command: execution error</b>
\$2C32	TCSETPAR RSTTYPE <b>command: missing or invalid command option</b>
<b>Infineon XC2000 INF_B - Specific Errors</b>	
\$2D00	TCSETDEV <b>command: manufacturer not supported</b>
\$2D01	TCSETDEV <b>command: algorithm not found on card</b>
\$2D02	TCSETDEV <b>command: device not supported</b>
\$2D03	TCSETDEV <b>command: corrupted algorithm file</b>
\$2D04	TCSETDEV <b>command: internal hardware configuration error</b>
\$2D05	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$2D06	TCSETPAR <b>command: parameter not supported</b>
\$2D07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2D08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2D09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2D0A	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2D0B	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2D0C	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2D0D	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2D0E	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$2D10	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$2D11	TCSETPAR JTCLK <b>command: PLLFREQ parameter has not been set</b>
\$2D12	TPCMD <b>command: missing parameter</b>
\$2D13	TPCMD <b>command: parameter not supported</b>
\$2D14	TPCMD RUN <b>command: execution error</b>
\$2D15	TPCMD MASSERASE <b>command: missing parameter</b>
\$2D16	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2D17	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$2D18	TPCMD MASSERASE <b>command: device is protected</b>
\$2D1D	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$2D1E	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$2D1F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$2D20	TPCMD BLANKCHECK <b>command: device is protected</b>
\$2D21	TPCMD PROGRAM <b>command: missing parameter</b>
\$2D22	TPCMD PROGRAM <b>command: invalid parameter</b>

## Errors

Error #	Description
\$2D23	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2D24	TPCMD PROGRAM <b>command: Flash program error</b>
\$2D25	TPCMD PROGRAM <b>command: device is protected</b>
\$2D26	TPCMD VERIFY <b>command: missing parameter</b>
\$2D27	TPCMD VERIFY <b>command: invalid parameter</b>
\$2D28	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2D29	TPCMD VERIFY <b>command: Flash verify error</b>
\$2D2A	TPCMD VERIFY <b>command: device is protected</b>
\$2D2B	TPSTART <b>command: execution error</b>
\$2D2C	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$2D2D	TPEND <b>command: execution error</b>
\$2D2E	TCSETPAR PLLNDIV <b>command: missing, invalid or out of range parameter</b>
\$2D2F	TCSETPAR PLLPDIV <b>command: missing, invalid or out of range parameter</b>
\$2D30	TCSETPAR PLLKDIV <b>command: missing, invalid or out of range parameter</b>
\$2D31	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$2D32	TCSETPAR PLLFREQ <b>command: PLLNDIV, PLLPDIV or PLLKDIV parameters have not been set</b>
\$2D33	TCSETPAR DEVICE_STEP <b>command: missing, invalid or out of range parameter</b>
\$2D34	TPCMD ERASE <b>command: invalid parameter</b>
\$2D35	TPCMD ERASE <b>command: invalid target address or parameter out of range</b>
\$2D36	TPCMD ERASE <b>command: source offset parameter out of range</b>
\$2D37	TPCMD ERASE <b>command: Flash erase error</b>
\$2D38	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$2D39	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$2D3A	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$2D3B	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$2D3C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$2D3D	TPCMD READ <b>command: invalid parameter</b>
\$2D3E	TPCMD PROGVER <b>command: invalid parameter</b>
\$2D3F	TPCMD PROGVER <b>command: invalid target address or parameter out of range</b>
\$2D40	TPCMD PROGVER <b>command: source offset parameter out of range</b>
\$2D41	TPCMD PROGVER <b>command: Flash program version error</b>
\$2D42	TPCMD PROTECT <b>command: invalid parameter</b>
\$2D43	TPCMD PROTECT <b>command: memory protect error</b>
\$2D42	TPCMD UNPROTECT <b>command: invalid parameter</b>
\$2D43	TPCMD UNPROTECT <b>command: memory unprotect error</b>
<b>Cypress PSoC CY_A - Specific Errors</b>	
\$2E00	TCSETDEV <b>command: manufacturer not supported</b>
\$2E01	TCSETDEV <b>command: algorithm not found on card</b>

Error #	Description
\$2E02	TCSETDEV <b>command: device not supported</b>
\$2E03	TCSETDEV <b>command: internal hardware configuration error</b>
\$2E04	TCSETPAR <b>command: parameter not supported</b>
\$2E05	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$2E06	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$2E07	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2E08	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$2E09	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2E0A	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$2E0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2E0C	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$2E0D	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2E0E	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$2E0F	TCSETPAR ENTRY_MODE <b>command: missing or invalid command option</b>
\$2E10	TPSTART <b>command: command not preceded by TCSETPAR ENTRY_MODE command</b>
\$2E11	TPCMD <b>command: parameter not supported</b>
\$2E12	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2E13	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2E14	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2E15	TPCMD VERIFY <b>command: Flash verify error (read-out method)</b>
\$2E16	TPCMD VERIFY <b>command: Flash verify error (checksum method)</b>
\$2E17	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2E18	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2E19	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2E1A	TPCMD PROGRAM <b>command: Flash program error</b>
\$2E1B	TPCMD PROGRAM <b>command: Security blocks program error</b>
\$2E1C	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$2E1D	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$2E1E	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$2E20	TPCMD READ <b>command: missing or invalid parameter</b>
\$2E21	TPCMD READ F <b>command: target start address parameter or length parameter out of range</b>
\$2E22	TPCMD READ F <b>command: Flash read error</b>
\$2E23	TPCMD READ S <b>command: device checksum read error</b>
\$2E24	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$2E25	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$2E26	TPEND <b>command: execution error</b>
\$2E27	TPCMD <b>command: execution error (could not communicate with target device)</b>
\$2E28	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>

## Errors

Error #	Description
<b>STMicroelectronics STM8 STM_A - Specific Errors</b>	
\$2F00	TCSETDEV <b>command: manufacturer not supported</b>
\$2F01	TCSETDEV <b>command: algorithm not found on card</b>
\$2F02	TCSETDEV <b>command: device not supported</b>
\$2F03	TCSETDEV <b>command: corrupted algorithm file</b>
\$2F04	TCSETDEV <b>command: internal hardware configuration error</b>
\$2F05	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$2F06	TCSETPAR <b>command: parameter not supported</b>
\$2F07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2F08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2F09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2F0A	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2F0B	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2F0C	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$2F0D	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2F0E	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$2F12	TPCMD <b>command: missing parameter</b>
\$2F13	TPCMD <b>command: parameter not supported</b>
\$2F14	TPCMD RUN <b>command: execution error</b>
\$2F15	TPCMD MASSERASE <b>command: missing parameter</b>
\$2F16	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2F17	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$2F18	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$2F19	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$2F1A	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$2F1B	TPCMD ERASE <b>command: Flash erase error</b>
\$2F1C	TPCMD ERASE <b>command: EEPROM erase error</b>
\$2F1D	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$2F1E	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$2F1F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$2F20	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$2F21	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2F22	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2F23	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2F24	TPCMD PROGRAM <b>command: Flash program error</b>
\$2F25	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$2F26	TPCMD PROGRAM <b>command: Option Bytes program error</b>
\$2F27	TPCMD VERIFY <b>command: missing or invalid parameter</b>

Error #	Description
\$2F28	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2F29	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2F2A	TPCMD VERIFY <b>command: Flash verify error</b>
\$2F2B	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$2F2C	TPCMD VERIFY <b>command: Option Bytes verify error</b>
\$2F2D	TPSTART <b>command: execution error</b>
\$2F2E	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$2F2F	TPEND <b>command: execution error</b>
\$2F30	TPCMD CONNECT <b>command: execution error</b>
\$2F31	TPCMD DISCONNECT <b>command: execution error</b>
\$2F32	<b>Device is protected</b>
\$2F33	<b>Device is not connected</b> (TPCMD CONNECT <b>command has not been executed</b> )
\$2F34	TPCMD READ <b>command: missing or invalid parameter</b>
\$2F35	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2F36	TPCMD READ <b>command: read error</b>
\$2F38	TPCMD SWIM_RST <b>command: error on MCU reset execution</b>
\$2F39	<b>MCU reset execution error</b>
<b>Samsung SAM8 SMSNG_A - Specific Errors</b>	
\$3000	TCSETDEV <b>command: manufacturer not supported</b>
\$3001	TCSETDEV <b>command: algorithm not found on card</b>
\$3002	TCSETDEV <b>command: device not supported</b>
\$3003	TCSETDEV <b>command: internal hardware configuration error</b>
\$3004	TCSETPAR <b>command: parameter not supported</b>
\$3005	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$3006	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$3007	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$3008	TPSTART <b>command: command not preceded by</b> TCSETPAR PWUP <b>command</b>
\$3009	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$300A	TPSTART <b>command: command not preceded by</b> TCSETPAR PWDOWN <b>command</b>
\$300B	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$300C	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTUP <b>command</b>
\$300D	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$300E	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTDOWN <b>command</b>
\$300F	TCSETPAR WRITE_CLOCK <b>command: missing, invalid or out of range parameter</b>
\$3010	TPSTART <b>command: command not preceded by</b> TCSETPAR WRITE_CLOCK <b>command</b>
\$3011	TCSETPAR READ_CLOCK <b>command: missing, invalid or out of range parameter</b>
\$3012	TPSTART <b>command: command not preceded by</b> TCSETPAR READ_CLOCK <b>command</b>
\$3016	TPCMD <b>command: invalid command option</b>
\$3017	TPCMD MASSERASE <b>command: missing or invalid command option</b>

## Errors

Error #	Description
\$3018	TPCMD MASSERASE C <b>command: chip mass erase error</b>
\$3019	TPCMD BLANKCHECK <b>command: missing or invalid parameter or command option</b>
\$3020	TPCMD BLANKCHECK F <b>command: target start address parameter or length parameter out of range</b>
\$3021	TPCMD BLANKCHECK F <b>command: Flash blank check error</b>
\$3022	TPCMD BLANKCHECK S <b>command: target start address parameter or length parameter out of range</b>
\$3023	TPCMD BLANKCHECK S <b>command: Smart Option blank check error</b>
\$3024	TPCMD PROGRAM <b>command: missing or invalid parameter or command option</b>
\$3025	TPCMD PROGRAM F <b>command: target start address parameter or length parameter out of range</b>
\$3026	TPCMD PROGRAM F <b>command: source offset parameter out of range</b>
\$3027	TPCMD PROGRAM F <b>command: Flash memory program error</b>
\$3028	TPCMD PROGRAM S <b>command: target start address parameter or length parameter out of range</b>
\$3029	TPCMD PROGRAM S <b>command: source offset parameter out of range</b>
\$302A	TPCMD PROGRAM S <b>command: Smart Option program error</b>
\$302B	TPCMD PROGRAM R H L <b>command: target start address parameter or length parameter out of range</b>
\$302C	TPCMD PROGRAM R H L <b>command: source offset parameter out of range</b>
\$302D	TPCMD PROGRAM R <b>command: ReadProtection program error</b>
\$302E	TPCMD PROGRAM H <b>command: HardLock program error</b>
\$302F	TPCMD PROGRAM L <b>command: LDCProtection program error</b>
\$3030	TPCMD VERIFY <b>command: missing or invalid parameter or command option</b>
\$3031	TPCMD VERIFY F <b>command: target start address parameter or length parameter out of range</b>
\$3032	TPCMD VERIFY F <b>command: source offset parameter out of range</b>
\$3033	TPCMD VERIFY F <b>command: Flash memory verify error</b>
\$3034	TPCMD VERIFY S <b>command: target start address parameter or length parameter out of range</b>
\$3035	TPCMD VERIFY S <b>command: source offset parameter out of range</b>
\$3036	TPCMD VERIFY S <b>command: Smart Option verify error</b>
\$3037	TPCMD VERIFY R H L <b>command: target start address parameter or length parameter out of range</b>
\$3038	TPCMD VERIFY R H L <b>command: source offset parameter out of range</b>
\$3039	TPCMD VERIFY R <b>command: ReadProtection verify error</b>
\$303A	TPCMD VERIFY H <b>command: HardLock verify error</b>
\$303B	TPCMD VERIFY L <b>command: LDCProtection verify error</b>
\$303C	TPCMD READ <b>command: missing or invalid parameter or command option</b>
\$303D	TPCMD READ F <b>command: target start address parameter or length parameter out of range</b>
\$303E	TPCMD READ F <b>command: Flash memory read error</b>
\$303F	TPCMD READ S <b>command: target start address parameter or length parameter out of range</b>
\$3040	TPCMD READ S <b>command: Smart Option read error</b>
\$3041	TPCMD READ R H L <b>command: target start address parameter or length parameter out of range</b>
\$3042	TPCMD READ R <b>command: ReadProtection read error</b>
\$3043	TPCMD READ H <b>command: HardLock read error</b>

Error #	Description
\$3044	TPCMD READ L <b>command: LDCProtection read error</b>
\$3045	TPCMD RUN <b>command: execution error</b>
\$3046	TPSTART <b>command: execution error</b>
\$3047	TPEND <b>command: execution error</b>
\$3048	<b>Can't establish communication with the target device</b>
<b>Freescale 56F800E FSL_C - Specific Errors</b>	
\$3100	TCSETDEV <b>command: manufacturer not supported</b>
\$3101	TCSETDEV <b>command: algorithm not found on card</b>
\$3102	TCSETDEV <b>command: device not supported</b>
\$3103	TCSETDEV <b>command: internal hardware configuration error</b>
\$3104	TCSETPAR <b>command: parameter not supported</b>
\$3105	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$3106	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$3107	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$3108	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$3109	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$310A	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$310B	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$310C	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$310D	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$310E	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$310F	TCSETPAR INIT_JTCLK <b>command: missing, invalid or out of range parameter</b>
\$3110	TPSTART <b>command: command not preceded by TCSETPAR INIT_JTCLK command</b>
\$3111	TCSETPAR PROG_JTCLK <b>command: missing, invalid or out of range parameter</b>
\$3112	TPSTART <b>command: command not preceded by TCSETPAR PROG_JTCLK command</b>
\$3113	TPCMD <b>command: invalid command option</b>
\$3114	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$3120	TPCMD MASSERASE <b>command: missing or invalid command option</b>
\$3121	TPCMD MASSERASE F <b>command: Flash mass erase error</b>
\$3122	TPCMD MASSERASE C <b>command: device is secured</b>
\$3123	TPCMD ERASE <b>command: missing or invalid command option or parameter</b>
\$3124	TPCMD ERASE F <b>command: target start address parameter or length parameter out of range</b>
\$3125	TPCMD ERASE F <b>command: Flash erase error</b>
\$3126	TPCMD PROGRAM <b>command: missing or invalid parameter or command option</b>
\$3127	TPCMD PROGRAM F <b>command: target start address parameter or length parameter out of range</b>
\$3128	TPCMD PROGRAM F <b>command: source offset parameter out of range</b>
\$3129	TPCMD PROGRAM F <b>command: Flash memory program error</b>
\$312A	TPCMD VERIFY <b>command: missing or invalid parameter or command option</b>

## Errors

Error #	Description
\$312B	TPCMD VERIFY F <b>command: target start address parameter or length parameter out of range</b>
\$312C	TPCMD VERIFY F <b>command: source offset parameter out of range</b>
\$312D	TPCMD VERIFY F <b>command: Flash memory verify error</b>
\$312E	TPCMD BLANKCHECK <b>command: missing or invalid parameter or command option</b>
\$312F	TPCMD BLANKCHECK F <b>command: target start address parameter or length parameter out of range</b>
\$3130	TPCMD BLANKCHECK F <b>command: Flash blank check error</b>
\$3131	TPCMD READ <b>command: missing or invalid parameter or command option</b>
\$3132	TPCMD READ F <b>command: target start address parameter or length parameter out of range</b>
\$3133	TPCMD READ F <b>command: Flash memory read error</b>
\$3134	TPCMD RUN <b>command: execution error</b>
\$3135	TPSTART <b>command: execution error</b>
\$3136	TPEND <b>command: execution error</b>
\$3137	<b>Can't establish communication with the target device</b>
\$3138	TPCMD GET_JTAG_ID <b>command: GET_JTAG_ID error</b>
\$3139	TPCMD <b>command: command not supported</b>
<b>Texas Instruments TMS320 TL_B - Specific Errors</b>	
\$3200	TCSETDEV <b>command: manufacturer not supported</b>
\$3201	TCSETDEV <b>command: algorithm not found on card</b>
\$3202	TCSETDEV <b>command: device not supported</b>
\$3204	TCSETDEV <b>command: internal hardware configuration error</b>
\$3205	TCSETDEV <b>command: old internal fw version</b>
\$3206	TCSETPAR <b>command: not standard parameter</b>
\$3207	TCSETPAR <b>command: wrong clock out parameter</b>
\$3208	TCSETPAR <b>command: wrong reset down parameter</b>
\$3209	TCSETPAR <b>command: wrong reset up parameter</b>
\$320A	TCSETPAR <b>command: wrong reset drive parameter</b>
\$320B	TCSETPAR <b>command: wrong power down parameter</b>
\$320C	TCSETPAR <b>command: wrong power up parameter</b>
\$320D	TCSETPAR <b>command: wrong vdd parameter</b>
\$320E	TCSETPAR <b>command: wrong vdd aux parameter</b>
\$320F	TCSETPAR <b>command: communication mode missing</b>
\$3210	TCSETPAR <b>command: wrong baudrate value</b>
\$3211	TCSETPAR <b>command: wrong fosc clock parameter</b>
\$3212	TCSETPAR <b>command: wrong fpll parameter</b>
\$3213	TCSETPAR <b>command: missing fosc parameter</b>
\$3214	TCSETPAR <b>command: wrong baudrate parameter</b>
\$3215	TCSETPAR <b>command: missing fosc parameter</b>
\$3216	TCSETPAR <b>command: pwd missing</b>



Error #	Description
\$3217	TPCMD <b>command: command missing</b>
\$3218	TPCMD <b>command: invalid command</b>
\$3219	TPCMD RUN <b>command: run ececution error</b>
\$321A	TPCMD MASSERASE <b>command: masserese syntax error</b>
\$321B	TPCMD MASSERASE <b>command: masserese memory parameter error</b>
\$321C	TPCMD MASSERASE <b>command: masserese execution error</b>
\$321D	TPCMD BLANKCHECK <b>command: blankcheck syntax error</b>
\$321E	TPCMD BLANKCHECK <b>command: blankcheck memory parameter error</b>
\$321F	TPCMD BLANKCHECK <b>command: blankcheck execution error</b>
\$3220	TPCMD PROGRAM <b>command: program syntax error</b>
\$3221	TPCMD PROGRAM <b>command: memory parameter error</b>
\$3222	TPCMD PROGRAM <b>command: source file error</b>
\$3223	TPCMD PROGRAM <b>command: program execution error</b>
\$3225	TPCMD VERIFY <b>command: verify syntax error</b>
\$3226	TPCMD VERIFY <b>command: verify memory parameter error</b>
\$3227	TPCMD VERIFY <b>command: source file error</b>
\$3228	TPCMD VERIFY <b>command: verify execution error</b>
\$3228	TPCMD TPSTART <b>command: unable to execute start</b>
\$3229	TPCMD TPSTART <b>command: execution error</b>
\$322A	TPCMD TPEND <b>command: execution error</b>
\$322B	TPCMD CONNECT <b>command: execution error</b>
\$322C	TPCMD DISCONNECT <b>command: execution error</b>
\$322D	TPCMD CONNECT <b>command: device not connected</b>
\$322E	TPCMD RSTDLY <b>command: reset delay parameter error</b>
<b>ELMOS EL16/E135/E909 EL_A - Specific Errors</b>	
\$3300	TCSETDEV <b>command: manufacturer not supported</b>
\$3301	TCSETDEV <b>command: algorithm not found on card</b>
\$3302	TCSETDEV <b>command: invalid device code</b>
\$3303	TCSETDEV <b>command: invalid device data</b>
\$3304	TCSETDEV <b>command: internal hardware configuration error</b>
\$3305	TCSETPAR <b>command: parameter not supported</b>
\$3306	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$3307	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$3308	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$3309	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$330A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$330B	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$330C	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>

## Errors

Error #	Description
\$330D	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$330E	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$330F	TPCMD <b>command: missing parameter</b>
\$3310	TPCMD <b>command: parameter not supported</b>
\$3311	TPCMD RUN <b>command: execution error</b>
\$3312	TPCMD MASSERASE <b>command: missing parameter</b>
\$3313	TPCMD MASSERASE <b>command: invalid parameter</b>
\$3314	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$3315	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$3316	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$3317	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$3318	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$3319	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$331A	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$331B	TPCMD PROGRAM <b>command: Flash program error</b>
\$331C	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$331D	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$331E	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$331F	TPCMD VERIFY <b>command: Flash verify error</b>
\$3320	TPCMD READ <b>command: missing or invalid parameter</b>
\$3321	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$3322	TPCMD READ <b>command: read error</b>
\$3323	TPSTART <b>command: execution error</b>
\$3324	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$3325	TPEND <b>command: execution error</b>
<b>Infineon TriCore INF_C - Specific Errors</b>	
\$3500	TCSETDEV <b>command: manufacturer not supported</b>
\$3501	TCSETDEV <b>command: algorithm not found on card</b>
\$3502	TCSETDEV <b>command: device not supported</b>
\$3503	TCSETDEV <b>command: internal hardware configuration error</b>
\$3504	TCSETPAR <b>command: parameter not supported</b>
\$3505	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$3506	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$3507	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$3508	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$3509	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$350A	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$350B	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$350C	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$350D	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$350E	TCSETPAR JTCLK <b>command: missing or invalid command option</b>
\$350F	TCSETPAR JTCLK <b>command: the protocol set is not JTAG</b>
\$3510	TCSETPAR JTCLK <b>command: FCPU parameter missing</b>
\$3511	TPCMD <b>command: missing parameter</b>
\$3512	TPCMD <b>command: parameter not supported</b>
\$3513	TPCMD RUN <b>command: execution error</b>
\$3514	TPCMD WRITE32 <b>command: missing or invalid parameter</b>
\$3515	TPCMD WRITE32 <b>command: execution error</b>
\$3516	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$3517	TPCMD MASSERASE <b>command: invalid parameter</b>
\$3518	TPCMD MASSERASE <b>command: the device is secured</b>
\$3519	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$351A	TPCMD SECTORERASE <b>command: missing or invalid parameter</b>
\$351B	TPCMD SECTORERASE <b>command: missing or invalid parameter</b>
\$351C	TPCMD SECTORERASE <b>command: Flash device secured</b>
\$351D	TPCMD SECTORERASE <b>command: sector erase Flash error</b>
\$351E	TPCMD SECTORERASE <b>command: source offset parameter out of range</b>
\$351F	TPCMD SECTORERASE <b>command: UC Block not protected</b>
\$3520	TPCMD SECTORERASE <b>command: UC Block not erasable</b>
\$3521	TPCMD SECTORERASE <b>command: UC Block not wrong data</b>
\$3522	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$3523	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$3524	TPCMD BLANKCHECK <b>command: Flash device secured</b>
\$3525	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$3526	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$3527	TPCMD PROGRAM <b>command: invalid parameter</b>
\$3528	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$3529	TPCMD PROGRAM <b>command: Flash device secured</b>
\$352A	TPCMD PROGRAM <b>command: Flash program error</b>
\$352B	TPCMD PROGRAM <b>command: UC Block wrong data</b>
\$352C	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$352D	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$352E	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$352F	TPCMD VERIFY <b>command: Flash device secured</b>
\$3530	TPCMD VERIFY <b>command: Flash verify error</b>
\$3531	TPSTART <b>command: missing parameter</b>

## Errors

Error #	Description
\$3532	TFSTART <b>command: load FPGA</b>
\$3533	TFSTART <b>command: execution error</b>
\$3534	TFEND <b>command: execution error</b>
\$3535	<b>Algorithm not supported</b>
\$3536	TPCMD READ <b>command: missing or invalid parameter</b>
\$3537	TPCMD <b>command: command not available</b>
\$3538	TCSETPAR DBG_IF_PWD <b>command: Set password to unprotect error</b>
<b>Freescal S12 FSL_D - Specific Errors</b>	
\$3600	TCSETDEV <b>command: manufacturer not supported</b>
\$3601	TCSETDEV <b>command: algorithm not found on card</b>
\$3602	TCSETDEV <b>command: device not supported</b>
\$3603	TCSETDEV <b>command: corrupted algorithm file</b>
\$3604	TCSETDEV <b>command: internal hardware configuration error</b>
\$3610	TCSETPAR <b>command: parameter not supported</b>
\$3611	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$3612	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$3613	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$3614	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$3615	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$3616	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$3617	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$3618	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$361A	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$361B	TCSETPAR REFDIV <b>command: missing, invalid or out of range parameter</b>
\$361C	TCSETPAR SYNR <b>command: missing, invalid or out of range parameter</b>
\$361D	TPCMD <b>command: missing parameter</b>
\$361E	TPCMD <b>command: parameter not supported</b>
\$361F	TPCMD RUN <b>command: execution error</b>
\$3620	TPCMD UNSECURE <b>command: execution error</b>
\$3621	TPCMD MASSERASE <b>command: missing parameter</b>
\$3622	TPCMD MASSERASE <b>command: invalid parameter</b>
\$3623	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$3624	TPCMD MASSERASE <b>command: device is protected</b>
\$3625	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$3626	TPCMD MASSERASE <b>command: device is protected</b>
\$3627	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$3628	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$3629	TPCMD BLANKCHECK <b>command: Flash blank check error</b>

Error #	Description
\$362A	TPCMD BLANKCHECK <b>command: device is protected</b>
\$362B	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$362C	TPCMD BLANKCHECK <b>command: device is protected</b>
\$362D	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$362E	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$362F	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$3630	TPCMD PROGRAM <b>command: Flash program error</b>
\$3631	TPCMD PROGRAM <b>command: device is protected</b>
\$3632	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$3633	TPCMD PROGRAM <b>command: device is protected</b>
\$3634	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$3635	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$3636	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$3637	TPCMD VERIFY <b>command: Flash verify error</b>
\$3638	TPCMD VERIFY <b>command: device is protected</b>
\$3639	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$363A	TPCMD VERIFY <b>command: device is protected</b>
\$363B	TPSTART <b>command: execution error</b>
\$363C	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$363D	TPEND <b>command: execution error</b>
\$363E	TPCMD READ <b>command: missing or invalid parameter</b>
\$363F	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$3640	TPCMD READ F <b>command: Flash read error</b>
\$3641	TPCMD READ F <b>command: device is protected</b>
\$3642	TPCMD READ E <b>command: EEPROM read error</b>
\$3643	TPCMD READ E <b>command: device is protected</b>
\$3644	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$3645	TCSETDEV <b>command: OS version too old</b>
\$3646	TPCMD PROGRAM <b>command: System memory program error</b>
\$3647	TPCMD PROGRAM <b>command: device is protected</b>
\$3648	TPCMD PROGRAM <b>command: Emulated EEPROM program error</b>
\$3649	TPCMD PROGRAM <b>command: device is protected</b>
\$364A	TPCMD VERIFY <b>command: Emulated EEPROM verify error</b>
\$364B	TPCMD VERIFY <b>command: device is protected</b>
\$364C	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$364D	TPCMD DUMP <b>command: destination offset parameter out of range</b>
\$364E	TPCMD DUMP <b>command: Flash dump error</b>
\$364F	TPCMD DUMP <b>command: EEprom dump error</b>

## Errors

Error #	Description
<b>Fujitsu MB91 FJTS_B - Specific Errors</b>	
\$3700	TCSETDEV command: manufacturer not supported
\$3701	TCSETDEV command: algorithm not found on card
\$3702	TCSETDEV command: device not supported
\$3703	TCSETDEV command: corrupted algorithm file
\$3704	TCSETDEV command: internal hardware configuration error
\$3705	TCSETDEV command: programming algorithm requires FlashRunner firmware version 01.03.00.00 or newer
\$3706	TCSETPAR command: parameter not supported
\$3707	TCSETPAR CLKOUT command: missing, invalid or out of range parameter
\$3708	TCSETPAR RSTDOWN command: missing, invalid or out of range parameter
\$3709	TCSETPAR RSTUP command: missing, invalid or out of range parameter
\$370A	TCSETPAR RSTDRV command: missing, invalid or out of range parameter
\$370B	TCSETPAR PWDOWN command: missing, invalid or out of range parameter
\$370C	TCSETPAR PWUP command: missing, invalid or out of range parameter
\$370D	TCSETPAR VDD command: missing, invalid or out of range parameter
\$370E	TCSETPAR VDD_AUX command: missing, invalid or out of range parameter
\$370F	TCSETPAR CMODE command: missing, invalid or out of range parameter
\$3710	TCSETPAR FOSC command: missing, invalid or out of range parameter
\$3711	TCSETPAR FPLL command: missing, invalid or out of range parameter
\$3712	TCSETPAR FPLL command: FOSC, DIVN or DIVM parameters have not been set
\$3713	TCSETPAR BAUDRATE command: missing, invalid or out of range parameter
\$3714	TCSETPAR BAUDRATE command: FOSC or FPLL parameters have not been set
\$3715	TCSETPAR DIVN command: missing, invalid or out of range parameter
\$3716	TCSETPAR DIVM command: missing, invalid or out of range parameter
\$3734	TCSETPAR DIVG command: missing, invalid or out of range parameter
\$3735	TCSETPAR MULG command: missing, invalid or out of range parameter
\$3717	TPCMD command: missing parameter
\$3718	TPCMD command: parameter not supported
\$3719	TPCMD RUN command: execution error
\$371A	TPCMD MASSERASE command: missing parameter
\$371B	TPCMD MASSERASE command: invalid parameter
\$371C	TPCMD MASSERASE command: Flash mass erase error
\$371D	TPCMD BLANKCHECK command: missing or invalid parameter
\$371E	TPCMD BLANKCHECK command: target start address parameter or length parameter out of range
\$371F	TPCMD BLANKCHECK command: Flash blank check error
\$3720	TPCMD PROGRAM command: missing or invalid parameter
\$3721	TPCMD PROGRAM command: target start address parameter or length parameter out of range
\$3722	TPCMD PROGRAM command: source offset parameter out of range

Error #	Description
\$3723	TPCMD PROGRAM command: Flash program error
\$3724	TPCMD VERIFY command: missing or invalid parameter
\$3725	TPCMD VERIFY command: target start address parameter or length parameter out of range
\$3726	TPCMD VERIFY command: source offset parameter out of range
\$3727	TPCMD VERIFY command: Flash verify error
\$3728	TPSTART command: execution error
\$3729	TPSTART command: one or more required TCSETPAR commands have not been sent
\$372A	TPEND command: execution error
\$372B	TPCMD CONNECT command: execution error
\$372C	TPCMD DISCONNECT command: execution error
\$372D	Device is not connected (TPCMD CONNECT command has not been executed)
\$372E	TPCMD ERASE command: missing parameter
\$372F	TPCMD ERASE command: invalid parameter
\$3730	TPCMD ERASE command: Flash erase error
\$3731	TPCMD READ command: missing or invalid parameter
\$3732	TPCMD READ command: read error
\$3733	TPCMD READ command: target start address parameter or length parameter out of range
\$3734	TCSETPAR DIVG command: invalid or out of range parameter
\$3735	TCSETPAR MULG command: invalid or out of range parameter
\$3736	TPCMD READ command: read execution error
\$3737	TPCMD UNSECURE command: Unsecure execution error

## Errors

Error #	Description
<b>MICROCHIP MCHP_B - Specific Errors</b>	
\$3800	TCSETDEV <b>command: manufacturer not supported</b>
\$3801	TCSETDEV <b>command: algorithm not found on card</b>
\$3802	TCSETDEV <b>command: device not supported</b>
\$3803	TCSETDEV <b>command: internal hardware configuration error</b>
\$3804	TCSETPAR <b>command: parameter not supported</b>
\$3805	TCSETPAR <b>command: wrong clockout value</b>
\$3806	TCSETPAR <b>command: missing clockout value</b>
\$3807	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$3808	TCSETPAR VDD <b>command: missing VDD value</b>
\$3809	TCSETPAR VPP <b>command: invalid or out of range parameter</b>
\$380A	TCSETPAR VPP <b>command: missing VDD value</b>
\$380B	TCSETPAR PWDOWN <b>command: PWDOWN invalid or out of range parameter</b>
\$380C	TCSETPAR PWDOWN <b>command: missing PWDOWN value</b>
\$380D	TCSETPAR PWUP <b>command: PWUP invalid or out of range parameter</b>
\$381E	TCSETPAR PWDUP <b>command: missing PWUP value</b>
\$3810	TPCMD <b>command: invalid command option</b>
\$3811	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$3812	TPCMD PROGRAM <b>command: wrong Eprom parameter</b>
\$3813	TPCMD PROGRAM <b>command: source invalid parameter</b>
\$3814	TPCMD PROGRAM <b>command: Eprom program error</b>
\$3815	TPCMD PROGRAM <b>command: VDD error</b>
\$3816	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$3817	TPCMD VERIFY <b>command: wrong Eprom parameter</b>
\$3818	TPCMD VERIFY <b>command: source invalid parameter</b>
\$3819	TPCMD VERIFY <b>command: Eprom program error</b>
\$3821	TPCMD READ <b>command: missing or invalid parameter or command option</b>
\$3822	TPCMD READ <b>command: read eprom parameter error</b>
\$3823	TPCMD READ <b>command: read eprom error</b>
\$3824	TPCMD RUN: <b>missing or invalid parameter</b>
\$3825	TFSTART: <b>start execution error</b>
\$3826	TFEND: <b>end execution error</b>



Error #	Description
<b>Atmel ATXMEGA - Specific Errors</b>	
\$3900	TCSETDEV <b>command: manufacturer not supported</b>
\$3901	TCSETDEV <b>command: algorithm not found on card</b>
\$3902	TCSETDEV <b>command: device not supported</b>
\$3903	TCSETDEV <b>command: internal hardware configuration error</b>
\$3904	TPCMD MASSERASE <b>command: invalid parameter</b>
\$3905	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$3906	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$3907	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$3908	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$3909	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$390A	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$390B	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$390C	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$390D	TPCMD PROGRAM <b>command: Flash program error</b>
\$390E	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$390F	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$3910	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$3911	TPCMD VERIFY <b>command: Flash program error</b>
\$3912	TPCMD READ <b>command: missing or invalid parameter or command option</b>
\$3913	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$3914	TPCMD READ <b>command: Flash memory read error</b>
\$3915	TPCMD RUN <b>command: execution error</b>
\$3916	TPCMD <b>command: invalid command option</b>
\$3917	TCSETPAR <b>command: parameter not supported</b>
\$3918	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$3919	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$391A	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$391B	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$391C	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$391D	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$391E	TCSETPAR RSTDRV <b>command: or invalid parameter</b>
\$391F	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$3920	TCSETPAR JTCLK <b>command: invalid or out of range parameter</b>
\$3921	TCSETPAR CMODE <b>command: missing parameter</b>
\$3922	TCSETPAR VDD <b>command: missing parameter</b>
\$3923	TCSETPAR JTCLK <b>command: missing parameter</b>
\$3924	TCSETPAR PWUP <b>command: missing parameter</b>

## Errors

Error #	Description
\$3925	TCSETPAR PWDOWN <b>command: missing parameter</b>
\$3926	TCSETPAR RSTUP <b>command: missing parameter</b>
\$3927	TCSETPAR RSTDOWN <b>command: missing parameter</b>
\$3928	TCSETPAR CMODE <b>command: missing parameter</b>
\$3929	TCSETPAR RSTDRV <b>command: missing parameter</b>
\$3930	TFSTART <b>command: execution error</b>
\$3931	TPEND <b>command: execution error</b>
<b>NXP LPC1100L NXP_A - Specific Errors</b>	
\$3A00	TCSETDEV <b>command: manufacturer not supported</b>
\$3A01	TCSETDEV <b>command: algorithm not found on card</b>
\$3A02	TCSETDEV <b>command: device not supported</b>
\$3A03	TCSETDEV <b>command: corrupted algorithm file</b>
\$3A04	TCSETDEV <b>command: internal hardware configuration error</b>
\$3A05	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$3A06	TCSETPAR <b>command: parameter not supported</b>
\$3A07	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$3A08	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$3A09	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$3A0A	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$3A0B	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$3A0C	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$3A0D	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$3A0E	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$3A0F	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$3A12	TPCMD <b>command: missing parameter</b>
\$3A13	TPCMD <b>command: parameter not supported</b>
\$3A14	TPCMD RUN <b>command: execution error</b>
\$3A15	TPCMD MASSERASE <b>command: missing parameter</b>
\$3A16	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$3A1D	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$3A1E	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$3A1F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$3A21	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$3A21	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$3A23	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$3A24	TPCMD PROGRAM <b>command: Flash program error</b>
\$3A26	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$3A27	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>

Error #	Description
\$3A28	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$3A29	TPCMD VERIFY <b>command: Flash verify error</b>
\$3A2A	TPCMD READ <b>command: missing or invalid parameter</b>
\$3A2B	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$3A2C	TPCMD READ F <b>command: Flash read error</b>
\$3A30	TPSTART <b>command: execution error</b>
\$3A31	TPCMD PROGRAMMING <b>command: entry error</b>
\$3A32	TPEND <b>command: execution error</b>
\$3A3A	TCSETPAR VDD <b>command: missing parameter</b>
\$3A3B	TCSETPAR VDD_AUX <b>command: missing parameter</b>
\$3A3C	TCSETPAR PWUP <b>command: missing parameter</b>
\$3A3D	TCSETPAR PWDOWN <b>command: missing parameter</b>
\$3A3E	TCSETPAR RSTUP <b>command: missing parameter</b>
\$3A3F	TCSETPAR RSTDOWN <b>command: missing parameter</b>
\$3A40	TCSETPAR CLKOUT <b>command: missing parameter</b>
<b>Infineon XC2000 (DAP protocol) INF_D - Specific Errors</b>	
\$3B00	TCSETDEV <b>command: manufacturer not supported</b>
\$3B01	TCSETDEV <b>command: algorithm not found on card</b>
\$3B02	TCSETDEV <b>command: device not supported</b>
\$3B03	TCSETDEV <b>command: corrupted algorithm file</b>
\$3B04	TCSETDEV <b>command: internal hardware configuration error</b>
\$3B05	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$3B06	TCSETPAR <b>command: parameter not supported</b>
\$3B07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$3B08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$3B09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$3B0A	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$3B0B	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$3B0C	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$3B0D	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$3B0E	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$3B12	TPCMD <b>command: missing parameter</b>
\$3B13	TPCMD <b>command: parameter not supported</b>
\$3B14	TPCMD RUN <b>command: execution error</b>
\$3B15	TPCMD MASSERASE <b>command: missing parameter</b>
\$3B16	TPCMD MASSERASE <b>command: invalid parameter</b>
\$3B17	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$3B18	TPCMD MASSERASE <b>command: device is protected</b>

## Errors

Error #	Description
\$3B1D	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$3B1E	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$3B1F	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$3B20	TPCMD BLANKCHECK <b>command: device is protected</b>
\$3B21	TPCMD PROGRAM <b>command: missing parameter</b>
\$3B22	TPCMD PROGRAM <b>command: invalid parameter</b>
\$3B23	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$3B24	TPCMD PROGRAM <b>command: Flash program error</b>
\$3B25	TPCMD PROGRAM <b>command: device is protected</b>
\$3B26	TPCMD VERIFY <b>command: missing parameter</b>
\$3B27	TPCMD VERIFY <b>command: invalid parameter</b>
\$3B28	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$3B29	TPCMD VERIFY <b>command: Flash verify error</b>
\$3B2A	TPCMD VERIFY <b>command: device is protected</b>
\$3B2B	TPSTART <b>command: execution error</b>
\$3B2C	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$3B2D	TPEND <b>command: execution error</b>
\$3B2E	TCSETPAR PLLNDIV <b>command: missing, invalid or out of range parameter</b>
\$3B2F	TCSETPAR PLLPDIV <b>command: missing, invalid or out of range parameter</b>
\$3B30	TCSETPAR PLLKDIV <b>command: missing, invalid or out of range parameter</b>
\$3B31	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$3B32	TCSETPAR PLLFREQ <b>command: PLLNDIV, PLLPDIV or PLLKDIV parameters have not been set</b>
\$3B33	TCSETPAR DEVICE_STEP <b>command: missing, invalid or out of range parameter</b>
\$3B34	TCSETPAR DAPCLK <b>command: missing, invalid or out of range parameter</b>
\$3B35	TPSTART <b>command: one or more required</b> TCSETPAR DAPCLK <b>commands have not been sent</b>
\$3B36	TCSETPAR BMI_CONF <b>command: missing, invalid or out of range parameter</b>
\$3B38	TPSTART <b>command: one or more required</b> TCSETPAR JTCLK <b>commands have not been sent</b>
\$3B39	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$3B3A	TPSTART <b>command: one or more required</b> TCSETPAR BAUDRATE <b>commands have not been sent</b>
\$3B3B	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$3B3C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$3B3E	TPCMD READ <b>command: missing, invalid or out of range parameter</b>
\$3B3F	TPSTART <b>command: one or more required</b> TCSETPAR DAPCLK <b>commands have not been sent</b>
\$3B40	TPCMD PROGVER <b>command: missing or invalid parameter</b>
\$3B41	TPCMD PROGVER <b>command: target address missing or out of range parameter</b>
\$3B42	TPCMD PROGVER <b>command: source offset parameter out of range</b>
\$3B43	TPCMD PROGVER <b>command: Flash progver execution error</b>
\$3B44	<b>Progver address missing or out of range parameter</b>

<b>Error #</b>	<b>Description</b>
\$3B45	TPCMD DUMP <b>command: algorithm not supported</b>
\$3B46	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$3B47	TPCMD DUMP <b>command: destination missing or invalid parameter</b>
\$3B48	TPCMD DUMP <b>command: missing, invalid or out of range parameter</b>
\$3B49	TPCMD DUMP <b>command: execution error</b>
\$3B4A	TPCMD DUMP <b>command: dump Flash execution error</b>
\$3B4B	TPCMD DUMP <b>command: binary file write byte error</b>
<b>Toshiba TLC8-870/C TSHB_A - Specific Errors</b>	
\$3C00	TCSETDEV <b>command: manufacturer not supported</b>
\$3C01	TCSETDEV <b>command: algorithm not found on card</b>
\$3C02	TCSETDEV <b>command: device not supported</b>
\$3C04	TCSETDEV <b>command: internal hardware configuration error</b>
\$3C05	TCSETDEV <b>command: old internal fw version</b>
\$3C06	TCSETPAR <b>command: not standard parameter</b>
\$3C07	TCSETPAR <b>command: wrong clock out parameter</b>
\$3C08	TCSETPAR <b>command: wrong reset down parameter</b>
\$3C09	TCSETPAR <b>command: wrong reset up parameter</b>
\$3C0A	TCSETPAR <b>command: wrong power down parameter</b>
\$3C0B	TCSETPAR <b>command: wrong power up parameter</b>
\$3C0C	TCSETPAR <b>command: wrong vdd parameter</b>
\$3C0D	TCSETPAR <b>command: wrong auxiliary vdd parameter</b>
\$3C0E	TCSETPAR <b>command: wrong reset drive parameter</b>
\$3C0F	TCSETPAR <b>command: baudrate value missing</b>
\$3C10	TCSETPAR <b>command: wrong baudrate value</b>
\$3C11	TCSETPAR <b>command: wrong fosc clock parameter</b>
\$3C12	TCSETPAR <b>command: wrong Password Count Storage Address parameter</b>
\$3C13	TCSETPAR <b>command: wrong Password Comparison Start Address parameter</b>
\$3C14	TCSETPAR <b>command: wrong Password Values size parameter</b>
\$3C15	TCSETPAR <b>command: wrong Password Values value parameter</b>
\$3C1A	TCSETPAR <b>command: vdd parameter missing</b>
\$3C1B	TCSETPAR <b>command: auxiliary vdd parameter missing</b>
\$3C1C	TCSETPAR <b>command: pwup parameter missing</b>
\$3C1D	TCSETPAR <b>command: pwdown parameter missing</b>
\$3C1E	TCSETPAR <b>command: rstup parameter missing</b>
\$3C1F	TCSETPAR <b>command: rstdown parameter missing</b>
\$3C20	TCSETPAR <b>command: clockout parameter missing</b>
\$3C21	TCSETPAR <b>command: fosc parameter missing</b>
\$3C22	TCSETPAR <b>command: Password Count Storage Address parameter missing</b>

## Errors

Error #	Description
\$3C23	TCSETPAR <b>command: Comparison Start parameter missing</b>
\$3C24	TCSETPAR <b>command: Password Value not set</b>
\$3C30	TPCMD <b>command: parameter missing</b>
\$3C31	TPCMD <b>command: parameter not supported</b>
\$3C32	TPCMD RUN <b>command: parameter not supported</b>
\$3C33	TPCMD MASSERASE <b>command: masserase syntax error</b>
\$3C34	TPCMD MASSERASE <b>command: masserase execution error</b>
\$3C35	TPCMD ERASE <b>command: erase syntax error</b>
\$3C36	TPCMD ERASE <b>command: erase memory parameter error</b>
\$3C37	TPCMD ERASE <b>command: erase memory flash execution error</b>
\$3C38	TPCMD BLANKCHECK <b>command: blankcheck syntax error</b>
\$3C39	TPCMD BLANKCHECK <b>command: blankcheck memory parameter error</b>
\$3C3A	TPCMD BLANKCHECK <b>command: blankcheck execution error</b>
\$3C3B	TPCMD PROGRAM <b>command: program syntax error</b>
\$3C3C	TPCMD PROGRAM <b>command: memory parameter error</b>
\$3C3D	TPCMD PROGRAM <b>command: source file error</b>
\$3C3E	TPCMD PROGRAM <b>command: program execution error</b>
\$3C3F	TPCMD VERIFY <b>command: verify syntax error</b>
\$3C40	TPCMD VERIFY <b>command: verify memory parameter error</b>
\$3C41	TPCMD VERIFY <b>command: source file error</b>
\$3C42	TPCMD VERIFY <b>command: verify execution error</b>
\$3C43	TPCMD READ <b>command: read syntax error</b>
\$3C44	TPCMD READ <b>command: memory parameter error</b>
\$3C45	TPCMD READ <b>command: read execution error</b>
\$3C46	TPCMD SECURE <b>command: secure execution error</b>
\$3C4A	TFSTART <b>command: execution error</b>
\$3C4B	TPCMD <b>command: command entry error</b>
\$3C4C	TFEND <b>command: execution error</b>
<b>STMicroelectronics STM32 (SWD interface) CORTEX_B - Specific Errors</b>	
\$3D00	TCSETDEV <b>command: manufacturer not supported</b>
\$3D01	TCSETDEV <b>command: algorithm not found on card</b>
\$3D02	TCSETDEV <b>command: device not supported</b>
\$3D03	TCSETDEV <b>command: internal hardware configuration error</b>
\$3D04	TCSETDEV <b>command: corrupted algorithm file</b>
\$3D05	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$3D06	TCSETPAR <b>command: parameter not supported</b>
\$3D07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$3D08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$3D09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$3D0A	TCSETPAR RSTDRV <b>command: missing or invalid command option</b>
\$3D0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$3D0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$3D0D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$3D0E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$3D0F	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$3D10	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$3D11	TPCMD <b>command: missing parameter</b>
\$3D12	TPCMD <b>command: parameter not supported</b>
\$3D13	TPCMD RUN <b>command: execution error</b>
\$3D14	TPCMD MASSERASE <b>command: missing parameter</b>
\$3D15	TPCMD MASSERASE <b>command: invalid parameter</b>
\$3D16	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$3D17	TPCMD MASSERASE <b>command: Option Bytes masserase error</b>
\$3D18	TPCMD ERASE <b>command: missing parameter</b>
\$3D19	TPCMD ERASE <b>command: invalid parameter</b>
\$3D1A	TPCMD ERASE <b>command: Flash erase error</b>
\$3D1B	TPCMD ERASE <b>command: Flash device secured</b>
\$3D1C	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$3D1D	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$3D1E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$3D1F	TPCMD BLANKCHECK <b>command: Option Bytes blankcheck error</b>
\$3D20	TPCMD PROGRAM <b>command: missing parameter</b>
\$3D21	TPCMD PROGRAM <b>command: invalid parameter</b>
\$3D22	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$3D23	TPCMD PROGRAM <b>command: Flash program error</b>
\$3D24	TPCMD PROGRAM <b>command: Option Bytes program error</b>
\$3D25	TPCMD VERIFY <b>command: missing parameter</b>
\$3D26	TPCMD VERIFY <b>command: invalid parameter</b>
\$3D27	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$3D28	TPCMD VERIFY <b>command: Flash verify error</b>
\$3D29	TPCMD VERIFY <b>command: Option Bytes verify error</b>
\$3D2A	TPCMD READ <b>command: missing or invalid parameter</b>
\$3D2B	TPCMD READ <b>command: Flash read error</b>
\$3D2C	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$3D2E	TPCMD WRITE <b>command: Flash write error</b>
\$3D2F	TPSTART <b>command: execution error</b>

## Errors

Error #	Description
\$3D30	TFSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$3D31	TFEND <b>command: execution error</b>
\$3D32	TPCMD MASSERASE <b>command: EEPROM masserase error</b>
\$3D32	TCSETPAR RSTTYPE <b>command: missing or invalid command option</b>
\$3D33	TPCMD PROGRAM <b>command: command: EEPROM program error</b>
\$3D34	TCSETPAR RSTTYPE <b>command: missing or invalid command option</b>
\$3D34	TPCMD DUMP <b>command: missing parameter</b>
\$3D35	TPCMD DUMP <b>command: Algorithm not supported</b>
\$3D36	TPCMD DUMP <b>command: Algorithm not supported</b>
\$3D37	TPCMD DUMP <b>command: invalid parameter</b>
\$3D3F	TPCMD DUMP <b>command: Flash device protected</b>
<b>Texas Instruments CC2533 TI_C - Specific Errors</b>	
\$3E00	TCSETPAR <b>command: manufacturer not supported</b>
\$3E01	TCSETPAR <b>command: algorithm not found on card</b>
\$3E02	TCSETPAR <b>command: device not supported</b>
\$3E03	TCSETPAR <b>command: internal hardware configuration error</b>
\$3E04	TCSETPAR <b>command: algorithm not supported</b>
\$3E05	TPCMD <b>command: invalid command</b>
\$3E07	TPCMD <b>command: Error on reading chip ID</b>
\$3E08	TPCMD <b>command: wrong chip ID</b>
\$3E09	TCSETPAR <b>command: missing or invalid parameter</b>
\$3E0A	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$3E0B	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$3E0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$3E0D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$3E0F	TCSETPAR SCLK <b>command: missing or invalid command parameter</b>
\$3E10	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$3E11	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$3E12	TCSETPAR VDD <b>command: missing parameter</b>
\$3E13	TCSETPAR PWUP <b>command: missing parameter</b>
\$3E14	TCSETPAR PWDOWN <b>command: missing parameter</b>
\$3E15	TCSETPAR SCLK <b>command: missing parameter</b>
\$3E16	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$3E17	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$3E18	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$3E19	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$3E1A	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$3E1B	TPCMD PROGRAM <b>command: missing or invalid parameter</b>



Error #	Description
\$3E1C	TPCMD PROGRAM <b>command: invalid parameter</b>
\$3E1D	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$3E1E	TPCMD PROGRAM <b>command: Flash program error</b>
\$3E1F	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$3E20	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$3E21	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$3E22	TPCMD VERIFY <b>command: Flash verify error</b>
\$3E23	TPCMD RUN <b>command: execution error</b>
\$3E24	TPCMD RUN <b>command: Run execution error</b>
\$3E30	TPCMD READ <b>command: missing or invalid parameter</b>
\$3E31	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$3E32	TPCMD READ <b>command: Flash read error</b>
\$3E33	TPSTART <b>command: execution error</b>
\$3E34	TPEND <b>command: execution error</b>
\$3E54	TPCMD UNLOCK <b>command: Flash unlock error</b>
\$3E35	TPCMD DBG_ENTRY <b>command: debug entry error</b>
<b>Serial Memories (I2C Interface) SERMEM_B - Specific Errors</b>	
\$3F00	TCSETDEV <b>command: manufacturer not supported</b>
\$3F01	TCSETDEV <b>command: algorithm not found on card</b>
\$3F02	TCSETDEV <b>command: device not supported</b>
\$3F03	TCSETDEV <b>command: internal hardware configuration error</b>
\$3F04	TCSETPAR <b>command: parameter not supported</b>
\$3F05	TCSETPAR SCLK <b>command: invalid command parameter</b>
\$3F06	TCSETPAR SCLK <b>command: missing parameter</b>
\$3F07	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$3F08	TCSETPAR VDD <b>command: missing parameter</b>
\$3F09	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$3F0A	TCSETPAR VDD_AUX <b>command: missing parameter</b>
\$3F0D	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$3F0E	TCSETPAR PWDOWN <b>command: missing parameter</b>
\$3F0F	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$3F10	TCSETPAR PWUP <b>command: missing parameter</b>
\$3F11	TPCMD <b>command: invalid command</b>
\$3F12	TPCMD <b>command: entry error</b>
\$3F13	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$3F14	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$3F15	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$3F16	TPCMD VERIFY <b>command: Flash verify error</b>

## Errors

Error #	Description
\$3F18	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$3F19	TPCMD PROGRAM <b>command: invalid parameter</b>
\$3F1A	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$3F1B	TPCMD PROGRAM <b>command: Flash program error</b>
\$3F1D	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$3F1E	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$3F1F	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$3F20	TPCMD READ <b>command: missing or invalid parameter</b>
\$3F21	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$3F22	TPCMD READ <b>command: Flash read error</b>
\$3F31	TPSTART <b>command: execution error</b>
\$3F32	TPEND <b>command: target off error</b>
\$3F33	TPEND <b>command: execution error</b>
\$3F34	TCSETPAR <b>DEVICE_ADDR</b> <b>command: invalid or out of range parameter</b>
\$3F35	TCSETPAR <b>DEVICE_ADDR</b> <b>command: missing parameter</b>
\$3F36	TCSETPAR <b>DATAWIDTH</b> <b>command: data width parameter not supported</b>
\$3F37	TCSETPAR <b>DEVICE_ADDR</b> <b>command: device address parameter not supported</b>
\$3F41	TCSETPAR <b>PAGESIZE</b> <b>command: invalid or out of range parameter</b>
\$3F42	TCSETPAR <b>PAGESIZE</b> <b>command: page size parameter not supported</b>
\$3F43	TPSTART <b>command: one or more required</b> TCSETPAR <b>PAGESIZE</b> <b>commands have not been sent</b>
\$3F44	TPCMD <b>BLOCK_ERASE</b> <b>command: block erase not supported</b>
\$3F4B	TPCMD PROGRAM <b>command: program INFO execution error</b>
\$3F4C	TPCMD READ <b>command: read LR execution error</b>
\$3F4D	TPCMD PROGRAM <b>command: program LR execution error</b>
\$3F4E	TCSETPAR <b>SKIP_IDCODE_CHECK</b> <b>command: missing or invalid parameter</b>
\$3F4F	TCSETDEV <b>command: algorithm not supported</b>
\$3F50	TPCMD READ <b>command: ID Page read error</b>
<b>Texas Instruments TI_D - Specific Errors</b>	
\$4000	TCSETDEV <b>command: manufacturer not supported</b>
\$4001	TCSETDEV <b>command: algorithm not found on card</b>
\$4002	TCSETDEV <b>command: device not supported</b>
\$4003	TCSETDEV <b>command: internal hardware configuration error</b>
\$4004	TCSETPAR <b>command: parameter not supported</b>
\$4005	TCSETPAR <b>command: wrong VDD parameter</b>
\$4006	TCSETPAR <b>command: VDD missing parameter</b>
\$4007	TCSETPAR <b>command: wrong power up parameter</b>
\$4008	TCSETPAR <b>command: power up missing parameter</b>
\$4009	TCSETPAR <b>command: wrong power down parameter</b>

Error #	Description
\$400A	TCSETPAR <b>command: power down missing parameter</b>
\$400D	TCSETPAR <b>command: wrong reset down parameter</b>
\$400E	TCSETPAR <b>command: missing reset down parameter</b>
\$400F	TCSETPAR <b>command: wrong serial clock parameter</b>
\$4010	TCSETPAR <b>command: missing serial clock parameter</b>
\$4011	TCSETPAR <b>command: wrong VDD Aux parameter</b>
\$4016	TPCMD <b>command: command not recognized</b>
\$4017	TPCMD MASSERASE <b>command: invalid command/parameter</b>
\$4018	TPCMD MASSERASE <b>command: masserase execution command</b>
\$4019	TPCMD BLANKCHECK <b>command: blankcheck syntax error</b>
\$4020	TPCMD BLANKCHECK <b>command: blankcheck memory parameter error</b>
\$4021	TPCMD BLANKCHECK <b>command: blankcheck execution error</b>
\$4024	TPCMD PROGRAM <b>command: program syntax error</b>
\$4025	TPCMD PROGRAM <b>command: memory parameter error</b>
\$4026	TPCMD PROGRAM <b>command: source file error</b>
\$4027	TPCMD PROGRAM <b>command: memory size error</b>
\$4030	TPCMD VERIFY <b>command: verify syntax error</b>
\$4031	TPCMD VERIFY <b>command: verify memory parameter error</b>
\$4032	TPCMD VERIFY <b>command: source file error</b>
\$4033	TPCMD VERIFY <b>command: verify memory size error</b>
\$403C	TPCMD READ <b>command: missing or invalid parameter</b>
\$403D	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$403E	TPCMD READ <b>command: read error</b>
\$403F	TPCMD ERASE <b>command: erase syntax error</b>
\$4040	TPCMD ERASE <b>command: erase execution error</b>
\$4045	TPCMD RUN <b>command: run execution error</b>
\$4046	TPSTART <b>command: start execution error</b>
\$4047	TPEND <b>command: end execution error</b>
\$4048	TPCMD ENTRY <b>command: entry execution error</b>
<b>NEC V850_B - Specific Errors</b>	
\$4200	TCSETDEV <b>command: manufacturer not supported</b>
\$4201	TCSETDEV <b>command: algorithm not found on card</b>
\$4202	TCSETDEV <b>command: device not supported</b>
\$4203	TCSETDEV <b>command: internal hardware configuration error</b>
\$4204	TCSETPAR <b>command: parameter not supported</b>
\$4205	TCSETPAR CMODE <b>command: missing parameter</b>
\$4206	TCSETPAR CMODE <b>command: communication mode not supported by target device</b>
\$4207	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>

## Errors

Error #	Description
\$4208	TFSTART CMODE <b>command: invalid parameter</b>
\$4209	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$420A	TCSETPAR CLKOUT <b>command: invalid frequency</b>
\$420B	TFSTART <b>command: command not preceded by</b> TCSETPAR CLKOUT <b>command</b>
\$420C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$420D	TFSTART <b>command: command not preceded by</b> TCSETPAR FOSC <b>command</b>
\$420E	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$420F	TFSTART <b>command: command not preceded by</b> TCSETPAR SCLK <b>command</b>
\$4210	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4211	TFSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$4212	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4213	TFSTART <b>command: command not preceded by</b> TCSETPAR VDD_AUX <b>command</b>
\$4214	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4215	TFSTART <b>command: command not preceded by</b> TCSETPAR RSTDOWN <b>command</b>
\$4216	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4217	TFSTART <b>command: command not preceded by</b> TCSETPAR RSTUP <b>command</b>
\$4218	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4219	TFSTART <b>command: command not preceded by</b> TCSETPAR PWDOWN <b>command</b>
\$421A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$421B	TFSTART <b>command: command not preceded by</b> TCSETPAR PWUP <b>command</b>
\$421C	TPCMD <b>command: parameter not supported</b>
\$421D	<b>Can't establish communication with the target device</b>
\$421E	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$421F	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4220	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4221	TPCMD VERIFY <b>command: Flash verify error</b>
\$4223	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$4224	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$4225	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4226	TPCMD PROGRAM <b>command: Flash program error</b>
\$4227	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$4228	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$4229	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$422A	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$422B	TPCMD PROTECT <b>command: missing, invalid or out of range parameter(s)</b>
\$422C	TPCMD PROTECT <b>command: chip protection error</b>
\$422D	TPCMD PROTECT <b>command: operation not supported by target device</b>
\$422E	TPCMD BLOCKERASE <b>command: missing or invalid parameter</b>

Error #	Description
\$422F	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$4230	TPCMD BLOCKERASE <b>command: Flash block erase error</b>
\$4231	TPCMD BLOCKERASE <b>command: operation not supported by target device</b>
\$4232	TPCMD RUN <b>command: execution error</b>
\$4233	TPSTART <b>command: execution error</b>
\$4234	TPEND <b>command: execution error</b>
\$4235	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$4236	TPSTART <b>command: command not preceded by TCSETPAR BAUDRATE command</b>
\$4237	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$4238	TPCMD READ <b>command: missing or invalid parameter</b>
\$4239	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$423A	TPCMD READ <b>command: operation not supported by target device</b>
\$423B	TPCMD READ <b>command: read error</b>
\$423C	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$423D	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$423E	TPCMD BLANKCHECK <b>command: EEPROM blank check error</b>
\$423F	TPCMD MASSERASE <b>command: EEPROM mass erase error</b>
\$4240	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$4241	TPCMD BLOCKERASE <b>command: EEPROM block erase error</b>
\$4242	TPCMD READ <b>command: EEPROM read error</b>
\$4243	TPCMD PROGRAM <b>command: Option Byte program error</b>
\$4244	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$4245	TCSETPAR SIPU <b>command: serial input Pull-Up error</b>
\$4246	TCSETPAR SOPU <b>command: serial output Pull-Up error</b>
\$4247	TCSETDEV OS <b>command: OS version old</b>
\$4248	TPCMD UNPROTECT <b>command: unprotect flash memory error</b>
<b>Renesas RX62 RNSS_E - Specific Errors</b>	
\$4300	TCSETDEV <b>command: manufacturer not supported</b>
\$4301	TCSETDEV <b>command: algorithm not found on card</b>
\$4302	TCSETDEV <b>command: device not supported</b>
\$4303	TCSETDEV <b>command: corrupted algorithm file</b>
\$4304	TCSETDEV <b>command: algo not supported</b>
\$4305	TCSETDEV <b>command: hw configuration error</b>
\$4306	TCSETPAR <b>command: parameter error</b>
\$4307	TCSETPAR <b>command: VDD Value error</b>
\$4308	TCSETPAR <b>command: missing VDD Value</b>
\$4309	TCSETPAR PWUP <b>command: out of range parameter</b>
\$430A	TCSETPAR PWUP <b>command: missing</b>

## Errors

---

Error #	Description
\$430B	TCSETPAR PWDOWN <b>command: out of range parameter</b>
\$430C	TCSETPAR PWDOWN <b>command: missing</b>
\$430D	TCSETPAR RSTUP <b>command: out of range parameter</b>
\$430E	TCSETPAR RSTUP <b>command: missing</b>
\$430F	TCSETPAR RSTDOWN <b>command: out of range parameter</b>
\$4310	TCSETPAR RSTDOWN <b>command: missing</b>
\$4311	TCSETPAR PWUP DELAY <b>command: invalid value</b>
\$4312	TCSETPAR RST DELAY <b>command: invalid value</b>
\$4313	TCSETPAR FOSC <b>command: invalid value</b>
\$4314	TCSETPAR FOSC <b>command: missing value</b>
\$4315	TCSETPAR KEYCODE <b>command: invalid value</b>
\$4316	TCSETPAR KEYCODE <b>command: missing value</b>
\$4317	TPCMD <b>command: parameter error</b>
\$4318	TPCMD JTCLK <b>command: missing parameter</b>
\$4319	TPCMD JTCLK <b>command: setting error</b>
\$431A	TPCMD JTCLK <b>command: invalid parameter</b>
\$431B	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$431C	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$431D	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$431E	TPCMD VERIFY <b>command: User Flash verify error</b>
\$431F	TPCMD VERIFY <b>command: EEPROM verify error</b>
\$4320	TPCMD VERIFY <b>command: ID CODE Error</b>
\$4321	TPCMD READ <b>command: missing parameter</b>
\$4322	TPCMD READ <b>command: invalid parameter</b>
\$4323	TPCMD READ <b>command: read flash process error</b>
\$4324	TPCMD READ <b>command: read eeprom process error</b>
\$4325	TPCMD READ <b>command: read IDCODE process error</b>
\$4326	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$4327	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$4328	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4329	TPCMD PROGRAM <b>command: User Flash program error</b>
\$432A	TPCMD PROGRAM <b>command: EEPROM Flash program error</b>
\$432B	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$432C	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$432D	TPCMD BLANKCHECK <b>command: User Flash blank check error</b>
\$432E	TPCMD BLANKCHECK <b>command: EEPROM Flash blank check error</b>
\$432F	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$4330	TPCMD MASSERASE <b>command: User Flash mass erase error</b>

Error #	Description
\$4331	TPCMD MASSERASE <b>command: EEPROM Flash mass erase error</b>
\$4332	TPCMD BLOCKERASE <b>command: missing parameter</b>
\$4333	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$4334	TPCMD BLOCKERASE <b>command: User Flash block erase error</b>
\$4335	TPCMD BLOCKERASE <b>command: Data Flash block erase error</b>
\$4336	TPSTART <b>command: execution error</b>
\$4337	TPEND <b>command: execution error</b>
\$4338	TPCMD RUN <b>command: execution error</b>
\$4339	TPCMD READ32 <b>command: missing parameter</b>
\$433A	TPCMD READ32 <b>command: target start address parameter or length parameter out of range</b>
\$433B	WRONG ID CODE <b>mismatching ID code</b>
\$433D	TPCMD DBG <b>command: execution error</b>
\$433E	TPCMD MASSERASE <b>command: User boot area mass erase error</b>
\$433F	<b>Check UBA memory address</b>
\$4340	TPCMD BLANKCHECK <b>command: User boot area blank check error</b>
\$4341	TPCMD PROGRAM <b>command: User boot area program error</b>
\$4342	TPCMD VERIFY <b>command: User boot area verify error</b>
<b>NXP P89C51R NXP_B - Specific Errors</b>	
\$4500	TCSETDEV <b>command: manufacturer not supported</b>
\$4501	TCSETDEV <b>command: algorithm not found on card</b>
\$4502	TCSETDEV <b>command: device not supported</b>
\$4503	TCSETDEV <b>command: internal hardware configuration error</b>
\$4504	TCSETDEV <b>command: load FPGA</b>
\$4505	TCSETDEV <b>command: algorithm not supported</b>
\$4506	TCSETPAR <b>command: invalid command</b>
\$4507	TCSETPAR CLOCKOUT <b>command: missing, invalid or out of range parameter</b>
\$4508	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4509	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$450A	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$450B	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$450C	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$450D	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$450E	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$450F	TCSETPAR BAUDRATE <b>command: missing or invalid parameter</b>
\$4510	TCSETPAR BAUDRATE <b>command wrong parameter</b>
\$4511	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$451A	TCSETPAR VDD <b>command: missing parameter</b>
\$451B	TCSETPAR VDD_AUX <b>command: missing parameter</b>

## Errors

Error #	Description
\$451C	TCSETPAR PWUP <b>command: missing parameter</b>
\$451D	TCSETPAR PWDOWN <b>command: missing parameter</b>
\$451E	TCSETPAR RSTUP <b>command: missing parameter</b>
\$451F	TCSETPAR RSTDOWN <b>command: missing parameter</b>
\$4520	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$4521	TCSETPAR FOSC <b>command: missing parameter</b>
\$4531	TPCMD <b>command: invalid command</b>
\$4532	TPCMD RUN <b>command: execution error</b>
\$4533	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$4534	TPCMD MASSERASE <b>command: Flash masserese error</b>
\$4535	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$4536	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$4537	TPCMD ERASE <b>command: Flash erase error</b>
\$4538	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$4539	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$453A	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$453B	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$453C	TPCMD PROGRAM <b>command: invalid parameter</b>
\$453D	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$453E	TPCMD PROGRAM <b>command: Flash program error</b>
\$453F	TPCMD PROGRAM <b>command: Program registers error</b>
\$4546	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$4547	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4548	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4549	TPCMD VERIFY <b>command: Flash verify error</b>
\$454A	TPCMD READ <b>command: missing or invalid parameter</b>
\$454B	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$454C	TPCMD READ <b>command: Flash read error</b>
\$454D	TPCMD READ <b>command: read registers error</b>
\$4550	TFSTART <b>command: execution error</b>
\$4551	TPCMD <b>command: UART entry error</b>
\$4552	TFEND <b>command: execution error</b>
<b>Infineon XC800 (DAP and UART 1 Wire protocols) INF_E - Specific Errors</b>	
\$4600	TCSETDEV <b>command: manufacturer not supported</b>
\$4601	TCSETDEV <b>command: algorithm not found on card</b>
\$4602	TCSETDEV <b>command: device not supported</b>
\$4603	TCSETDEV <b>command: internal hardware configuration error</b>
\$4604	TCSETDEV <b>command: corrupted algorithm file</b>



Error #	Description
\$4605	TCSETPAR <b>command: parameter not supported</b>
\$4606	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$4607	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4608	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4609	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$460A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$460B	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$460C	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$460D	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter (DAP protocol)</b> TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter (UART protocol)</b>
\$460E	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$460F	TPCMD <b>command: missing parameter</b>
\$4610	TPCMD <b>command: parameter not supported</b>
\$4611	TPCMD RUN <b>command: execution error</b>
\$4612	TPCMD MASSERASE <b>command: missing parameter</b>
\$4613	TPCMD MASSERASE <b>command: invalid parameter</b>
\$4614	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$4615	TPCMD MASSERASE <b>command: device is protected</b>
\$4616	TPCMD SECTORERASE <b>command: missing parameter</b>
\$4617	TPCMD SECTORERASE <b>command: invalid parameter</b>
\$4618	TPCMD SECTORERASE <b>command: Flash sector erase error</b>
\$4619	TPCMD SECTORERASE <b>command: device is protected</b>
\$461A	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$461B	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$461C	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$461D	TPCMD BLANKCHECK <b>command: device is protected</b>
\$461E	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$461F	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$4620	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4621	TPCMD PROGRAM <b>command: Flash program error</b>
\$4622	TPCMD PROGRAM <b>command: device is protected</b>
\$4623	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$4624	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4625	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4626	TPCMD VERIFY <b>command: Flash verify error</b>
\$4627	TPCMD VERIFY <b>command: device is protected</b>
\$4628	TPSTART <b>command: execution error</b>
\$4629	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$462A	TPEND <b>command: execution error</b>

## Errors

Error #	Description
\$462B	TPCMD READ <b>command: missing or invalid parameter</b>
\$462C	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$462D	TPCMD READ <b>command: read error</b>
\$462E	TPCMD USER_ID_CONFIG <b>command: missing or invalid parameter (UART protocol)</b>
\$462F	TPCMD USER_ID_CONFIG <b>command: User ID configuration error (UART protocol)</b>
\$4630	TCSETPAR CMODE <b>command: missing or invalid parameter (UART protocol)</b>
\$4631	TPCMD USER_ID_READ <b>command: missing or invalid parameter (UART protocol)</b>
<b>Texas Instruments TMS570 - Specific Errors</b>	
\$4800	TCSETDEV <b>command: manufacturer not supported</b>
\$4801	TCSETDEV <b>command: algorithm not found on card</b>
\$4802	TCSETDEV <b>command: device not supported</b>
\$4803	TCSETDEV <b>command: internal hardware configuration error</b>
\$4804	TCSETDEV <b>command: internal kernel error</b>
\$4805	TCSETDEV <b>command: old internal fw version</b>
\$4806	TCSETPAR <b>command: not standard parameter</b>
\$4807	TCSETPAR <b>command: wrong clock out parameter</b>
\$4808	TCSETPAR <b>command: wrong reset down parameter</b>
\$4809	TCSETPAR <b>command: wrong reset up parameter</b>
\$480A	TCSETPAR <b>command: wrong reset drive parameter</b>
\$480B	TCSETPAR <b>command: wrong power down parameter</b>
\$480C	TCSETPAR <b>command: wrong power up parameter</b>
\$480D	TCSETPAR <b>command: wrong VDD parameter</b>
\$480E	TCSETPAR <b>command: wrong VDD Aux parameter</b>
\$480F	TCSETPAR <b>command: wrong CPU Frequency parameter</b>
\$4810	TCSETPAR <b>command: wrong JTAG Clock parameter</b>
\$4811	TPCMD <b>command: invalid/missing command</b>
\$4812	TPCMD <b>command: command not recognized</b>
\$4813	TPCMD <b>command: low level hw error</b>
\$4814	TPCMD MASSERASE <b>command: missing command</b>
\$4815	TPCMD MASSERASE <b>command: invalid command/parameter</b>
\$4816	TPCMD MASSERASE <b>command: Flash masserase execution command</b>
\$4817	TPCMD MASSERASE <b>command: Option byte masserase execution command</b>
\$481C	TPCMD BLANKCHECK <b>command: blankcheck syntax error</b>
\$481D	TPCMD BLANKCHECK <b>command: blankcheck memory parameter error</b>
\$481E	TPCMD BLANKCHECK <b>command: Flash blankcheck execution error</b>
\$481F	TPCMD BLANKCHECK <b>command: Option byte blankcheck execution error</b>
\$4820	TPCMD PROGRAM <b>command: program syntax error</b>
\$4821	TPCMD PROGRAM <b>command: memory parameter error</b>

Error #	Description
\$4822	TPCMD PROGRAM <b>command: source file error</b>
\$4823	TPCMD PROGRAM <b>command: memory size error</b>
\$4824	TPCMD PROGRAM <b>command: Option bytes program error</b>
\$4825	TPCMD VERIFY <b>command: verify syntax error</b>
\$4826	TPCMD VERIFY <b>command: verify memory parameter error</b>
\$4827	TPCMD VERIFY <b>command: source file error</b>
\$4828	TPCMD VERIFY <b>command: Flash verify memory size error</b>
\$4829	TPCMD VERIFY <b>command: Option bytes verify memory size error</b>
\$482A	TPCMD READ <b>command: read syntax error</b>
\$482B	TPCMD READ <b>command: read execution error</b>
\$482C	TPCMD WRITE <b>command: sintax error</b>
\$482D	TPCMD WRITE <b>command: execution error</b>
\$482E	TPCMD DUMP <b>command: sintax error</b>
\$482F	TPCMD DUMP <b>command: execution error</b>
\$4830	TPCMD TPSTART <b>command: unable to execute start</b>
\$4831	TPCMD TPSTART <b>command: execution error</b>
\$4832	TPCMD TPEND <b>command: execution error</b>
\$4832	TCSETPAR <b>command: wrong reset type parameter</b>
<b>Renesas RL78 RNSS_F - Specific Errors</b>	
\$4A00	TCSETDEV <b>command: manufacturer not supported</b>
\$4A01	TCSETDEV <b>command: algorithm not found on card</b>
\$4A02	TCSETDEV <b>command: device not supported</b>
\$4A03	TCSETDEV <b>command: internal hardware configuration error</b>
\$4A04	TCSETPAR <b>command: parameter not supported</b>
\$4A05	TCSETPAR CMODE <b>command: missing parameter</b>
\$4A06	TCSETPAR CMODE <b>command: communication mode not supported by target device</b>
\$4A07	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>
\$4A08	TPSTART CMODE <b>command: invalid parameter</b>
\$4A09	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$4A0A	TCSETPAR CLKOUT <b>command: invalid frequency</b>
\$4A0B	TPSTART <b>command: command not preceded by TCSETPAR CLKOUT command</b>
\$4A0C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$4A0D	TPSTART <b>command: command not preceded by TCSETPAR FOSC command</b>
\$4A10	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4A11	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$4A12	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4A13	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$4A14	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4A15	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>

## Errors

Error #	Description
\$4A16	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4A17	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$4A18	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4A19	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$4A1A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$4A1B	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$4A1C	TPCMD <b>command: parameter not supported</b>
\$4A1D	<b>Can't establish communication with the target device</b>
\$4A1E	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$4A1F	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4A20	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4A21	TPCMD VERIFY <b>command: Flash verify error</b>
\$4A22	TPCMD VERIFY <b>command: operation not supported by target device</b>
\$4A23	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$4A24	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$4A25	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4A26	TPCMD PROGRAM <b>command: Flash program error</b>
\$4A27	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$4A28	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$4A29	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$4A2A	TPCMD BLANKCHECK <b>command: operation not supported by target device</b>
\$4A2B	TPCMD MASSERASE <b>command: chip erase error</b>
\$4A2C	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$4A2D	TPCMD PROTECT <b>command: chip protection error</b>
\$4A2E	TPCMD PROTECT <b>command: operation not supported by target device</b>
\$4A2F	TPCMD BLOCKERASE <b>command: missing or invalid parameter</b>
\$4A30	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$4A31	TPCMD BLOCKERASE <b>command: block erase error</b>
\$4A32	TPCMD BLOCKERASE <b>command: operation not supported by target device</b>
\$4A33	TPCMD RUN <b>command: execution error</b>
\$4A34	TPSTART <b>command: execution error</b>
\$4A35	TPEND <b>command: could not power target off</b>
\$4A36	TPEND <b>command: execution error</b>
\$4A37	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$4A38	TPSTART <b>command: command not preceded by TCSETPAR BAUDRATE command</b>
\$4A39	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$4A40	TCSETPAR VDD AUX <b>command: command not allowed</b>
\$4A41	TCSETDEV <b>command: OS version not supported</b>

Error #	Description
\$4A42	TPCMD GETSIGNATURE <b>command: get signature not supported</b>
\$4A43	TPCMD GETSIGNATURE <b>command: get signature error</b>
\$4A44	TPCMD PROGRAMBYTE <b>command: program byte not supported</b>
\$4A45	TPCMD PROGRAMBYTE <b>command: program byte error</b>
\$4A46	TPCMD GETSECURITY <b>command: getsecurity error</b>
\$4A47	TPCMD GETSECURITY <b>command: operation not supported by target device</b>
\$4A48	TPCMD SECURITYRELEASE <b>command: security error</b>
\$4A49	TPCMD SECURITYRELEASE <b>command: operation not supported by target device</b>
\$4A4A	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$4A4B	TPCMD CHECKSUM <b>command: target start address parameter or length parameter out of range</b>
\$4A4C	TPCMD CHECKSUM <b>command: check sum error</b>
\$4A4D	TPCMD CHECKSUM <b>command: check sum not supported</b>
<b>NXP LPC122x - Specific Errors</b>	
\$4B00	TCSETDEV <b>command: manufacturer not supported</b>
\$4B01	TCSETDEV <b>command: algorithm not found on card</b>
\$4B02	TCSETDEV <b>command: device not supported</b>
\$4B03	TCSETDEV <b>command: internal hardware configuration error</b>
\$4B04	TCSETDEV <b>command: device invalid data</b>
\$4B05	TCSETDEV <b>command: algorithm not supported</b>
\$4B06	TCSETPAR <b>command: parameter not supported</b>
\$4B07	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$4B08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4B09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4B0A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$4B0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4B0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$4B0D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4B0E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4B10	TCSETPAR SWCLK <b>command: missing, invalid or out of range parameter</b>
\$4B11	TPCMD <b>command: parameter not supported</b>
\$4B12	TPCMD <b>command: invalid command</b>
\$4B13	TPCMD RUN <b>command: execution error</b>
\$4B14	TPCMD MASSERASE <b>command: missing parameter</b>
\$4B15	TPCMD MASSERASE <b>command: invalid parameter</b>
\$4B16	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$4B1C	TPCMD BLANKCHECK/SECTOR_BLANKCHECK <b>command: missing or invalid parameter</b>
\$4B1D	TPCMD BLANKCHECK/SECTOR_BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$4B1E	TPCMD BLANKCHECK/SECTOR_BLANKCHECK <b>command: Flash blank check error</b>

## Errors

Error #	Description
\$4B20	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$4B21	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$4B22	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4B23	TPCMD PROGRAM <b>command: Flash program error</b>
\$4B25	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$4B26	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4B27	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4B28	TPCMD VERIFY <b>command: Flash verify error</b>
\$4B2A	TPCMD READ <b>command: missing or invalid parameter</b>
\$4B2B	TPCMD READ <b>command: Flash read error</b>
\$4B2F	TPSTART <b>command: execution error</b>
\$4B30	TPSTART <b>command: standard algorithm or device specific parameter missing</b>
\$4B31	TPEND <b>command: execution error</b>
<b>Freescal Kinetic FSL_K - Specific Errors</b>	
\$4C00	TCSETDEV <b>command: manufacturer not supported</b>
\$4C01	TCSETDEV <b>command: algorithm not found on card</b>
\$4C02	TCSETDEV <b>command: device not supported</b>
\$4C03	TCSETDEV <b>command: load FPGA</b>
\$4C04	TCSETDEV <b>command: internal hardware configuration error</b>
\$4C05	TCSETDEV <b>command: algorithm not supported</b>
\$4C06	TCSETPAR <b>command: invalid command</b>
\$4C07	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$4C08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4C09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4C0A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$4C0B	TCSETPAR RSTTYPE <b>command: missing or invalid parameter</b>
\$4C0C	TCSETPAR CMODE <b>command: missing or invalid parameter</b>
\$4C0D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4C0E	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$4C0F	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4C10	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4C11	TCSETPAR <b>command: CPU_FREQ invalid or out of range parameter</b>
\$4C12	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$4C13	TCSETPAR <b>command: DEV invalid or out of range parameter</b>
\$4C14	TPCMD <b>command: missing or invalid parameter</b>
\$4C15	TPCMD <b>command: invalid command</b>
\$4C16	TPCMD RUN <b>command: execution error</b>
\$4C17	TPCMD MASSERASE <b>command: missing or invalid parameter</b>

Error #	Description
\$4C18	TPCMD MASSERASE <b>command: invalid parameter</b>
\$4C19	TPCMD MASSERASE <b>command: Flash masserese error</b>
\$4C1A	TPCMD MASSERASE <b>command: D-Flash masserese error</b>
\$4C1B	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$4C1C	TPCMD ERASE <b>command: : target start address parameter or length parameter out of range</b>
\$4C1D	TPCMD ERASE <b>command: Flash erase error</b>
\$4C1E	TPCMD ERASE <b>command: D-Flash erase error</b>
\$4C1F	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$4C20	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$4C21	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$4C22	TPCMD BLANKCHECK <b>command: D-Flash blankcheck error</b>
\$4C23	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$4C24	TPCMD PROGRAM <b>command: invalid parameter</b>
\$4C25	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4C26	TPCMD PROGRAM <b>command: Flash program error</b>
\$4C27	TPCMD PROGRAM <b>command: D-Flash program error</b>
\$4C28	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$4C29	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4C2A	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4C2B	TPCMD VERIFY <b>command: Flash verify error</b>
\$4C2C	TPCMD VERIFY <b>command: D-Flash verify error</b>
\$4C2D	TPCMD READ <b>command: missing or invalid parameter</b>
\$4C2E	TPCMD READ <b>command: Flash read error</b>
\$4C2F	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$4C30	TPCMD WRITE <b>command: write execution error</b>
\$4C31	TPSTART <b>command: execution error</b>
\$4C32	TPSTART <b>command: missing parameter</b>
\$4C33	TPEND <b>command: execution error</b>
\$4C34	TPCMD PROGRAM <b>command: Once memory program error</b>
\$4C40	TPCMD DUMP <b>command: algorithm not supported</b>
\$4C41	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$4C42	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$4C43	TPCMD DUMP <b>command: dump error</b>
\$4C44	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$4C45	TPCMD DUMP <b>command: execution error</b>
\$4C46	TPCMD PARTITION <b>command: missing or invalid parameter</b>
\$4C47	<b>Set partition execution error</b>
<b>Cypress PSoC3 CY_B - Specific Errors</b>	

## Errors

Error #	Description
\$4D00	TCSETDEV <b>command: manufacturer not supported</b>
\$4D01	TCSETDEV <b>command: algorithm not found on card</b>
\$4D02	TCSETDEV <b>command: device not supported</b>
\$4D03	TCSETDEV <b>command: internal hardware configuration error</b>
\$4D04	TCSETDEV <b>command: internal error in algorithm closure</b>
\$4D05	TCSETDEV <b>command: error due to algorithm name mismatch</b>
\$4D06	TCSETPAR <b>command: missing parameter</b>
\$4D07	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$4D08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4D09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4D0A	TCSETPAR JTCLK <b>command: missing parameter</b>
\$4D0B	TCSETPAR JTCLK <b>command: invalid or out of range parameter</b>
\$4D0C	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4D0D	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$4D0E	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4D0F	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4D10	TCSETPAR CONFDATA <b>command: missing or invalid parameter</b>
\$4D11	TPCMD <b>command: missing parameter</b>
\$4D12	TPCMD <b>command: parameter not supported</b>
\$4D13	TPCMD RUN <b>command: execution error</b>
\$4D14	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$4D15	TPCMD MASSERASE <b>command: invalid parameter</b>
\$4D16	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$4D17	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$4D18	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$4D19	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$4D1A	TPCMD BLANKCHECK <b>command: EEPROM blankcheck error</b>
\$4D1B	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$4D1C	TPCMD PROGRAM <b>command: invalid parameter</b>
\$4D1D	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4D1E	TPCMD PROGRAM <b>command: Flash program error</b>
\$4D1F	TPCMD PROGRAM <b>command: Non-Volatile Latch program error</b>
\$4D20	TPCMD PROGRAM <b>command: Write Once Non-Volatile Latch program error</b>
\$4D21	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$4D22	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$4D23	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4D24	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4D25	TPCMD VERIFY <b>command: Flash verify error Algo layer</b>



Error #	Description
\$4D26	TPCMD VERIFY <b>command: Flash verify error API layer</b>
\$4D27	TPCMD VERIFY <b>command: checksum error</b>
\$4D28	TPCMD READ <b>command: missing parameter</b>
\$4D29	TPCMD READ <b>command: invalid parameter</b>
\$4D2A	TPCMD READ <b>command: read execution error</b>
\$4D2B	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$4D2C	TPCMD PROTECT <b>command: protection execution error</b>
\$4D2D	TPCMD PROTECT <b>command: error in opening protection source file</b>
\$4D2E	TPCMD ECC <b>command: ECC check error</b>
\$4D2F	TPCMD ECC <b>command: error opening ECC source file</b>
\$4D30	TPSTART <b>command: execution error</b>
\$4D31	TPSTART <b>command: missing parameter</b>
\$4D32	TPEND <b>command: execution error</b>
\$4D68	TPCMD <b>read function: attempt to read a protected sector</b>
\$4D69	TPCMD <b>write function: attempt to write a protected sector</b>
<b>STMicroelectronics STM32 (SWD interface) CORTEX_C - Specific Errors</b>	
\$4F00	TCSETDEV <b>command: manufacturer not supported</b>
\$4F01	TCSETDEV <b>command: algorithm not found on card</b>
\$4F02	TCSETDEV <b>command: device not supported</b>
\$4F03	TCSETDEV <b>command: internal hardware configuration error</b>
\$4F04	TCSETDEV <b>command: corrupted algorithm file</b>
\$4F05	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$4F06	TCSETPAR <b>command: parameter not supported</b>
\$4F07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$4F08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4F09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4F0A	TCSETPAR RSTDRV <b>command: missing or invalid command option</b>
\$4F0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4F0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$4F0D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4F0E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4F0F	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$4F10	TCSETPAR SWCLK <b>command: missing, invalid or out of range parameter</b>
\$4F11	TPCMD <b>command: missing parameter</b>
\$4F12	TPCMD <b>command: parameter not supported</b>
\$4F13	TPCMD RUN <b>command: execution error</b>
\$4F14	TPCMD MASSERASE <b>command: missing parameter</b>
\$4F15	TPCMD MASSERASE <b>command: invalid parameter</b>

## Errors

Error #	Description
\$4F16	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$4F17	TPCMD MASSERASE <b>command: Option Bytes masserase error</b>
\$4F18	TPCMD ERASE <b>command: missing parameter</b>
\$4F19	TPCMD ERASE <b>command: invalid parameter</b>
\$4F1A	TPCMD ERASE <b>command: Flash erase error</b>
\$4F1B	TPCMD ERASE <b>command: Device Flash secured</b>
\$4F1C	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$4F1D	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$4F1E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$4F1F	TPCMD BLANKCHECK <b>command: Option Bytes blankcheck error</b>
\$4F20	TPCMD PROGRAM <b>command: missing parameter</b>
\$4F21	TPCMD PROGRAM <b>command: invalid parameter</b>
\$4F22	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4F23	TPCMD PROGRAM <b>command: Flash program error</b>
\$4F24	TPCMD PROGRAM <b>command: Option Bytes program error</b>
\$4F25	TPCMD VERIFY <b>command: missing parameter</b>
\$4F26	TPCMD VERIFY <b>command: invalid parameter</b>
\$4F27	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4F28	TPCMD VERIFY <b>command: Flash verify error</b>
\$4F29	TPCMD VERIFY <b>command: Option Bytes verify error</b>
\$4F2A	TPCMD READ <b>command: missing or invalid parameter</b>
\$4F2B	TPCMD READ <b>command: Flash read error</b>
\$4F2C	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$4F2E	TPCMD WRITE <b>command: Flash write error</b>
\$4F2F	TFSTART <b>command: execution error</b>
\$4F30	TFSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$4F31	TFEND <b>command: execution error</b>
\$4F32	TCSETPAR RSTTYPE <b>command: missing or invalid command option</b>
\$4F32	TPCMD MASSERASE <b>command: EEPROM masserase error</b>
\$4F33	TPCMD PROGRAM <b>command: EEPROM program error</b>
<b>Texas Instruments Stellaris TI_ARM -Specific Errors</b>	
\$5000	TCSETDEV <b>command: manufacturer not supported</b>
\$5001	TCSETDEV <b>command: algorithm not found on card</b>
\$5002	TCSETDEV <b>command: device not supported</b>
\$5003	TCSETDEV <b>command: internal hardware configuration error</b>
\$5004	TCSETDEV <b>command: corrupted algorithm file</b>
\$5005	TCSETDEV <b>command: programming algorithm requires FlashRunner firmware version 01.01.00.00 or newer</b>
\$5006	TCSETPAR <b>command: parameter not supported</b>

Error #	Description
\$5007	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5008	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5009	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$500A	TCSETPAR RSTDRV <b>command: missing or invalid command option</b>
\$500B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$500C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$500D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$500E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$5017	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$500F	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$5010	TCSETPAR SWCLK <b>command: missing, invalid or out of range parameter</b>
\$5011	TPCMD <b>command: missing parameter</b>
\$5012	TPCMD <b>command: parameter not supported</b>
\$5013	TPCMD RUN <b>command: execution error</b>
\$5014	TPCMD MASSERASE <b>command: missing parameter</b>
\$5015	TPCMD MASSERASE <b>command: invalid parameter</b>
\$5016	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$5018	TPCMD ERASE <b>command: missing parameter</b>
\$5019	TPCMD ERASE <b>command: invalid parameter</b>
\$501A	TPCMD ERASE <b>command: Flash erase error</b>
\$501C	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$501D	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$501E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$5020	TPCMD PROGRAM <b>command: missing parameter</b>
\$5021	TPCMD PROGRAM <b>command: invalid parameter</b>
\$5022	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5023	TPCMD PROGRAM <b>command: Flash program error</b>
\$5025	TPCMD VERIFY <b>command: missing parameter</b>
\$5026	TPCMD VERIFY <b>command: invalid parameter</b>
\$5027	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5028	TPCMD VERIFY <b>command: Flash verify error</b>
\$502A	TPCMD READ <b>command: missing or invalid parameter</b>
\$502B	TPCMD READ <b>command: Flash read error</b>
\$502F	TPSTART <b>command: execution error</b>
\$5030	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$5031	TPEND <b>command: execution error</b>
\$5032	TPCMD MASSERASE <b>command: Eeprom masserase error</b>
\$5033	TPCMD PROGRAM <b>command: Eeprom masserase error</b>

## Errors

Error #	Description
<b>Texas Instruments TMS470 - Specific Errors</b>	
\$5200	TCSETDEV <b>command: manufacturer not supported</b>
\$5201	TCSETDEV <b>command: algorithm not found on card</b>
\$5202	TCSETDEV <b>command: device not supported</b>
\$5203	TCSETDEV <b>command: internal hardware configuration error</b>
\$5204	TCSETDEV <b>command: internal kernel error</b>
\$5205	TCSETDEV <b>command: old internal fw version</b>
\$5206	TCSETPAR <b>command: not standard parameter</b>
\$5207	TCSETPAR <b>command: wrong clock out parameter</b>
\$5208	TCSETPAR <b>command: wrong reset down parameter</b>
\$5209	TCSETPAR <b>command: wrong reset up parameter</b>
\$520A	TCSETPAR <b>command: wrong reset drive parameter</b>
\$5232	TCSETPAR <b>command: wrong reset type parameter</b>
\$520B	TCSETPAR <b>command: wrong power down parameter</b>
\$520C	TCSETPAR <b>command: wrong power up parameter</b>
\$520D	TCSETPAR <b>command: wrong VDD parameter</b>
\$520E	TCSETPAR <b>command: wrong VDD Aux parameter</b>
\$520F	TCSETPAR <b>command: wrong CPU Frequency parameter</b>
\$5210	TCSETPAR <b>command: wrong JTAG Clock parameter</b>
\$5211	TPCMD <b>command: invalid/missing command</b>
\$5212	TPCMD <b>command: command not recognized</b>
\$5213	TPCMD <b>command: low level hw error</b>
\$5214	TPCMD MASSERASE <b>command: missing command</b>
\$5215	TPCMD MASSERASE <b>command: invalid command/parameter</b>
\$5216	TPCMD MASSERASE <b>command: masserase execution command</b>
\$521C	TPCMD BLANKCHECK <b>command: blankcheck syntax error</b>
\$521D	TPCMD BLANKCHECK <b>command: blankcheck memory parameter error</b>
\$521E	TPCMD BLANKCHECK <b>command: blankcheck execution error</b>
\$5220	TPCMD PROGRAM <b>command: program syntax error</b>
\$5221	TPCMD PROGRAM <b>command: memory parameter error</b>
\$5222	TPCMD PROGRAM <b>command: source file error</b>
\$5223	TPCMD PROGRAM <b>command: memory size error</b>
\$5225	TPCMD VERIFY <b>command: verify syntax error</b>
\$5226	TPCMD VERIFY <b>command: verify memory parameter error</b>
\$5227	TPCMD VERIFY <b>command: source file error</b>
\$5228	TPCMD VERIFY <b>command: verify memory size error</b>
\$522A	TPCMD READ <b>command: read syntax error</b>
\$522B	TPCMD READ <b>command: read execution error</b>
\$522F	TPCMD TPSTART <b>command: unable to execute start</b>
\$5230	TPCMD TPSTART <b>command: execution error</b>

Error #	Description
\$5231	TPCMD TPEND <b>command: execution error</b>
<b>Microchip PIC32 - Specific Errors</b>	
\$5500	TCSETDEV <b>command: manufacturer not supported</b>
\$5501	TCSETDEV <b>command: algorithm not found on card</b>
\$5502	TCSETDEV <b>command: wrong device code</b>
\$5503	TCSETDEV <b>command: wrong kernel</b>
\$5504	TCSETDEV <b>command: algorithm not supported</b>
\$5505	TCSETDEV <b>command: load FPGA</b>
\$5506	TCSETPAR <b>command: invalid command</b>
\$5507	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$5508	TCSETPAR VDD <b>command: missing parameter</b>
\$5509	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$550A	TCSETPAR PWUP <b>command: missing parameter</b>
\$550B	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$550C	TCSETPAR PWDOWN <b>command: missing parameter</b>
\$550D	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$550E	TCSETPAR RSTUP <b>command: missing parameter</b>
\$550F	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$5510	TCSETPAR RSTDOWN <b>command: missing parameter</b>
\$5511	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$5512	TCSETPAR JTLCK <b>command: missing, invalid or out of range parameter</b>
\$5513	TPSTART <b>command: command not preceded by TCSETPAR JTCLK command</b>
\$5514	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5515	TCSETPAR CMODE <b>command: missing or invalid parameter</b>
\$5517	TPCMD <b>command: invalid command</b>
\$5518	TPCMD SETJTCLK <b>command: invalid or out of range parameter</b>
\$5519	TPCMD SETJTCLK <b>command: missing parameter</b>
\$551A	TPCMD SETJTCLK <b>command: wrong parameter</b>
\$551B	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$551C	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$551D	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$551E	TPCMD VERIFY <b>command: Flash verify error</b>
\$551F	TPCMD VERIFY <b>command: Eeprom verify error</b>
\$5520	TPCMD VERIFY <b>command: IDCODE verify error</b>
\$5521	TPCMD READ <b>command: missing or invalid parameter</b>
\$5522	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$5523	TPCMD READ <b>command: Flash read error</b>
\$5526	TPCMD PROGRAM <b>command: missing or invalid parameter</b>

## Errors

Error #	Description
\$5527	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5528	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5529	TPCMD PROGRAM <b>command: Flash program error</b>
\$552B	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$552C	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$552D	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$552E	TPCMD BLANKCHECK <b>command: Eeprom blankcheck error</b>
\$552F	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$5530	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$5531	TPCMD MASSERASE <b>command: Eeprom masserase error</b>
\$5532	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$5533	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$5534	TPCMD ERASE <b>command: Flash erase error</b>
\$5535	TPCMD ERASE <b>command: Eeprom erase error</b>
\$5536	TPSTART <b>command: execution error</b>
\$5537	TPEND <b>command: execution error</b>
\$5538	TPCMD RUN <b>command: execution error</b>
<b>Atmel ATSAM - Specific Errors</b>	
\$5800	TCSETDEV <b>command: manufacturer not supported</b>
\$5801	TCSETDEV <b>command: algorithm not found on card</b>
\$5802	TCSETDEV <b>command: wrong device code</b>
\$5803	TCSETDEV <b>command: load FPGA</b>
\$5804	TCSETDEV <b>command: wrong device data</b>
\$5805	TCSETDEV <b>command: algorithm not supported</b>
\$5806	TCSETPAR <b>command: invalid command</b>
\$5807	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$5808	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$5809	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$580A	TCSETPAR RSTDRV <b>command: invalid or out of range parameter</b>
\$580B	TCSETPAR RSTTYPE <b>command: invalid or out of range parameter</b>
\$580C	TCSETPAR CMODE <b>command: invalid or out of range parameter</b>
\$580D	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$580E	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$580F	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$5810	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$5811	TCSETPAR CPUFREQ <b>command: invalid or out of range parameter</b>
\$5812	TCSETPAR JTCLK <b>command: missing or invalid parameter</b>
\$5813	TCSETPAR DEV <b>command: invalid or out of range parameter</b>

Error #	Description
\$5814	TPCMD <b>command: invalid command</b>
\$5815	TPCMD <b>command: invalid command</b>
\$5816	TPCMD RUN <b>command: invalid or out of range parameter</b>
\$5817	TPCMD MASSERASE <b>command: missing parameter</b>
\$5818	TPCMD MASSERASE <b>command: invalid parameter</b>
\$5819	TPCMD MASSERASE <b>command: Flash masserese error</b>
\$581A	TPCMD MASSERASE <b>command: Data Flash masserese error</b>
\$581B	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$581C	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$581D	TPCMD ERASE <b>command: Flash erase error</b>
\$581E	TPCMD ERASE <b>command: Flash device secured</b>
\$581F	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$5820	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5821	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$5822	TPCMD BLANKCHECK <b>command: Data Flash blankcheck error</b>
\$5823	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5824	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5825	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5826	TPCMD PROGRAM <b>command: Flash program error</b>
\$5827	TPCMD PROGRAM <b>command: Data Flash program error</b>
\$5828	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5829	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$582A	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$582B	TPCMD VERIFY <b>command: Flash verify error</b>
\$582C	TPCMD VERIFY <b>command: Data Flash verify error</b>
\$582D	TPCMD READ <b>command: missing or invalid parameter</b>
\$582E	TPCMD READ <b>command: read error</b>
\$582F	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$5830	TPCMD WRITE <b>command: write error</b>
\$5831	TPSTART <b>command: execution error</b>
\$5832	TPSTART <b>command: missing or invalid parameter</b>
\$5833	TPEND <b>command: execution error</b>
\$5839	TPCMD CLEARGNVM <b>command: missing or invalid parameter</b>
\$583A	TPCMD SETGNVM <b>command: missing or invalid parameter</b>
\$583B	TPCMD CLEARGNVM <b>command: Clear GPNVM error</b>
\$583C	TPCMD SETGNVM <b>command: Set GPNVM error</b>
\$583D	TPCMD LOCKREGION <b>command: missing or invalid parameter</b>
\$583E	TPCMD LOCKREGION <b>command: Lock Region error</b>

## Errors

Error #	Description
\$583F	TPCMD LOCKREGION <b>command: target start address parameter or length parameter out of range</b>
\$5840	TPCMD UNLOCKREGION <b>command: missing or invalid parameter</b>
\$5841	TPCMD UNLOCKREGION <b>command: Lock Region error</b>
\$5842	TPCMD UNLOCKREGION <b>command: target start address parameter or length parameter out of range</b>
\$5843	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$5844	TPCMD PROTECT <b>command: Protect error</b>
\$5845	TPCMD PROTECT <b>command: target start address parameter or length parameter out of range</b>
\$5846	TPCMD UNPROTECT <b>command: missing or invalid parameter</b>
\$5847	TPCMD UNPROTECT <b>command: Unprotect error</b>
\$5848	TPCMD UNPROTECT <b>command: target start address parameter or length parameter out of range</b>
\$5849	TPCMD CHECKPROTECT <b>command: missing or invalid parameter</b>
\$584A	TPCMD CHECKPROTECT <b>command: Check protect error</b>
\$584B	TPCMD CHECKPROTECT <b>command: target start address parameter or length parameter out of range</b>
<b>FREESCALE MX6/S6J CORTEX_A - Specific Errors</b>	
\$6100	TCSETDEV <b>command: manufacturer not supported</b>
\$6101	TCSETDEV <b>command: algorithm not found on card</b>
\$6102	TCSETDEV <b>command: wrong device code</b>
\$6103	TCSETDEV <b>command: load FPGA</b>
\$6104	TCSETDEV <b>command: wrong device data</b>
\$6105	TCSETDEV <b>command: algorithm not supported</b>
\$6106	TCSETPAR <b>command: invalid command</b>
\$6107	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$6108	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$6109	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$610A	TCSETPAR RSTDRV <b>command: invalid or out of range parameter</b>
\$610B	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$610C	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$610D	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$610E	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$610F	TCSETPAR CPUFREQ <b>command: invalid or out of range parameter</b>
\$6110	TCSETPAR JTCLK <b>command: missing or invalid parameter</b>
\$6111	TPCMD <b>command: invalid command</b>
\$6112	TPCMD <b>command: invalid command</b>
\$6113	TPCMD RUN <b>command: invalid or out of range parameter</b>
\$6114	TPCMD MASSERASE <b>command: missing parameter</b>
\$6115	TPCMD MASSERASE <b>command: invalid parameter</b>
\$6116	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$6118	TPCMD ERASE <b>command: missing or invalid parameter</b>



Error #	Description
\$6119	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$611A	TPCMD ERASE <b>command: Flash erase error</b>
\$611B	TPCMD ERASE <b>command: Flash device secured</b>
\$611C	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$611D	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$611E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$6120	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$6121	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6122	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$6123	TPCMD PROGRAM <b>command: Flash program error</b>
\$6124	TPCMD PROGRAM <b>command: OTP Fuses program error</b>
\$6125	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6126	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6127	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$6128	TPCMD VERIFY <b>command: Flash verify error</b>
\$6129	TPCMD READ <b>command: read OTP fuses error</b>
\$612A	TPCMD READ <b>command: missing or invalid parameter</b>
\$612B	TPCMD READ <b>command: read error</b>
\$612C	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$612D	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$612E	TPCMD WRITE <b>command: write error</b>
\$6132	TPCMD WRITE <b>command: target start address parameter or length parameter out of range</b>
\$612F	TPSTART <b>command: execution error</b>
\$6130	TPSTART <b>command: missing or invalid parameter</b>
\$6131	TPCMD RUN <b>command: execution error</b>
<b>STMICROELECTRONICS STM32 SWD CORTEX_D - Specific Errors</b>	
\$6200	TCSETDEV <b>command: manufacturer not supported</b>
\$6201	TCSETDEV <b>command: algorithm not found on card</b>
\$6202	TCSETDEV <b>command: wrong device code</b>
\$6203	TCSETDEV <b>command: load FPGA</b>
\$6204	TCSETDEV <b>command: wrong device data</b>
\$6205	TCSETDEV <b>command: algorithm not supported</b>
\$6206	TCSETPAR <b>command: invalid command</b>
\$6207	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$6208	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$6209	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>

## Errors

Error #	Description
§620A	TCSETPAR RSTDRV <b>command: invalid or out of range parameter</b>
§620B	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
§620C	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
§620D	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
§620E	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
§620F	TCSETPAR CPUFREQ <b>command: invalid or out of range parameter</b>
§6210	TCSETPAR SWCLK <b>command: missing or invalid parameter</b>
§6211	TPCMD <b>command: invalid command</b>
§6212	TPCMD <b>command: invalid command</b>
§6213	TPCMD RUN <b>command: invalid or out of range parameter</b>
§6214	TPCMD MASSERASE <b>command: missing parameter</b>
§6215	TPCMD MASSERASE <b>command: invalid parameter</b>
§6216	TPCMD MASSERASE <b>command: Flash masserase error</b>
§6216	TPCMD MASSERASE <b>command: Option Bytes masserase error</b>
§6218	TPCMD ERASE <b>command: missing or invalid parameter</b>
§6219	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
§621A	TPCMD ERASE <b>command: Flash erase error</b>
§621B	TPCMD ERASE <b>command: Flash device secured</b>
§621C	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
§621D	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
§621E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
§621E	TPCMD BLANKCHECK <b>command: Option Bytes blankcheck error</b>
§6220	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
§6221	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
§6222	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
§6223	TPCMD PROGRAM <b>command: Flash program error</b>
§6224	TPCMD PROGRAM <b>command: Option bytes program error</b>
§6225	TPCMD VERIFY <b>command: missing or invalid parameter</b>
§6226	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
§6227	TPCMD VERIFY <b>command: source offset parameter out of range</b>
§6228	TPCMD VERIFY <b>command: Flash verify error</b>
§6228	TPCMD VERIFY <b>command: Option bytes verify error</b>
§622A	TPCMD READ <b>command: missing or invalid parameter</b>
§622B	TPCMD READ <b>command: read error</b>
§622D	TPCMD WRITE <b>command: missing or invalid parameter</b>
§622E	TPCMD WRITE <b>command: write error</b>
§622F	TPSTART <b>command: execution error</b>
§6230	TPSTART <b>command: missing or invalid parameter</b>

Error #	Description
\$6231	TPEND <b>command: execution error</b>
\$6232	TCSETPAR RSTTYPE <b>command: invalid or out of range parameter</b>
\$6232	TPCMD RESET <b>command: Reset Option byte error</b>
\$6232	TPCMD MASSERASE <b>command: EEPROM masserase error</b>
\$6233	TPCMD PROGRAM <b>command: EEPROM program error</b>
\$6234	TPCMD MASSERASE <b>command: masserase Option bytes not supported</b>
\$6235	TPCMD MASSERASE <b>command: Chiperase error</b>
\$6236	TPCMD PROGRAM <b>command: Lock bits program error</b>
<b>NORDIC NRF52 SWD CORTEX_E - Specific Errors</b>	
\$4F00	TCSETDEV <b>command: manufacturer not supported</b>
\$4F01	TCSETDEV <b>command: algorithm not found on card</b>
\$4F02	TCSETDEV <b>command: wrong device code</b>
\$4F03	TCSETDEV <b>command: internal hardware configuration error or load FPGA</b>
\$4F04	TCSETDEV <b>command: wrong device data</b>
\$4F05	TCSETDEV <b>command: algorithm not supported</b>
\$4F06	TCSETPAR <b>command: parameter not supported</b>
\$4F07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$4F08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4F09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4F0A	TCSETPAR RSTDRV <b>command: missing or invalid command option</b>
\$4F0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4F0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$4F0D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4F0E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4F0F	TCSETPAR CUFREQ <b>command: missing, invalid or out of range parameter</b>
\$4F10	TCSETPAR SWCLK <b>command: missing, invalid or out of range parameter</b>
\$4F11	TPCMD <b>command: missing parameter</b>
\$4F12	TPCMD <b>command: parameter not supported</b>
\$4F13	TPCMD RUN <b>command: execution error</b>
\$4F14	TPCMD MASSERASE <b>command: missing parameter</b>
\$4F15	TPCMD MASSERASE <b>command: invalid parameter</b>
\$4F16	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$4F17	TPCMD MASSERASE <b>command: Option Bytes masserase error</b>
\$4F18	TPCMD ERASE <b>command: missing parameter</b>
\$4F19	TPCMD ERASE <b>command: invalid parameter</b>
\$4F1A	TPCMD ERASE <b>command: Flash erase error</b>
\$4F1B	TPCMD ERASE <b>command: Device flash secured</b>
\$4F1C	TPCMD BLANKCHECK <b>command: missing parameter</b>

## Errors

Error #	Description
\$4F1D	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$4F1E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$4F1F	TPCMD BLANKCHECK <b>command: Option Bytes blankcheck error</b>
\$4F20	TPCMD PROGRAM <b>command: missing parameter</b>
\$4F21	TPCMD PROGRAM <b>command: invalid parameter</b>
\$4F22	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4F23	TPCMD PROGRAM <b>command: Flash program error</b>
\$4F24	TPCMD PROGRAM <b>command: Option Bytes program error</b>
\$4F25	TPCMD VERIFY <b>command: missing parameter</b>
\$4F26	TPCMD VERIFY <b>command: invalid parameter</b>
\$4F27	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$4F28	TPCMD VERIFY <b>command: Flash verify error</b>
\$4F29	TPCMD VERIFY <b>command: Option Bytes verify error</b>
\$4F2A	TPCMD READ <b>command: missing or invalid parameter</b>
\$4F2B	TPCMD READ <b>command: Flash read error</b>
\$4F2C	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$4F2E	TPCMD WRITE <b>command: Flash write error</b>
\$4F0F	TPSTART <b>command: execution error</b>
\$4F30	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$4F31	TPEND <b>command: execution error</b>
\$4F32	TCSETPAR RSTTYPE <b>command: missing or invalid command option</b>
\$4F32	TPCMD MASSErase <b>command: EEPROM masserase error</b>
\$4F33	TPCMD PROGRAM <b>command: EEPROM program error</b>
<b>STMICROELECTRONICS STM32 SWD CORTEX_F - Specific Errors</b>	
\$6F00	TCSETDEV <b>command: manufacturer not supported</b>
\$6F01	TCSETDEV <b>command: algorithm not found on card</b>
\$6F02	TCSETDEV <b>command: wrong device code</b>
\$6F03	TCSETDEV <b>command: load FPGA</b>
\$6F04	TCSETDEV <b>command: wrong device data</b>
\$6F05	TCSETDEV <b>command: algorithm not supported</b>
\$6F06	TCSETPAR <b>command: invalid command</b>
\$6F07	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$6F08	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$6F09	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$6F0A	TCSETPAR RSTDRV <b>command: invalid or out of range parameter</b>
\$6F0B	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$6F0C	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$6F0D	TCSETPAR VDD <b>command: invalid or out of range parameter</b>

Error #	Description
\$6F0E	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$6F0F	TCSETPAR CPUFREQ <b>command: invalid or out of range parameter</b>
\$6F10	TCSETPAR SWCLK <b>command: missing or invalid parameter</b>
\$6F11	TPCMD <b>command: invalid command</b>
\$6F12	TPCMD <b>command: invalid command</b>
\$6F13	TPCMD RUN <b>command: invalid or out of range parameter</b>
\$6F14	TPCMD MASSERASE <b>command: missing parameter</b>
\$6F15	TPCMD MASSERASE <b>command: invalid parameter</b>
\$6F16	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$6F16	TPCMD MASSERASE <b>command: Option Bytes masserase error</b>
\$6F18	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$6F19	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$6F1A	TPCMD ERASE <b>command: Flash erase error</b>
\$6F1B	TPCMD ERASE <b>command: Flash device secured</b>
\$6F1C	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$6F1D	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$6F1E	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$6F1E	TPCMD BLANKCHECK <b>command: Option Bytes blankcheck error</b>
\$6F20	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$6F21	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6F22	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$6F23	TPCMD PROGRAM <b>command: Flash program error</b>
\$6F24	TPCMD PROGRAM <b>command: Option bytes program error</b>
\$6F25	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6F26	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6F27	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$6F28	TPCMD VERIFY <b>command: Flash verify error</b>
\$6F28	TPCMD VERIFY <b>command: Option bytes verify error</b>
\$6F2A	TPCMD READ <b>command: missing or invalid parameter</b>
\$6F2B	TPCMD READ <b>command: read error</b>
\$6F2D	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$6F2E	TPCMD WRITE <b>command: write error</b>
\$6F2F	TPSTART <b>command: execution error</b>
\$6F30	TPSTART <b>command: missing or invalid parameter</b>
\$6F31	TPEND <b>command: execution error</b>
\$6F32	TCSETPAR RSTTYPE <b>command: missing or invalid command option</b>
\$6F32	TPCMD MASSERASE <b>command: EEPROM masserase error</b>
\$6F32	TCSETPAR RSTTYPE <b>command: invalid or out of range parameter</b>

## Errors

---

Error #	Description
\$6F33	TPCMD PROGRAM <b>command: EEPROM program error</b>
<b>ALTERA EPM240 CPLD_A and LATTICE LA-XP2-5E/LCMXO CPLD_A - Specific Errors</b>	
\$2700	TCSETDEV <b>command: manufacturer not supported</b>
\$2701	TCSETDEV <b>command: algorithm not found on card</b>
\$2702	TCSETDEV <b>command: wrong device code</b>
\$2703	TCSETDEV <b>command: wrong device data</b>
\$2704	TCSETDEV <b>command: load FPGA</b>
\$2706	TCSETPAR <b>command: invalid command</b>
\$2707	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$2708	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$2709	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$270A	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$270B	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$270C	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$270D	TCSETPAR VDD_AUX <b>command: invalid or out of range parameter</b>
\$270E	TCSETPAR JTCLK <b>command: missing or invalid parameter</b>
\$270F	TCSETPAR <b>command: invalid command</b>
\$2710	TPCMD <b>command: invalid command</b>
\$2711	TPCMD <b>command: invalid command</b>
\$2712	TPCMD RUN <b>command: invalid or out of range parameter</b>
\$2718	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2719	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$271A	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$271B	TPCMD PROGRAM <b>command: Flash program error</b>
\$2723	TFSTART <b>command: execution error</b>
\$2724	TFSTART <b>command: missing or invalid parameter</b>
\$2725	TPEND <b>command: execution error</b>

Error #	Description
<b>CAST R8051XC2 CST_A- Specific Errors</b>	
\$5B00	TCSETDEV <b>command: manufacturer not supported</b>
\$5B01	TCSETDEV <b>command: algorithm not found on card</b>
\$5B02	TCSETDEV <b>command: wrong device code</b>
\$5B03	TCSETDEV <b>command: load FPGA</b>
\$5B04	TCSETPAR <b>command: invalid command</b>
\$5B05	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$5B06	TCSETPAR VDD <b>command: missing parameter</b>
\$5B07	TCSETPAR PWUP <b>command: invalid or out of range parameter</b>
\$5B08	TCSETPAR PWUP <b>command: missing parameter</b>
\$5B09	TCSETPAR PWDOWN <b>command: invalid or out of range parameter</b>
\$5B0A	TCSETPAR PWDOWN <b>command: missing parameter</b>
\$5B0B	TCSETPAR RSTUP <b>command: invalid or out of range parameter</b>
\$5B0C	TCSETPAR RSTUP <b>command: missing parameter</b>
\$5B0D	TCSETPAR RSTDOWN <b>command: invalid or out of range parameter</b>
\$5B0E	TCSETPAR RSTDOWN <b>command: missing parameter</b>
\$5B0F	TCSETPAR INIT_JTCLK <b>command: missing, invalid or out of range parameter</b>
\$5B10	TCSETPAR INIT_JTCLK <b>command: missing or invalid parameter</b>
\$5B11	TCSETPAR PROG_JTCLK <b>command: invalid command</b>
\$5B12	TCSETPAR PROG_JTCLK <b>command: invalid or out of range parameter</b>
\$5B13	TPCMD <b>command: invalid command</b>
\$5B14	TPCMD TCSETPAR_FOSC <b>command: missing parameter</b>
\$5B15	TPCMD TCSETPAR_FOSC <b>command: wrong parameter</b>
\$5B20	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$5B21	TPCMD MASSERASE <b>command: Flash masserase error</b>
\$5B26	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5B27	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5B28	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5B29	TPCMD PROGRAM <b>command: Flash program error</b>
\$5B2A	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5B2B	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5B2C	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5B2D	TPCMD VERIFY <b>command: Flash verify error</b>
\$5B2E	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$5B2F	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5B30	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$5B31	TPCMD READ <b>command: missing or invalid parameter</b>
\$5B32	TPCMD READ <b>command: command: target start address parameter or length parameter out of range</b>

## Errors

Error #	Description
\$5B33	TPCMD READ <b>command: Flash read error</b>
\$5B34	TPCMD RUN <b>command: execution error</b>
\$5B35	TPSTART <b>command: execution error</b>
\$5B36	TPEND <b>command: execution error</b>
\$5B37	TPCMD ENTRY <b>command: execution error</b>
<b>CYPRESS CY8C4xx/CYBL1xx CY_C - Specific Errors</b>	
\$5A00	TCSETDEV <b>command: manufacturer not supported</b>
\$5A01	TCSETDEV <b>command: algorithm not found on card</b>
\$5A02	TCSETDEV <b>command: wrong device code</b>
\$5A03	TCSETDEV <b>command: load FPGA</b>
\$5A04	TCSETPAR <b>command: invalid command</b>
\$5A05	TCSETPAR FCPU <b>command: invalid or out of range parameter</b>
\$5A06	TCSETPAR FCPU <b>command: missing parameter</b>
\$5A07	TCSETPAR VDD <b>command: invalid or out of range parameter</b>
\$5A08	TCSETPAR VDD <b>command: missing parameter</b>
\$5A09	TCSETPAR SWDCLK <b>command: invalid or out of range parameter</b>
\$5A0A	TCSETPAR SWDCLK <b>command: missing parameter</b>
\$5A0B	TCSETPAR ENTRYMODE <b>command: invalid or out of range parameter</b>
\$5A0C	TCSETPAR ENTRYMODE <b>command: missing parameter</b>
\$5A0D	TCSETPAR CLKOUT <b>command: invalid or out of range parameter</b>
\$5A14	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$5A0E	TPCMD <b>command: missing parameter</b>
\$5A0F	TPCMD <b>command: invalid command</b>
\$5A10	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5A11	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5A12	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5A13	TPCMD VERIFY <b>command: Flash verify error</b>
\$5A16	TPCMD VERIFY <b>command: Lock verify error</b>
\$5A17	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5A18	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5A19	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5A1A	TPCMD PROGRAM <b>command: Flash program error</b>
\$5A1D	TPCMD PROGRAM <b>command: Lock program error</b>
\$5A1E	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$5A1F	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5A20	TPCMD BLANKCHECK <b>command: Flash blankcheck error</b>
\$5A22	TPCMD MASSErase <b>command: missing or invalid parameter</b>
\$5A25	TPCMD RUN <b>command: execution error</b>



Error #	Description
\$5A26	TPEND <b>command: execution error</b>
\$5A27	TPSTART <b>command: execution error</b>
\$5A28	TPCMD ENTRY <b>command: execution error</b>
\$5A30	TPCMD READ <b>command: missing or invalid parameter</b>
\$5A31	TPCMD READ <b>command: command: target start address parameter or length parameter out of range</b>
\$5A32	TPCMD READ <b>command: Flash read error</b>
\$5A33	TCSETPAR SETIDCODE <b>command: missing or invalid parameter</b>
\$5A34	TCSETPAR SETIDCODE <b>command: IDCODE constant value</b>
\$5A35	TCSETPAR SETIDCODE <b>command: source offset parameter out of range</b>
\$5A36	TCSETPAR SETIDCODE <b>command: IDCODE already set</b>
<b>ELMOS E523 EL_B - Specific Errors</b>	
\$6800	TCSETDEV <b>command: manufacturer not supported</b>
\$6801	TCSETDEV <b>command: algorithm not found on card</b>
\$6802	TCSETDEV <b>command: invalid device code</b>
\$6803	TCSETDEV <b>command: internal hardware configuration error or FPGA load</b>
\$6804	TCSETDEV <b>command: invalid device data</b>
\$6805	TCSETDEV <b>command: algorithm not supported</b>
\$6806	TCSETPAR <b>command: parameter not supported</b>
\$6807	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$6808	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$6809	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$680A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$680B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$680C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$680D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$680E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$680F	TCSETPAR CUFREQ <b>command: missing, invalid or out of range parameter</b>
\$6810	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$6811	TPCMD <b>command: missing parameter</b>
\$6812	TPCMD <b>command: parameter not supported</b>
\$6813	TPCMD RUN <b>command: execution error</b>
\$6814	TPCMD MASSERASE <b>command: missing parameter</b>
\$6815	TPCMD MASSERASE <b>command: invalid parameter</b>
\$6816	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$6818	TPCMD ERASE <b>command: missing parameter</b>
\$6819	TPCMD ERASE <b>command: invalid parameter</b>
\$681A	TPCMD ERASE <b>command: Flash erase error</b>
\$681B	TPCMD BLANKCHECK <b>command: missing parameter</b>

## Errors

Error #	Description
\$681C	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$681D	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$681E	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$681F	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6820	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$6821	TPCMD PROGRAM <b>command: Flash program error</b>
\$6822	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6823	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6824	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$6825	TPCMD VERIFY <b>command: Flash verify error</b>
\$6826	TPCMD READ <b>command: missing or invalid parameter</b>
\$6827	TPCMD READ <b>command: read error</b>
\$6828	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$6829	TPCMD WRITE <b>command: target start address parameter or length parameter out of range (INT_REG)</b>
\$682A	TPCMD WRITE <b>command: INT_REG write error</b>
\$682B	TPCMD READ <b>command: target start address parameter or length parameter out of range (INT_REG)</b>
\$682C	TPCMD READ <b>command: INT_REG read error</b>
\$682F	TPSTART <b>command: execution error</b>
\$6830	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$6831	TPEND <b>command: execution error</b>
<b>STMICROELECTRONICS SPC56 FSL_B- Specific Errors</b>	
\$2B00	TCSETDEV <b>command: manufacturer not supported</b>
\$2B01	TCSETDEV <b>command: algorithm not found on card</b>
\$2B02	TCSETDEV <b>command: device not supported</b>
\$2B03	TCSETDEV <b>command: internal hardware configuration error</b>
\$2B04	TCSETDEV <b>command: corrupted algorithm file</b>
\$2B05	<b>This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$2B06	TCSETPAR <b>command: parameter not supported</b>
\$2B07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2B08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2B09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2B0A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$2B0B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$2B0C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$2B0D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$2B0E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$2B0F	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$2B10	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$2B11	TCSETPAR PLLFREQ <b>command: FOSC parameter has not been set</b>
\$2B12	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$2B13	TPCMD <b>command: missing parameter</b>
\$2B14	TPCMD <b>command: parameter not supported</b>
\$2B15	TPCMD RUN <b>command: execution error</b>
\$2B16	TPCMD MASSERASE <b>command: missing parameter</b>
\$2B17	TPCMD MASSERASE <b>command: invalid parameter</b>
\$2B18	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$2B19	TPCMD MASSERASE <b>command: Shadow memory mass erase error</b>
\$2B1A	TPCMD MASSERASE <b>command: device is protected</b>
\$2B1B	TPCMD ERASE <b>command: missing parameter</b>
\$2B1C	TPCMD ERASE <b>command: invalid parameter</b>
\$2B1D	TPCMD ERASE <b>command: Flash memory erase error</b>
\$2B1E	TPCMD ERASE <b>command: Shadow memory erase error</b>
\$2B1F	TPCMD ERASE <b>command: device is protected</b>
\$2B20	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$2B21	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$2B22	TPCMD BLANKCHECK <b>command: Flash memory blank check error</b>
\$2B23	TPCMD BLANKCHECK <b>command: Shadow memory blank check error</b>
\$2B24	TPCMD BLANKCHECK <b>command: device is protected</b>
\$2B25	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2B26	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2B27	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2B28	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$2B29	TPCMD PROGRAM <b>command: Shadow memory program error</b>
\$2B2A	TPCMD PROGRAM <b>command: device is protected</b>
\$2B2B	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2B2C	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2B2D	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2B2E	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$2B2F	TPCMD VERIFY <b>command: Shadow memory verify error</b>
\$2B30	TPCMD VERIFY <b>command: device is protected</b>
\$2B31	TPSTART <b>command: execution error</b>
\$2B32	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$2B33	TPEND <b>command: execution error</b>

## Errors

Error #	Description
\$2B34	TPCMD MASSErase <b>command: Data Flash memory mass erase error</b>
\$2B35	TPCMD ERASE <b>command: Data Flash memory erase error</b>
\$2B36	TPCMD BLANKCHECK <b>command: Data Flash memory blankcheck error</b>
\$2B37	TPCMD PROGRAM <b>command: Data Flash memory program error</b>
\$2B38	TPCMD VERIFY <b>command: Data Flash memory verify error</b>
\$2B39	TPCMD ERASE <b>command: invalid parameter</b>
\$2B3A	TPCMD BLANKCHECK <b>command: Test Flash memory blankcheck error</b>
\$2B3B	TPCMD BLANKCHECK <b>command: Data Test Flash memory blankcheck error</b>
\$2B3C	TPCMD PROGRAM <b>command: Test Flash memory program error</b>
\$2B3D	TPCMD PROGRAM <b>command: Data Test Flash memory program error</b>
\$2B3E	TPCMD VERIFY <b>command: Test Flash memory verify error</b>
\$2B3F	TPCMD VERIFY <b>command: Data Test Flash memory verify error</b>
\$2B40	TPCMD READ <b>command: missing or invalid parameter</b>
\$2B41	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$2B42	TPCMD READ <b>command: Flash read error</b>
\$2B43	TPCMD READ <b>command: Shadow Flash read error</b>
\$2B44	TPCMD READ <b>command: Device flash secured</b>
\$2B45	TPCMD DUMP <b>command: algorithm not supported</b>
\$2B46	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$2B47	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$2B48	TPCMD DUMP <b>command: dump error</b>
\$2B49	TPCMD JTAGID <b>command: invalid JTAGID</b>
\$2B4A	TCSETPAR RSTEXIT <b>command: missing or invalid parameter</b>
<b>FREESCALE MPC57/S32R FSL_E- Specific Errors</b>	
\$5600	TCSETDEV <b>command: manufacturer not supported</b>
\$5601	TCSETDEV <b>command: algorithm not found on card</b>
\$5602	TCSETDEV <b>command: device not supported</b>
\$5603	TCSETDEV <b>command: internal hardware configuration error</b>
\$5604	TCSETDEV <b>command: invalid device data</b>
\$5605	TCSETDEV <b>command: This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$5606	TCSETPAR <b>command: parameter not supported</b>
\$5607	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5608	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5609	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$560A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$560B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$560C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$560D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>

Error #	Description
\$560E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$560F	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$5610	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$5611	TCSETPAR PLLFREQ <b>command: FOSC parameter has not been set</b>
\$5612	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$5613	TPCMD <b>command: missing parameter</b>
\$5614	TPCMD <b>command: parameter not supported</b>
\$5615	TPCMD RUN <b>command: execution error</b>
\$5616	TPCMD MASSERASE <b>command: missing parameter</b>
\$5617	TPCMD MASSERASE <b>command: invalid parameter</b>
\$5618	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$5619	TPCMD VERIFY <b>command: BAF flash verify error</b>
\$561A	TPCMD MASSERASE <b>command: device is protected</b>
\$561B	TPCMD ERASE <b>command: missing parameter</b>
\$561C	TPCMD ERASE <b>command: invalid parameter</b>
\$561D	TPCMD ERASE <b>command: Flash memory erase error</b>
\$561E	TPCMD PROGRAM <b>command: BAF flash memory program error</b>
\$561F	TPCMD ERASE <b>command: device is protected</b>
\$5620	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$5621	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$5622	TPCMD BLANKCHECK <b>command: Flash memory blank check error</b>
\$5623	TPCMD BLANKCHECK <b>command: HSMCODE flash memory blankcheck error</b>
\$5624	TPCMD BLANKCHECK <b>command: device is protected</b>
\$5625	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5626	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5627	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5628	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$5629	TPCMD PROGRAM <b>command: HSMCODE flash memory program error</b>
\$562A	TPCMD PROGRAM <b>command: device is protected</b>
\$562B	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$562C	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$562D	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$562E	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$562F	TPCMD VERIFY <b>command: HSMCODE flash memory verify error</b>
\$5630	TPCMD VERIFY <b>command: device is protected</b>
\$5631	TPSTART <b>command: execution error</b>
\$5632	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$5633	TPEND <b>command: execution error</b>

## Errors

Error #	Description
\$5634	TPCMD MASSErase <b>command: Data Flash memory mass erase error</b>
\$5635	TPCMD ERASE <b>command: Data Flash memory erase error</b>
\$5636	TPCMD BLANKCHECK <b>command: Data Flash memory blankcheck error</b>
\$5637	TPCMD PROGRAM <b>command: Data Flash memory program error</b>
\$5638	TPCMD VERIFY <b>command: Data Flash memory verify error</b>
\$5639	TPCMD ERASE <b>command: invalid parameter</b>
\$563A	TPCMD BLANKCHECK <b>command: Test Flash memory blankcheck error</b>
\$563B	TPCMD BLANKCHECK <b>command: HSMDData Flash memory blankcheck error</b>
\$563C	TPCMD PROGRAM <b>command: Test Flash memory program error</b>
\$563D	TPCMD PROGRAM <b>command: HSMDData Flash memory program error</b>
\$563E	TPCMD VERIFY <b>command: Test Flash memory verify error</b>
\$563F	TPCMD VERIFY <b>command: HSMDData Test Flash memory verify error</b>
\$5640	TPCMD READ <b>command: missing or invalid parameter</b>
\$5641	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$5642	TPCMD READ <b>command: Flash read error</b>
\$5644	TPCMD READ <b>command: Device flash secured</b>
\$5645	TPCMD DUMP <b>command: algorithm not supported</b>
\$5646	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$5647	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$5648	TPCMD DUMP <b>command: dump error</b>
\$564B	TPCMD MASSErase <b>command: BAF flash memory mass erase error</b>
\$564C	TPCMD JTAGID <b>command: invalid JTAGID</b>
\$564D	TPCMD MASSErase <b>command: HSMCODE flash memory mass erase error</b>
\$564E	TPCMD BLANKCHECK <b>command: BAF flash memory blankcheck error</b>
\$564F	TPCMD MASSErase <b>command: HSMDATA flash memory mass erase error</b>
\$5650	TPCMD MASSErase <b>command: CSE flash memory mass erase error</b>
\$5651	TPCMD PROGRAM <b>command: CSE flash memory program error</b>
\$5652	TPCMD PROGRAM <b>command: UTEST flash memory program error</b>
\$5653	TPCMD VERIFY <b>command: CSE flash memory verify error</b>
\$5654	TPCMD VERIFY <b>command: UTEST flash memory verify error</b>
\$5655	TPCMD PROGRAM <b>command: DCF record error</b>
\$5656	TPCMD VERIFY <b>command: DCF record error</b>
<b>FREESCALE MKV3/S32K14 FSL_F - Specific Errors</b>	
\$6D00	TCSETDEV <b>command: manufacturer not supported</b>
\$6D01	TCSETDEV <b>command: algorithm not found on card</b>
\$6D02	TCSETDEV <b>command: device not supported</b>
\$6D03	TCSETDEV <b>command: internal hardware configuration error</b>
\$6D04	TCSETDEV <b>command: invalid device data</b>

Error #	Description
\$6D05	TCSETDEV <b>command: This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$6D06	TCSETPAR <b>command: parameter not supported</b>
\$6D08	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$6D09	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$6D0A	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$6D0B	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$6D0C	TCSETPAR RSTTYPE <b>command: missing or invalid parameter</b>
\$6D0D	TCSETPAR CMODE <b>command: missing or invalid parameter</b>
\$6D0E	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$6D0F	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$6D10	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$6D11	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$6D12	TCSETPAR CPUFREQ <b>command: missing, invalid or out of range parameter</b>
\$6D13	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$6D14	TCSETPAR DEV <b>command: missing, invalid or out of range parameter</b>
\$6D14	TPCMD <b>command: missing parameter</b>
\$6D15	TPCMD <b>command: parameter not supported</b>
\$6D16	TPCMD RUN <b>command: execution error</b>
\$6D17	TPCMD MASSERASE <b>command: missing parameter</b>
\$6D18	TPCMD MASSERASE <b>command: invalid parameter</b>
\$6D19	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$6D1A	TPCMD MASSERASE <b>command: Data Flash memory mass erase error</b>
\$6D1B	TPCMD ERASE <b>command: missing parameter</b>
\$6D1C	TPCMD ERASE <b>command: invalid parameter</b>
\$6D1D	TPCMD ERASE <b>command: Flash memory erase error</b>
\$6D1E	TPCMD ERASE <b>command: Device Flash secured</b>
\$6D1F	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$6D20	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$6D21	TPCMD BLANKCHECK <b>command: Flash memory blank check error</b>
\$6D22	TPCMD BLANKCHECK <b>command: Data Flash memory blank check error</b>
\$6D23	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$6D24	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6D25	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$6D26	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$6D27	TPCMD PROGRAM <b>command: Data Flash memory program error</b>
\$6D28	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6D29	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6D2A	TPCMD VERIFY <b>command: source offset parameter out of range</b>

## Errors

Error #	Description
\$6D2B	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$6D2C	TPCMD VERIFY <b>command: Data Flash memory verify error</b>
\$6D2D	TPCMD READ <b>command: missing or invalid parameter</b>
\$6D2E	TPCMD READ <b>command: Flash read error</b>
\$6D2F	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$6D30	TPCMD WRITE <b>command: Flash write error</b>
\$6D31	TPSTART <b>command: execution error</b>
\$6D32	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$6D33	TPEND <b>command: execution error</b>
\$6D34	TPCMD PROGRAM <b>command: Once memory program error</b>
\$6D35	TPCMD PARTITION <b>command: Partition error</b>
\$6D36	TPCMD SET_FLEXRAM <b>command: invalid parameter</b>
\$6D37	TPCMD SET_FLEXRAM <b>command: Test Flash memory blankcheck error</b>
\$6D40	TPCMD DUMP <b>command: algorithm not supported</b>
\$6D41	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$6D42	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$6D43	TPCMD DUMP <b>command: dump error</b>
\$6D44	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$6D45	TPCMD DUMP <b>command: execution error</b>
\$6D46	TPCMD PARTITION <b>command: missing or invalid parameter</b>
\$6D47	<b>Set partition execution error</b>
<b>FREESCALE S32K116 AND TI CC2640 FSL_G - Specific Errors</b>	
\$7000	TCSETDEV <b>command: manufacturer not supported</b>
\$7001	TCSETDEV <b>command: algorithm not found on card</b>
\$7002	TCSETDEV <b>command: device not supported</b>
\$7003	TCSETDEV <b>command: internal hardware configuration error</b>
\$7004	TCSETDEV <b>command: invalid device data</b>
\$7005	TCSETDEV <b>command: This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$7006	TCSETPAR <b>command: parameter not supported</b>
\$7007	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$7008	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$7009	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$700A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$700B	TCSETPAR RSTTYPE <b>command: missing or invalid parameter</b>
\$700C	TCSETPAR CMODE <b>command: missing or invalid parameter</b>
\$700D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$700E	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$700F	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>



Error #	Description
\$7010	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$7011	TCSETPAR CPUFREQ <b>command: missing, invalid or out of range parameter</b>
\$7012	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$7013	TCSETPAR DEV <b>command: missing, invalid or out of range parameter</b>
\$7014	TPCMD <b>command: missing parameter</b>
\$7015	TPCMD <b>command: parameter not supported</b>
\$7016	TPCMD RUN <b>command: execution error</b>
\$7017	TPCMD MASSERASE <b>command: missing parameter</b>
\$7018	TPCMD MASSERASE <b>command: invalid parameter</b>
\$7019	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$701A	TPCMD MASSERASE <b>command: Data Flash memory mass erase error</b>
\$701B	TPCMD ERASE <b>command: missing parameter</b>
\$701C	TPCMD ERASE <b>command: invalid parameter</b>
\$701D	TPCMD ERASE <b>command: Flash memory erase error</b>
\$701E	TPCMD ERASE <b>command: Device Flash secured</b>
\$701F	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$7020	TPCMD BLANKCHECK <b>command: invalid parameter</b>
\$7021	TPCMD BLANKCHECK <b>command: Flash memory blank check error</b>
\$7022	TPCMD BLANKCHECK <b>command: Data Flash memory blank check error</b>
\$7023	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$7024	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$7025	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$7026	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$7027	TPCMD PROGRAM <b>command: Data Flash memory program error</b>
\$7028	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$7029	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$702A	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$702B	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$702C	TPCMD VERIFY <b>command: Data Flash memory verify error</b>
\$702D	TPCMD READ <b>command: missing or invalid parameter</b>
\$702E	TPCMD READ <b>command: Flash read error</b>
\$702F	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$7030	TPCMD WRITE <b>command: Flash write error</b>
\$7031	TPSTART <b>command: execution error</b>
\$7032	TPSTART <b>command: one or more required TCSETPAR commands have not been sent</b>
\$7033	TPEND <b>command: execution error</b>
\$7034	TPCMD PROGRAM <b>command: Once memory program error</b>
\$7035	TPCMD PARTITION <b>command: Partition error</b>

## Errors

Error #	Description
\$7036	TPCMD SET_FLEXRAM <b>command: invalid parameter</b>
\$7037	TPCMD SET_FLEXRAM <b>command: Test Flash memory blankcheck error</b>
\$7040	TPCMD DUMP <b>command: algorithm not supported</b>
\$7041	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$7042	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$7043	TPCMD DUMP <b>command: dump error</b>
\$7044	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$7045	TPCMD DUMP <b>command: execution error</b>
\$7046	TPCMD PARTITION <b>command: missing or invalid parameter</b>
\$7047	<b>Set partition execution error</b>
<b>FREESCALE MC9S12Z/MM9Z1 FSL_J- Specific Errors</b>	
\$5C00	TCSETDEV <b>command: manufacturer not supported</b>
\$5C01	TCSETDEV <b>command: algorithm not found on card</b>
\$5C02	TCSETDEV <b>command: device not supported</b>
\$5C03	TCSETDEV <b>command: invalid device data</b>
\$5C04	TCSETDEV <b>command: internal hardware configuration error</b>
\$5C45	TCSETDEV <b>command: This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$5C10	TCSETPAR <b>command: parameter not supported</b>
\$5C11	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5C12	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5C13	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$5C44	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$5C14	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$5C15	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$5C16	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$5C17	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$5C18	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$5C1A	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$5C1B	TCSETPAR REFDIV <b>command: missing, invalid or out of range parameter</b>
\$5C1C	TCSETPAR SYNDIV <b>command: missing, invalid or out of range parameter</b>
\$5C1A	TCSETPAR VCOFREQ <b>command: missing, invalid or out of range parameter</b>
\$5C1D	TPCMD <b>command: parameter not supported</b>
\$5C1E	TPCMD <b>command: invalid command</b>
\$5C1F	TPCMD RUN <b>command: execution error</b>
\$5C20	TPCMD UNSECURE <b>command: unprotect Flash and Eeprom memory error</b>
\$5C21	TPCMD MASERASE <b>command: missing parameter</b>
\$5C22	TPCMD MASERASE <b>command: invalid parameter</b>
\$5C23	TPCMD MASERASE <b>command: Flash memory mass erase error</b>

Error #	Description
\$5C24	TPCMD MASSERASE <b>command: Flash memory secured</b>
\$5C25	TPCMD MASSERASE <b>command: Eeprom memory mass erase error</b>
\$5C26	TPCMD MASSERASE <b>command: Eeprom memory secured</b>
\$5C27	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$5C28	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5C29	TPCMD BLANKCHECK <b>command: Flash memory blank check error</b>
\$5C2A	TPCMD BLANKCHECK <b>command: Flash memory secured</b>
\$5C2B	TPCMD BLANKCHECK <b>command: Eeprom memory blank check error</b>
\$5C2C	TPCMD BLANKCHECK <b>command: Eeprom memory secured</b>
\$5C2D	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5C2E	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5C2F	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5C30	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$5C31	TPCMD PROGRAM <b>command: Flash memory secured</b>
\$5C32	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$5C33	TPCMD PROGRAM <b>command: Eeprom memory secured</b>
\$5C46	TPCMD VERIFY <b>command: Device Once memory secured</b>
\$5C47	TPCMD VERIFY <b>command: Once memory verify error</b>
\$5C34	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5C35	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5C36	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5C37	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$5C38	TPCMD VERIFY <b>command: Flash memory secured</b>
\$5C39	TPCMD VERIFY <b>command: Eeprom memory verify error</b>
\$5C3A	TPCMD VERIFY <b>command: Eeprom memory secured</b>
\$5C4A	TPCMD VERIFY <b>command: Emulated Eeprom memory verify error</b>
\$5C4B	TPCMD VERIFY <b>command: Emulated Eeprom memory secured</b>
\$5C4C	TPCMD PROGRAM <b>command: Device Once memory secured</b>
\$5C4D	TPCMD PROGRAM <b>command: Once memory program error</b>
\$5C3	TPCMD READ <b>command: Once memory read error</b>
\$5C33	TPCMD READ <b>command: Once memory secured</b>
\$5C3E	TPCMD READ <b>command: missing or invalid parameter</b>
\$5C3F	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$5C40	TPCMD READ <b>command: Flash read error</b>
\$5C41	TPCMD READ <b>command: Flash memory secured</b>
\$5C42	TPCMD READ <b>command: Eeprom read error</b>
\$5C43	TPCMD READ <b>command: Eeprom memory secured</b>
\$5C3B	TPSTART <b>command: execution error</b>

## Errors

Error #	Description
\$5C3C	TFSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$5C3D	TFEND <b>command: execution error</b>
<b>FREESCALE MKE02Z64VLH4 FSL_L - Specific Errors</b>	
\$7400	TCSETDEV <b>command: manufacturer not supported</b>
\$7401	TCSETDEV <b>command: algorithm not found on card</b>
\$7402	TCSETDEV <b>command: device not supported</b>
\$7403	TCSETDEV <b>command: invalid device data</b>
\$7404	TCSETDEV <b>command: internal hardware configuration error</b>
\$7405	TCSETDEV <b>command: This programming algorithm requires a newer version of the FlashRunner firmware</b>
\$7406	TCSETPAR <b>command: parameter not supported</b>
\$7407	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$7408	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$7409	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$740A	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$740B	TCSETPAR RSTTYPE <b>command: missing or invalid parameter</b>
\$740C	TCSETPAR CMODE <b>command: missing or invalid parameter</b>
\$740D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$740E	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$740F	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$7410	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$7411	TCSETPAR CPUFREQ <b>command: missing, invalid or out of range parameter</b>
\$7412	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$7413	TCSETPAR DEV <b>command: missing, invalid or out of range parameter</b>
\$7414	TPCMD <b>command: parameter not supported</b>
\$7415	TPCMD <b>command: invalid command</b>
\$7416	TPCMD MASSERASE <b>command: missing parameter</b>
\$7417	TPCMD MASSERASE <b>command: invalid parameter</b>
\$7418	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$7419	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$7420	TPCMD BLANKCHECK <b>command: Flash memory blank check error</b>
\$7421	TPCMD BLANKCHECK <b>command: missing parameter</b>
\$7422	TPCMD PROGRAM <b>command: missing parameter</b>
\$7423	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$7424	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$7425	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$7426	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$7427	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$7428	TPCMD VERIFY <b>command: source offset parameter out of range</b>

Error #	Description
\$7429	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$742A	TPCMD READ <b>command: missing or invalid parameter</b>
\$742B	TPCMD READ <b>command: Flash read error</b>
\$742C	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$742D	TPCMD WRITE <b>command: Flash write error</b>
\$742E	TPSTART <b>command: execution error</b>
\$742F	TPSTART <b>command: one or more required</b> TCSETPAR <b>commands have not been sent</b>
\$7430	TPEND <b>command: execution error</b>
\$7431	TPCMD DUMP <b>command: algorithm not supported</b>
\$7032	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$7033	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$7034	TPCMD DUMP <b>command: dump error</b>
\$7035	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$7036	<b>dump execution error</b>
<b>FREESCALE MC9S08PA HCS08_B - Specific Errors</b>	
\$5100	TCSETDEV <b>command: manufacturer not supported</b>
\$5101	TCSETDEV <b>command: algorithm not found on card</b>
\$5102	TCSETDEV <b>command: device not supported</b>
\$5103	TCSETDEV <b>command: invalid device data</b>
\$5104	TCSETDEV <b>command: internal hardware configuration error</b>
\$5110	TCSETPAR <b>command: parameter not supported</b>
\$5111	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5112	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$5113	TPSTART <b>command: command not preceded by</b> TCSETPAR PWDOWN <b>command</b>
\$5114	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$5115	TPSTART <b>command: command not preceded by</b> TCSETPAR PWUP <b>command</b>
\$5116	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5117	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTDOWN <b>command</b>
\$5118	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$5119	TPSTART <b>command: command not preceded by</b> TCSETPAR RSTUP <b>command</b>
\$511A	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$511B	TPSTART <b>command: command not preceded by</b> TCSETPAR VDD <b>command</b>
\$511C	TCSETPAR BDM_ENTRY_MODE <b>command: missing, invalid or out of range parameter</b>
\$511D	TPSTART <b>command: command not preceded by</b> TCSETPAR BDM_ENTRY_MODE <b>command</b>
\$511E	TCSETPAR FLL_OSC <b>command: missing, invalid or out of range parameter</b>
\$511F	TPSTART <b>command: command not preceded by</b> TCSETPAR FLL_OSC <b>command</b>
\$5120	TPCMD <b>command: invalid command</b>
\$5121	TPCMD RUN <b>command: invalid command</b>

## Errors

Error #	Description
\$5122	TPCMD <b>command: command secured</b>
\$5123	TPCMD TRIM <b>command: invalid parameter</b>
\$5124	TPCMD TRIM <b>command: trim not supported</b>
\$5125	TPCMD TRIM <b>command: trim error</b>
\$5126	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5127	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5128	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5129	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$512A	TPCMD VERIFY <b>command: Eeprom memory verify error</b>
\$512B	TPCMD PROGRAM <b>command: missing parameter</b>
\$512C	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$512D	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$512E	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$512F	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$5130	TPCMD VERIFY <b>command: Once memory verify error</b>
\$5131	TPCMD UNSECURE <b>command: Device unsecure error</b>
\$5132	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$5133	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5134	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$5135	TPCMD BLANKCHECK <b>command: Eeprom memory blankcheck error</b>
\$5136	TPCMD MASSERASE <b>command: All memory masserase error</b>
\$5138	TPCMD SECTORERASE <b>command: missing or invalid parameter</b>
\$5139	TPCMD SECTORERASE <b>command: Flash sectorerase error</b>
\$513A	TPCMD SECTORERASE <b>command: Eeprom sectorerase error</b>
\$513B	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$513C	TPCMD MASSERASE <b>command: Eeprom masserase error</b>
\$513D	TPCMD BLANKCHECK <b>command: All memory blankcheck error</b>
\$513E	TPCMD SETMARGIN <b>command: missing or invalid parameter</b>
\$513F	TPCMD SETMARGIN <b>command: set margin error</b>
\$5140	TPSTART <b>command: execution error</b>
\$5141	TPEND <b>command: execution error</b>
\$5142	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$5143	TPCMD VERIFY <b>command: target memory protected</b>
\$5144	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$5145	TPCMD READ <b>command: device protected</b>
\$5146	TPCMD READ <b>command: Flash read error</b>
\$5147	TPCMD DUMP <b>command: missing, invalid or out of range parameter</b>
\$5148	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>

Error #	Description
\$5149	TPCMD DUMP command: missing, invalid or out of range parameter
\$514A	TPCMD DUMP command: missing, invalid or out of range parameter
\$514B	TPCMD DUMP command: Flash dump error
\$514C	TPCMD DUMP command: Eeprom dump error
\$514D	TPCMD DUMP command: target device protected
\$514E	TPCMD DUMP command: Algorithm not supported
\$514F	TCSETPAR RSTDRV command: missing, invalid or out of range parameter
\$5150	TPCMD command: print error
\$5154	TPCMD MASSERASE command: Flash masserase error
\$5155	TPCMD BLANKCHECK command: Section blackcheck error
<b>INFINEON TLE9844-2QX INF_F - Specific Errors</b>	
\$7200	TCSETDEV command: manufacturer not supported
\$7201	TCSETDEV command: algorithm not found on card
\$7202	TCSETDEV command: device not supported
\$7203	TCSETDEV command: internal hardware configuration error
\$7204	TCSETDEV command: invalid device data
\$7205	TCSETDEV command: Algorithm not supported
\$7206	TCSETPAR command: parameter not supported
\$7207	TCSETPAR CLKOUT command: missing, invalid or out of range parameter
\$7208	TCSETPAR RSTDOWN command: missing, invalid or out of range parameter
\$7209	TCSETPAR RSTUP command: missing, invalid or out of range parameter
\$720A	TCSETPAR RSTDRV command: missing, invalid or out of range parameter
\$720B	TCSETPAR PWDOWN command: missing, invalid or out of range parameter
\$720C	TCSETPAR PWUP command: missing, invalid or out of range parameter
\$720D	TCSETPAR VDD command: missing, invalid or out of range parameter
\$720E	TCSETPAR VDD_AUX command: missing, invalid or out of range parameter
\$720F	TCSETPAR SWCLK command: missing, invalid or out of range parameter
\$7210	TCSETPAR ENTRY_MODE command: missing, invalid or out of range parameter
\$7211	TPCMD command: missing command
\$7212	TPCMD command: invalid command
\$7213	TPCMD RUN command: invalid command
\$7214	TPCMD MASSERASE command: missing or invalid parameter
\$7215	TPCMD MASSERASE command: target start address parameter or length parameter out of range
\$7216	TPCMD MASSERASE command: flash memory masserase error
\$7217	TPCMD ERASE command: missing or invalid parameter
\$7218	TPCMD ERASE command: target start address parameter or length parameter out of range
\$7219	TPCMD ERASE command: flash memory program error
\$721A	TPCMD BLANKCHECK command: missing or invalid parameter

## Errors

Error #	Description
\$721B	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$721C	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$721D	TPCMD PROGRAM <b>command: missing parameter</b>
\$721E	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$721F	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$7220	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$7221	TPCMD VERIFY <b>command: missing parameter</b>
\$7222	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$7223	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$7224	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$7225	TPCMD READ <b>command: missing parameter</b>
\$7226	TPCMD READ <b>command: Flash read error</b>
\$7227	TPCMD WRITE <b>command: missing parameter</b>
\$7228	TPCMD WRITE <b>command: Flash write error</b>
\$7229	TPSTART <b>command: execution error</b>
\$722A	TPSTART <b>command: missing or invalid parameter</b>
\$722B	TPEND <b>command: execution error</b>
\$722C	TPCMD MASSErase <b>command: Eeprom memory masserase error</b>
\$722D	TPCMD PROGRAM <b>command: Eepom memory program error</b>
<b>FREESCALE MCF51AC MCF_B - Specific Errors</b>	
\$7300	TCSETDEV <b>command: manufacturer not supported</b>
\$7301	TCSETDEV <b>command: algorithm not found on card</b>
\$7302	TCSETDEV <b>command: device not supported</b>
\$7303	TCSETDEV <b>command: invalid device data</b>
\$7304	TCSETDEV <b>command: internal hardware configuration error</b>
\$7305	TCSETDEV <b>command: Algorithm not supported</b>
\$7310	TCSETPAR <b>command: parameter not supported</b>
\$7311	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$7312	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$7313	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$7314	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$7315	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$7316	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$7317	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$7318	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$7320	TPCMD <b>command: missing command</b>
\$7321	TPCMD <b>command: invalid command</b>
\$7322	TPCMD RUN <b>command: invalid command</b>



Error #	Description
\$7323	TPCMD UNSECURE <b>command: device unsecure error</b>
\$7324	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$7325	TPCMD MASSERASE <b>command: target start address parameter or length parameter out of range</b>
\$7326	TPCMD MASSERASE <b>command: flash memory masserase error</b>
\$7327	TPCMD MASSERASE <b>command: device secured</b>
\$7328	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$7329	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$732A	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$732B	TPCMD BLANKCHECK <b>command: device secured</b>
\$732C	TPCMD PROGRAM <b>command: missing parameter</b>
\$732D	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$732E	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$732F	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$7330	TPCMD PROGRAM <b>command: device secured</b>
\$7331	TPCMD VERIFY <b>command: missing parameter</b>
\$7332	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$7333	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$7334	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$7335	TPCMD VERIFY <b>command: device secured</b>
\$7336	TPCMD READ <b>command: Flash read error</b>
\$7337	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$7338	TPCMD READ <b>command: Flash read error</b>
\$7339	TPCMD READ <b>command: device secured</b>
\$733A	TPCMD TRIM <b>command: missing parameter</b>
\$733B	TPCMD TRIM <b>command: trim error</b>
\$733C	TPCMD TRIM <b>command: device secured</b>
\$733D	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$733E	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$733F	TPCMD DUMP <b>command: Flash dump error</b>
\$7340	TPCMD DUMP <b>command: Device secured</b>
\$7341	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$7350	TPSTART <b>command: execution error</b>
\$7351	TPSTART <b>command: missing or invalid parameter</b>
\$7352	TPEND <b>command: execution error</b>
<b>NXP NCF29A1 NCF - Specific Errors</b>	
\$6A00	TCSETDEV <b>command: Algorithm not supported</b>
\$6A01	TCSETDEV <b>command: algorithm not found on card</b>
\$6A02	TCSETDEV <b>command: device not supported</b>

## Errors

Error #	Description
\$6A03	TCSETPAR <b>command: execution error</b>
\$6A04	TPCMD <b>command: memory parameter missing</b>
\$6A05	TPCMD <b>command: memory not supported</b>
\$6A06	TPCMD <b>command: target start address parameter or length parameter out of range</b>
\$6A07	TPCMD <b>command: command not supported</b>
\$6A08	TPCMD <b>command: connection error</b>
\$6A09	TPSTART <b>command: internal hardware configuration error</b>
\$6A0A	TPCMD <b>command: unknown command</b>
\$6A0B	TPEND <b>command: execution error</b>
\$6A0C	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$6A0D	TPCMD MASSERASE <b>command: flash memory masserase error</b>
\$6A0E	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$6A0F	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$6A10	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$6A11	TPCMD PROGRAM <b>command: missing parameter</b>
\$6A12	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6A13	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$6A14	TPCMD VERIFY <b>command: missing parameter</b>
\$6A15	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6A16	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$6A17	TPCMD READ <b>command: Flash read error</b>
\$6A18	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$6A19	TPCMD READ <b>command: Flash read error</b>
\$6A1A	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$6A1B	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$6A1C	TPCMD DUMP <b>command: Flash dump error</b>
\$6A1D	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$6A1E	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$6A1F	TPCMD <b>command: invalid command</b>
\$6A20	TPCMD <b>command: entry not supported</b>
\$6A21	TCSETPAR <b>command: invalid command</b>
\$6A22	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$6A23	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$6A24	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$6A25	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$6A26	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$6A27	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$6A28	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>

Error #	Description
\$6A29	TCSETPAR BAUDRATE command: missing, invalid or out of range parameter
\$6A2A	Entry execution error
\$6A2B	Set baudrate execution error
<b>NEC UPD78F NEC78K_B - Specific Errors</b>	
\$5D00	TCSETDEV command: manufacturer not supported
\$5D01	TCSETDEV command: algorithm not found on card
\$5D02	TCSETDEV command: device not supported
\$5D03	TCSETDEV command: internal hardware configuration error
\$5D04	TCSETPAR command: parameter not supported
\$5D05	TCSETPAR CMODE command: missing parameter
\$5D06	TCSETPAR CMODE command: communication mode not supported by target device
\$5D07	TPSTART command: command not preceded by TCSETPAR CMODE command
\$5D08	TPSTART CMODE command: invalid protocol
\$5D09	TCSETPAR CLKOUT command: missing parameter
\$5D0A	TCSETPAR CLKOUT command: invalid frequency
\$5D0B	TPSTART command: command not preceded by TCSETPAR CLKOUT command
\$5D0C	TCSETPAR FOSC command: missing, invalid or out of range parameter
\$5D0D	TPSTART command: command not preceded by TCSETPAR FOSC command
\$5D0E	TCSETPAR SCLK command: missing, invalid or out of range parameter
\$5D0F	TPSTART command: command not preceded by TCSETPAR SCLK command
\$5D10	TCSETPAR VDD command: missing, invalid or out of range parameter
\$5D11	TPSTART command: command not preceded by TCSETPAR VDD command
\$5D12	TCSETPAR VDD_AUX command: missing, invalid or out of range parameter
\$5D13	TPSTART command: command not preceded by TCSETPAR VDD_AUX command
\$5D14	TCSETPAR RSTDOWN command: missing, invalid or out of range parameter
\$5D15	TPSTART command: command not preceded by TCSETPAR RSTDOWN command
\$5D16	TCSETPAR RSTUP command: missing, invalid or out of range parameter
\$5D17	TPSTART command: command not preceded by TCSETPAR RSTUP command
\$5D18	TCSETPAR PWDOWN command: missing, invalid or out of range parameter
\$5D19	TPSTART command: command not preceded by TCSETPAR PWDOWN command
\$5D1A	TCSETPAR PWUP command: missing, invalid or out of range parameter
\$5D1B	TPSTART command: command not preceded by TCSETPAR PWUP command
\$5D1C	TPCMD command: parameter not supported
\$5D1D	Can't establish communication with the target device
\$5D1E	TPCMD VERIFY command: missing or invalid parameter
\$5D1F	TPCMD VERIFY command: target start address parameter or length parameter out of range
\$5D20	TPCMD VERIFY command: source offset parameter out of range
\$5D21	TPCMD VERIFY command: Flash verify error
\$5D22	TPCMD VERIFY command: operation not supported by target device

## Errors

Error #	Description
\$5D23	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5D24	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5D25	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5D26	TPCMD PROGRAM <b>command: Flash program error</b>
\$5D27	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$5D28	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5D29	TPCMD BLANKCHECK <b>command: Flash blank check error</b>
\$5D2A	TPCMD BLANKCHECK <b>command: operation not supported by target device</b>
\$5D2B	TPCMD MASSErase <b>command: Flash mass erase error</b>
\$5D2C	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$5D2D	TPCMD PROTECT <b>command: chip protection error</b>
\$5D2E	TPCMD PROTECT <b>command: operation not supported by target device</b>
\$5D2F	TPCMD BLOCKERASE <b>command: missing or invalid parameter</b>
\$5D30	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$5D31	TPCMD BLOCKERASE <b>command: block erase error</b>
\$5D32	TPCMD BLOCKERASE <b>command: operation not supported by target device</b>
\$5D33	TPCMD RUN <b>command: execution error</b>
\$5D34	TPSTART <b>command: execution error</b>
\$5D35	TPEND <b>command: could not power target off</b>
\$5D36	TPEND <b>command: execution error</b>
\$5D37	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$5D38	TPSTART <b>command: command not preceded by TCSETPAR BAUDRATE command</b>
\$5D39	TCSETPAR RSTDRV <b>command: missing or invalid parameter</b>
\$5D3A	TCSETPAR SOPU <b>command: missing or invalid parameter</b>
\$5D3B	TCSETPAR SIPU <b>command: missing or invalid parameter</b>
\$5D40	TCSETPAR VDD_AUX <b>command: missing or invalid parameter or not allowed</b>
\$5D41	TCSETDEV <b>command: OS version not supported</b>
\$5D42	TPCMD GETSECURITY <b>command: operation not supported by target device</b>
\$5D43	TPCMD GETSECURITY <b>command: get security error</b>
\$5D44	TPCMD PROGRAMBYTE <b>command: program byte error</b>
\$5D45	TPCMD PROGRAMBYTE <b>command: operation not supported by target device</b>
<b>NXP LPC812M101J NXP_C - Specific Errors</b>	
\$5400	TCSETDEV <b>command: manufacturer not supported</b>
\$5401	TCSETDEV <b>command: algorithm not found on card</b>
\$5402	TCSETDEV <b>command: device not supported</b>
\$5403	TCSETDEV <b>command: internal hardware configuration error</b>
\$5404	TCSETDEV <b>command: invalid device data</b>
\$5405	TCSETDEV <b>command: algorithm not supported</b>

Error #	Description
\$5406	TCSETPAR <b>command: invalid command</b>
\$5407	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5408	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5409	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$540A	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$540B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$540C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$540D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$540E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$540F	TCSETPAR CPUFREQ <b>command: missing, invalid or out of range parameter</b>
\$5410	TCSETPAR SWCLK <b>command: missing, invalid or out of range parameter</b>
\$5411	TPCMD <b>command: missing command</b>
\$5412	TPCMD <b>command: invalid command</b>
\$5413	TPCMD RUN <b>command: execution error</b>
\$5414	TPCMD MASSERASE <b>command: missing parameter</b>
\$5415	TPCMD MASSERASE <b>command: invalid parameter</b>
\$5416	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$5417	TPCMD MASSERASE <b>command: Option byte mass erase error</b>
\$5418	TPCMD PAGE_ERASE <b>command: missing or invalid parameter</b>
\$5419	TPCMD PAGE_ERASE <b>command: target start address parameter or length parameter out of range</b>
\$541A	TPCMD PAGE_ERASE <b>command: Flash mass erase error</b>
\$541B	TPCMD PAGE_ERASE <b>command: device secured</b>
\$541C	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$541D	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$541E	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$541F	TPCMD BLANKCHECK <b>command: Option byte memory blankcheck error</b>
\$5420	TPCMD PROGRAM <b>command: missing parameter</b>
\$5421	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5422	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5423	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$5424	TPCMD PROGRAM <b>command: Option byte memory program error</b>
\$5425	TPCMD VERIFY <b>command: missing parameter</b>
\$5426	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5427	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5428	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$5429	TPCMD VERIFY <b>command: Option byte memory verify error</b>
\$542A	TPCMD READ <b>command: missing or invalid parameter</b>
\$542B	TPCMD READ <b>command: memory read error</b>

## Errors

Error #	Description
\$542C	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$542E	TPCMD WRITE <b>command: memory write error</b>
\$542F	TPSTART <b>command: execution error</b>
\$5430	TPSTART <b>command: missing parameter</b>
\$5431	TPEND <b>command: execution error</b>
\$5432	TPCMD SECTORERASE <b>command: missing parameter</b>
\$5433	TPCMD SECTORERASE <b>command: Flash memory sector erase error</b>
\$5434	TPCMD SECTORERASE <b>command: invalid parameter</b>
\$5435	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
<b>MICROCHIP PIC18F24K40 PIC18FK - Specific Errors</b>	
\$6400	TCSETDEV <b>command: manufacturer not supported</b>
\$6401	TCSETDEV <b>command: algorithm not found on card</b>
\$6402	TCSETDEV <b>command: device not supported</b>
\$6403	TCSETDEV <b>command: internal hardware configuration error</b>
\$6404	TCSETPAR <b>command: invalid command</b>
\$6405	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$6406	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$6407	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$6408	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$6409	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$640A	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$640B	TCSETPAR PROGMODE <b>command: missing, invalid or out of range parameter</b>
\$640C	TPSTART <b>command: command not preceded by TCSETPAR PROGMODE command</b>
\$640D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$640E	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$640F	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$6410	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$6411	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$6412	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$641F	TPCMD <b>command: missing command</b>
\$6420	TPCMD MASSERASE <b>command: missing parameter</b>
\$6421	TPCMD MASSERASE <b>command: invalid parameter</b>
\$6422	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$6423	TPCMD MASSERASE <b>command: Eeprom mass erase error</b>
\$6424	TPCMD MASSERASE <b>command: chip erase error</b>
\$6425	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$6426	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$6427	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>

Error #	Description
\$6428	TPCMD BLANKCHECK <b>command: Eeprom memory blankcheck error</b>
\$6429	TPCMD PROGRAM <b>command: missing parameter</b>
\$642A	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$642B	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$642C	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$642D	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$642E	TPCMD PROGRAM <b>command: OSCCAL memory program error</b>
\$642F	TPCMD VERIFY <b>command: missing parameter</b>
\$6430	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6431	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$6432	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$6433	TPCMD VERIFY <b>command: Eeprom memory verify error</b>
\$6434	TPCMD READ <b>command: missing or invalid parameter</b>
\$6425	TPCMD READ <b>command: Flash read error</b>
\$6436	TPCMD READ <b>command: Eeprom read error</b>
\$6437	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$6438	TPCMD RUN <b>command: execution error</b>
\$6439	TPEND <b>command: execution error</b>
\$643A	TPSTART <b>command: execution error</b>
<b>RENESAS R7F701xx RH850 - Specific Errors</b>	
\$5700	TCSETDEV <b>command: manufacturer not supported</b>
\$5701	TCSETDEV <b>command: algorithm not found on card</b>
\$5702	TCSETDEV <b>command: device not supported</b>
\$5703	TCSETDEV <b>command: internal hardware configuration error</b>
\$5704	TCSETPAR <b>command: invalid command</b>
\$5705	TCSETPAR CMODE <b>command: missing parameter</b>
\$5706	TCSETPAR CMODE <b>command: communication mode not supported by target device</b>
\$5707	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>
\$5708	TPSTART CMODE <b>command: invalid protocol</b>
\$5709	TCSETPAR CLKOUT <b>command: missing parameter</b>
\$570A	TCSETPAR CLKOUT <b>command: invalid frequency</b>
\$570B	TPSTART <b>command: command not preceded by TCSETPAR CLKOUT command</b>
\$570C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$570D	TPSTART <b>command: command not preceded by TCSETPAR FOSC command</b>
\$570E	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$570F	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$5710	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$5711	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>

## Errors

Error #	Description
\$5712	TCSETPAR VDD_AUX command: missing, invalid or out of range parameter
\$5713	TPSTART command: command not preceded by TCSETPAR VDD_AUX command
\$5714	TCSETPAR RSTDOWN command: missing, invalid or out of range parameter
\$5715	TPSTART command: command not preceded by TCSETPAR RSTDOWN command
\$5716	TCSETPAR RSTUP command: missing, invalid or out of range parameter
\$5717	TPSTART command: command not preceded by TCSETPAR RSTUP command
\$5718	TCSETPAR PWDOWN command: missing, invalid or out of range parameter
\$5719	TPSTART command: command not preceded by TCSETPAR PWDOWN command
\$571A	TCSETPAR PWUP command: missing, invalid or out of range parameter
\$571B	TPSTART command: command not preceded by TCSETPAR PWUP command
\$571C	TPCMD command: execution error
\$571D	TPCMD command: entry error
\$571E	TPCMD VERIFY command: missing parameter
\$571F	TPCMD VERIFY command: target start address parameter or length parameter out of range
\$5720	TPCMD VERIFY command: source offset parameter out of range
\$5721	TPCMD VERIFY command: Flash memory verify error
\$5722	TCSETPAR CALC_CHKS command: missing, invalid or out of range parameter
\$5723	TPCMD PROGRAM command: missing parameter
\$5724	TPCMD PROGRAM command: target start address parameter or length parameter out of range
\$5725	TPCMD PROGRAM command: source offset parameter out of range
\$5726	TPCMD PROGRAM command: Flash memory program error
\$5727	TPCMD BLANKCHECK command: missing or invalid parameter
\$5728	TPCMD BLANKCHECK command: target start address parameter or length parameter out of range
\$5729	TPCMD BLANKCHECK command: Flash memory blankcheck error
\$572A	TPCMD BLANKCHECK command: block erase not supported
\$572B	TPCMD MASSErase command: Flash mass erase error
\$572C	TPCMD PROTECT command: missing or invalid parameter
\$572D	TPCMD PROTECT command: Flash memory protect error
\$572E	TPCMD PROTECT command: command not supported
\$572F	TPCMD BLOCKERASE command: missing or invalid parameter
\$5730	TPCMD BLOCKERASE command: target start address parameter or length parameter out of range
\$5731	TPCMD BLOCKERASE command: Flash memory block erase error
\$5732	TPCMD BLOCKERASE command: block erase not supported
\$5733	TPCMD RUN command: execution error
\$5734	TPSTART command: execution error
\$5735	TPEND command: execution error
\$5736	TCSETPAR BAUDRATE command: missing, invalid or out of range parameter
\$5737	TPSTART command: command not preceded by TCSETPAR BAUDRATE command



Error #	Description
\$5738	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$5739	TPCMD READ <b>command: missing or invalid parameter</b>
\$573A	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$573B	TPCMD READ <b>command: read not supported</b>
\$573C	TPCMD READ <b>command: Flash read error</b>
\$573D	TPCMD VERIFY <b>command: Eeprom memory verify error</b>
\$573E	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$573F	TPCMD BLANKCHECK <b>command: Eeprom memory blank check error</b>
\$5740	TPCMD MASSERASE <b>command: Eeprom memory mass erase error</b>
\$5741	TPCMD MASSERASE <b>command: target start address parameter or length parameter out of range</b>
\$5742	TPCMD BLOCKERASE <b>command: Eeprom memory block erase error</b>
\$5743	TPCMD READ <b>command: Eeprom memory read error</b>
\$5744	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$5745	TCSETPAR SIPU <b>command: missing, invalid or out of range parameter</b>
\$5746	TCSETPAR SOPU <b>command: missing, invalid or out of range parameter</b>
\$5747	TPCMD DUMP <b>command: algorithm not supported</b>
\$5748	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$5749	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$574A	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$574B	TPCMD DUMP <b>command: Flash dump error</b>
\$574C	TPCMD DUMP <b>command: Eeprom dump error</b>
\$574D	TPCMD VALIDATE_ICU_S <b>command: ICU_S validate error</b>
\$574E	TPCMD MASSERASE <b>command: ICU mass erase error</b>
\$574F	TPCMD VERIFY <b>command: Option byte memory verify error</b>
\$5750	TPCMD GET_LOCK_BITS <b>command: Get lock bits error</b>
\$5751	TPCMD SET_LOCK_BITS <b>command: Set lock bits error</b>
\$5752	TPCMD DISABLE_LOCK_BITS <b>command: Disable lock bits error</b>
\$5753	TPCMD ENABLE_LOCK_BITS <b>command: Enable lock bits error</b>
\$5754	TPCMD SERIAL_PROGRAMMING <b>command: Disable serial programming</b>
\$5755	TPCMD SET_OTP <b>command: target start address parameter or length parameter out of range</b>
\$5756	TPCMD GET_OTP <b>command: Get otp error</b>
\$5757	TPCMD VERIFY <b>command: Option byte memory verify error</b>
\$5758	TPCMD VERIFY <b>command: Extended option byte memory verify error</b>
\$576F	TCSETDEV <b>command: OS version is not supported</b>
<b>RENESAS R5F564ML RNSS_H - Specific Errors</b>	
\$6C00	TCSETDEV <b>command: manufacturer not supported</b>
\$6C01	TCSETDEV <b>command: algorithm not found on card</b>
\$6C02	TCSETDEV <b>command: device not supported</b>

## Errors

Error #	Description
\$6C03	TCSETDEV <b>command: internal hardware configuration error</b>
\$6C04	TCSETPAR <b>command: invalid command</b>
\$6C05	TCSETPAR CLKOUT <b>command: invalid clock frequency</b>
\$6C06	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$6C07	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$6C08	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$6C09	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$6C0A	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$6C0B	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$6C0C	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$6C0D	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$6C0E	TPSTART <b>command: command not preceded by TCSETPAR RSTUP command</b>
\$6C0F	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$6C10	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$6C11	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$6C12	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$6C13	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$6C14	TPSTART <b>command: command not preceded by TCSETPAR BAUDRATE command</b>
\$6C15	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$6C16	TPSTART <b>command: command not preceded by TCSETPAR FOSC command</b>
\$6C17	TCSETPAR ID_CODE <b>command: missing, invalid or out of range parameter</b>
\$6C18	TPSTART <b>command: command not preceded by TCSETPAR ID_CODE command</b>
\$6C19	TCSETPAR OFS <b>command: missing, invalid or out of range parameter</b>
\$6C1A	TPSTART <b>command: command not preceded by TCSETPAR OFS command</b>
\$6C1B	TCSETPAR CKP <b>command: missing, invalid or out of range parameter</b>
\$6C1C	TPSTART <b>command: command not preceded by TCSETPAR ENDIAN command</b>
\$6C1D	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$6C1E	TPSTART <b>command: command not preceded by TCSETPAR CMODE command</b>
\$6C1F	TCSETPAR DIO1 <b>command: missing, invalid or out of range parameter</b>
\$6C20	TPSTART <b>command: command not preceded by TCSETPAR DIO1 command</b>
\$6C21	TCSETPAR DIO2 <b>command: missing, invalid or out of range parameter</b>
\$6C22	TPSTART <b>command: command not preceded by TCSETPAR DIO2 command</b>
\$6C23	TCSETPAR DIO3 <b>command: missing, invalid or out of range parameter</b>
\$6C24	TPSTART <b>command: command not preceded by TCSETPAR DIO3 command</b>
\$6C25	TCSETPAR DIO4 <b>command: missing, invalid or out of range parameter</b>
\$6C26	TPSTART <b>command: command not preceded by TCSETPAR DIO4 command</b>
\$6C27	TPCMD <b>command: missing command</b>
\$6C28	TPCMD RUN <b>command: execution error</b>

Error #	Description
\$6C29	TPCMD UNLOCK <b>command: Unlock device error</b>
\$6C2A	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$6C2C	TPCMD MASSERASE <b>command: Flash mass erase error</b>
\$6C2D	TPCMD BLOCKERASE <b>command: missing or invalid parameter</b>
\$6C2E	TPCMD BLOCKERASE <b>command: target start address parameter or length parameter out of range</b>
\$6C2F	TPCMD BLOCKERASE <b>command: Flash block erase error</b>
\$6C30	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$6C31	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$6C32	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$6C33	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$6C34	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6C35	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$6C36	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$6C37	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6C38	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6C39	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$6C3A	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$6C3B	TPCMD READ <b>command: missing or invalid parameter</b>
\$6C3C	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$6C3D	TPCMD READ <b>command: Flash read error</b>
\$6C3E	TPSTART <b>command: execution error</b>
\$6C3F	TPEND <b>command: execution error</b>
\$6C40	TPCMD READ_INFO <b>command: read information error</b>
\$6C41	TCSETDEV <b>command: Algorithm not supported</b>
\$6C42	TCSETPAR TXDDRV <b>command: missing, invalid or out of range parameter</b>
\$6C44	TPSTART <b>command: command not preceded by TCSETPAR ID_CODE command</b>
\$6C45	TCSETPAR ID_CODE <b>command: missing, invalid or out of range parameter</b>
\$6C46	TPCMD ENABLE <b>command: missing or invalid parameter</b>
\$6C47	TPCMD DISABLE <b>command: missing or invalid parameter</b>
\$6C48	TPCMD ENABLE <b>command: lock bits execution error</b>
\$6C49	TPCMD DISABLE <b>command: lock bits execution error</b>
\$6C4A	TPCMD LB_ENABLE <b>command: enable lock bits error</b>
\$6C4B	TPCMD LB_DISABLE <b>command: enable lock bits error</b>
\$6C4C	TPCMD ENABLE <b>command: enable acknowledge error</b>
\$6C4D	TPCMD DISABLE <b>command: disable acknowledge error</b>
\$6C4E	<b>Lock bits not supported</b>
\$6C4F	<b>Set protection execution error</b>
\$6C50	TPCMD PROGRAM_OFS <b>command: OFS program error</b>

## Errors

Error #	Description
\$6C51	TPCMD ENDIAN <b>command: Set endian error</b>
<b>STMICROELECTRONICS M95320 SERMEM_C - Specific Errors</b>	
\$6300	TCSETDEV <b>command: manufacturer not supported</b>
\$6301	TCSETDEV <b>command: algorithm not found on card</b>
\$6302	TCSETDEV <b>command: device not supported</b>
\$6303	TCSETDEV <b>command: internal hardware configuration error</b>
\$6304	TCSETPAR <b>command: invalid command</b>
\$6305	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$6306	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$6307	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$6308	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$6309	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$630A	TPSTART <b>command: command not preceded by TCSETPAR VDD_AUX command</b>
\$630B	TCSETPAR DATAWIDTH <b>command: missing, invalid or out of range parameter</b>
\$630C	TPSTART <b>command: command not preceded by TCSETPAR DATAWIDTH command</b>
\$630D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$630E	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$630F	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$6310	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$6311	TPCMD <b>command: missing command</b>
\$6312	TPCMD ENTRY <b>command: execution error</b>
\$6313	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6314	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6315	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$6316	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$6317	TPCMD VERIFY <b>command: Status register verify error</b>
\$6318	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$6319	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$631A	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$631B	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$631C	TPCMD PROGRAM <b>command: Status register program error</b>
\$631D	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$631E	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$631F	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$6320	TPCMD READ <b>command: missing or invalid parameter</b>
\$6321	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$6322	TPCMD READ <b>command: Flash memory read error</b>
\$6323	TPCMD READ <b>command: Status register read error</b>

Error #	Description
\$6324	TPCMD CHIP_ERASE <b>command: chip erase error</b>
\$6325	TPCMD CHIP_ERASE <b>command: chip erase not supported</b>
\$6326	TPCMD SECTOR_ERASE <b>command: missing or invalid parameter</b>
\$6327	TPCMD SECTOR_ERASE <b>command: sector erase error</b>
\$6328	TPCMD SECTOR_ERASE <b>command: sector erase not supported</b>
\$6329	TPCMD BLOCKERASE <b>command missing or invalid parameter</b>
\$632A	TPCMD BLOCKERASE <b>command: block erase error</b>
\$632B	TPCMD PROGRAM <b>command: Configuration register program error</b>
\$632C	TCSETPAR CTRL_ON_RLY <b>command: missing, invalid or out of range parameter</b>
\$6330	TPCMD RUN <b>command: execution error</b>
\$6331	TPSTART <b>command: execution error</b>
\$6332	TPEND <b>command: Target off error</b>
\$6333	TPEND <b>command: execution error</b>
\$6334	TCSETPAR DEVICE_ADDR <b>command: missing, invalid or out of range parameter</b>
\$6335	TPSTART <b>command not preceded by TCSETPAR DEVICE_ADDR command</b>
\$6336	TCSETPAR DATAWIDTH <b>command: missing, invalid or out of range parameter</b>
\$6337	TCSETPAR DEVICE_ADDR <b>command: device address not supported</b>
\$6338	TPCMD ENABLE <b>command: missing or invalid parameter</b>
\$6339	TPCMD ENABLE <b>command: enable lock bits execution error</b>
\$633A	TPCMD ENABLE <b>command: enable lock bits not supported</b>
\$633B	TPCMD DISABLE <b>command: missing or invalid parameter</b>
\$633C	TPCMD DISABLE <b>command: disable lock bits execution error</b>
\$633D	TPCMD DISABLE <b>command: disable lock bits not supported</b>
\$633E	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$633F	TPCMD ERASE <b>command: erase special register error</b>
\$6340	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$6341	TCSETPAR PAGESIZE <b>command: missing, invalid or out of range parameter</b>
\$6342	TCSETPAR PAGESIZE <b>command: page size not supported</b>
\$6343	TPSTART <b>command: command not preceded by TCSETPAR PAGESIZE command</b>
\$6344	TPCMD BLOCKERASE <b>command: block erase not supported</b>
\$6345	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$6346	TPCMD PROTECT <b>command: sector protect error</b>
\$6347	TPCMD PROTECT <b>command: sector protect not supported</b>
\$6348	TPCMD UNPROTECT <b>command: missing or invalid parameter</b>
\$6349	TPCMD UNPROTECT <b>command: sector unprotect error</b>
\$634A	TPCMD UNPROTECT <b>command: sector unprotect not supported</b>
\$634B	TPCMD PROGRAM <b>command: Info page program error</b>
\$634C	TPCMD READ <b>command: Lock register read error</b>

## Errors

Error #	Description
\$634D	TPCMD PROGRAM <b>command: Lock register program error</b>
\$634E	TCSETPAR DEVICE_SKIPIDCODE <b>command: missing, invalid or out of range parameter</b>
\$634F	TCSETDEV <b>command: Algorithm not supported</b>
<b>STMICROELECTRONICS STM32F ST_CAN - Specific Errors</b>	
\$4E00	TCSETDEV <b>command: manufacturer not supported</b>
\$4E01	TCSETDEV <b>command: algorithm not found on card</b>
\$4E02	TCSETDEV <b>command: device not supported</b>
\$4E03	TCSETDEV <b>command: internal hardware configuration error</b>
\$4E04	TCSETDEV <b>command: invalid device data</b>
\$4E05	TCSETDEV <b>command: algorithm not supported</b>
\$4E06	TCSETPAR <b>command: invalid command</b>
\$4E07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$4E08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$4E09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$4E0A	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$4E0B	TCSETPAR RSTTYPE <b>command: missing, invalid or out of range parameter</b>
\$4E0C	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$4E0D	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$4E0E	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$4E0F	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$4E10	TCSETPAR CPUFREQ <b>command: missing, invalid or out of range parameter</b>
\$4E11	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$4E12	TPCMD <b>command: execution error</b>
\$4E13	TPCMD <b>command: invalid command</b>
\$4E14	TPCMD RUN <b>command: execution error</b>
\$4E15	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$4E16	TPCMD MASSERASE <b>command: target start address parameter or length parameter out of range</b>
\$4E17	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$4E18	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$4E19	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$4E1A	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$4E1B	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$4E1C	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$4E1D	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$4E1E	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$4E1F	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$4E20	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$4E21	TPCMD VERIFY <b>command: source offset parameter out of range</b>

Error #	Description
\$4E22	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$4E23	TPCMD READ <b>command: missing or invalid parameter</b>
\$4E24	TPCMD READ <b>command: Flash read error</b>
\$4E25	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$4E26	TPSTART <b>command: execution error</b>
\$4E27	TPSTART <b>command: missing or invalid parameter</b>
\$4E28	TPEND <b>command: execution error</b>
<b>INFINEON C161CS STM_IN_A - Specific Errors</b>	
\$2100	TCSETDEV <b>command: manufacturer not supported</b>
\$2101	TCSETDEV <b>command: algorithm not found on card</b>
\$2102	TCSETDEV <b>command: device not supported</b>
\$2103	TCSETDEV <b>command: invalid device data</b>
\$2104	TCSETDEV <b>command: internal hardware configuration error</b>
\$2105	TCSETPAR <b>command: invalid command</b>
\$2106	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$2107	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$2108	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$2109	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$210A	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$210B	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$210C	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$210D	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$212C	TCSETPAR FOSC <b>command: missing, invalid or out of range parameter</b>
\$212D	TCSETPAR PLLMUL <b>command: missing, invalid or out of range parameter</b>
\$212E	TCSETPAR PLLIDIV <b>command: missing, invalid or out of range parameter</b>
\$212F	TCSETPAR PLLDIV <b>command: missing, invalid or out of range parameter</b>
\$2130	TCSETPAR PLLFREQ <b>command: missing, invalid or out of range parameter</b>
\$2131	TPSTART <b>command: command not preceded by TCSETPAR PLLFREQ command</b>
\$2137	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$210E	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$2138	TCSETPAR BAUDRATE <b>command: communication baudrate not UART</b>
\$210F	TPSTART <b>command: command not preceded by TCSETPAR BAUDRATE command</b>
\$2139	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$213A	TCSETPAR JTCLK <b>command: communication baudrate not JTAG</b>
\$213B	TPSTART <b>command: command not preceded by TCSETPAR JTCLK command</b>
\$2110	TPCMD <b>command: missing command</b>
\$2111	TPCMD <b>command: invalid command</b>
\$2112	TPCMD RUN <b>command: execution error</b>

## Errors

Error #	Description
\$2113	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$2114	TPCMD MASSERASE <b>command: target start address parameter or length parameter out of range</b>
\$2115	TPCMD MASSERASE <b>command: Flash memory masserase error</b>
\$2116	TPCMD MASSERASE <b>command: Flash device secured</b>
\$2117	TPCMD SECTORERASE <b>command: missing or invalid parameter</b>
\$2118	TPCMD SECTORERASE <b>command: target start address parameter or length parameter out of range</b>
\$2119	TPCMD SECTORERASE <b>command: Flash memory sector erase error</b>
\$211A	TPCMD SECTORERASE <b>command: Flash device secured</b>
\$211B	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$211C	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$211D	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$211E	TPCMD BLANKCHECK <b>command: Flash device secured</b>
\$211F	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$2120	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$2121	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$2122	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$2123	TPCMD PROGRAM <b>command: Flash device secured</b>
\$2124	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$2125	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$2126	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$2127	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$2128	TPCMD VERIFY <b>command: Flash device secured</b>
\$2132	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$2133	TPCMD PROTECT <b>command: sector protect error</b>
\$2134	TPCMD UNPROTECT <b>command: missing or invalid parameter</b>
\$2135	TPCMD UNPROTECT <b>command: sector unprotect error</b>
\$2129	TFSTART <b>command: execution error</b>
\$212A	TFSTART <b>command: missing or invalid parameter</b>
\$213C	TFSTART <b>command: Load ukernel FPGA</b>
\$212B	TFEND <b>command: execution error</b>
\$2136	TCSETPAR RSTDEV <b>command: missing, invalid or out of range parameter</b>
<b>TI SC4C1/TM4C1 TI_E - Specific Errors</b>	
\$5F00	TCSETDEV <b>command: manufacturer not supported</b>
\$5F01	TCSETDEV <b>command: algorithm not found on card</b>
\$5F02	TCSETDEV <b>command: device not supported</b>
\$5F03	TCSETDEV <b>command: internal hardware configuration error</b>
\$5F04	TCSETDEV <b>command: invalid device data</b>
\$5F05	TCSETDEV <b>command: algorithm not supported</b>



Error #	Description
\$5F06	TCSETPAR <b>command: invalid command</b>
\$5F07	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5F08	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5F09	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$5F0A	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$5F0B	TCSETPAR RSTTYPE <b>command: missing, invalid or out of range parameter</b>
\$5F0C	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$5F0D	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$5F0E	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$5F0F	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$5F10	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$5F11	TCSETPAR CPUFREQ <b>command: missing, invalid or out of range parameter</b>
\$5F12	TCSETPAR JTCLK <b>command: missing, invalid or out of range parameter</b>
\$5F13	TCSETPAR DEV <b>command: missing, invalid or out of range parameter</b>
\$5F14	TPCMD <b>command: missing command</b>
\$5F15	TPCMD <b>command: invalid command</b>
\$5F16	TPCMD RUN <b>command: execution error</b>
\$5F17	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$5F18	TPCMD MASSERASE <b>command: target start address parameter or length parameter out of range</b>
\$5F19	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$5F1A	TPCMD MASSERASE <b>command: Data Flash memory mass erase error</b>
\$5F1B	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$5F1C	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$5F1D	TPCMD ERASE <b>command: Flash memory erase error</b>
\$5F1E	TPCMD ERASE <b>command: Flash device secured</b>
\$5F1F	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$5F20	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5F21	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$5F22	TPCMD BLANKCHECK <b>command: Data flash memory blankcheck error</b>
\$5F23	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5F24	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5F25	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5F26	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$5F27	TPCMD PROGRAM <b>command: Data flash memory program error</b>
\$5F28	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5F29	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5F2A	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5F2B	TPCMD VERIFY <b>command: Flash memory verify error</b>

## Errors

Error #	Description
\$5F2C	TPCMD VERIFY <b>command: Data Flash memory verify error</b>
\$5F2D	TPCMD READ <b>command: missing or invalid parameter</b>
\$5F2E	TPCMD READ <b>command: Flash read error</b>
\$5F2F	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$5F30	TPCMD WRITE <b>command: Flash write error</b>
\$5F31	TPSTART <b>command: execution error</b>
\$5F32	TPSTART <b>command: missing or invalid parameter</b>
\$5F33	TPEND <b>command: execution error</b>
\$5F40	TPCMD DUMP <b>command: algorithm not supported</b>
\$5F41	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$5F42	TPCMD DUMP <b>command: missing or invalid parameter</b>
\$5F43	TPCMD DUMP <b>command: Flash dump error</b>
\$5F44	TPCMD DUMP <b>command: target start address parameter or length parameter out of range</b>
\$5F45	<b>dump execution error</b>
\$5F46	TPCMD PROTECT <b>command: flash protect error</b>
\$5F47	TPCMD PROTECT <b>command: target start address parameter or length parameter out of range</b>
\$5F48	TPCMD PROTECT <b>command: missing or invalid parameter</b>
\$5F49	<b>Flash protect execution error</b>
\$5F4A	<b>Flash protect execution timeout</b>
\$5F4B	<b>Flash protect not supported</b>
<b>UBLOX UBXM8030 UBX - Specific Errors</b>	
\$6000	TCSETDEV <b>command: manufacturer not supported</b>
\$6001	TCSETDEV <b>command: algorithm not found on card</b>
\$6002	TCSETDEV <b>command: device not supported</b>
\$6003	TCSETDEV <b>command: internal hardware configuration error</b>
\$6004	TCSETPAR <b>command: invalid command</b>
\$6005	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$6006	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$6007	TCSETPAR BAUDRATE <b>command: missing, invalid or out of range parameter</b>
\$6008	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$6009	TPCMD <b>command: missing command</b>
\$600A	TPCMD ENTRY <b>command: entry execution error</b>
\$600B	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$600C	TPSTART <b>command: command not preceded by TCSETPAR BAUDRATE command</b>
\$600D	TPSTART <b>command: execution error</b>
\$600E	TPEND <b>command: execution error</b>
\$6010	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$6011	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$6012	TPCMD BLANKCHECK <b>command: One time programmable memory blankcheck error</b>

Error #	Description
\$6013	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$6014	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6015	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$6016	TPCMD PROGRAM <b>command: One time programmable memory program error</b>
\$6017	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6018	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6019	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$601A	TPCMD VERIFY <b>command: One time programmable memory verify error</b>
\$601B	TPCMD READ <b>command: missing or invalid parameter</b>
\$601C	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$601D	TPCMD READ <b>command: Flash read error</b>
\$601E	TPCMD PRINT_PROGRAM_ACK <b>command: print programming acknowledge error</b>
\$601F	TPCMD RUN <b>command: execution error</b>
<b>INFINEON XMC000 XMC - Specific Errors</b>	
\$5900	TCSETDEV <b>command: manufacturer not supported</b>
\$5901	TCSETDEV <b>command: algorithm not found on card</b>
\$5902	TCSETDEV <b>command: device not supported</b>
\$5903	TCSETDEV <b>command: internal hardware configuration error</b>
\$5904	TCSETDEV <b>command: invalid device data</b>
\$5905	TCSETDEV <b>command: algorithm not supported</b>
\$5906	TCSETPAR <b>command: invalid command</b>
\$5907	TCSETPAR CLKOUT <b>command: missing, invalid or out of range parameter</b>
\$5908	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5909	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$590A	TCSETPAR RSTDRV <b>command: missing, invalid or out of range parameter</b>
\$5932	TCSETPAR RSTTYPE <b>command: missing, invalid or out of range parameter</b>
\$590B	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$590C	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$590D	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$590E	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$590F	TCSETPAR CPUFREQ <b>command: missing, invalid or out of range parameter</b>
\$5910	TCSETPAR SWCLK <b>command: missing, invalid or out of range parameter</b>
\$5911	TPCMD <b>command: missing command</b>
\$5912	TPCMD <b>command: invalid command</b>
\$5913	TPCMD RUN <b>command: execution error</b>
\$5914	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$5915	TPCMD MASSERASE <b>command: target start address parameter or length parameter out of range</b>
\$5916	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>

## Errors

Error #	Description
\$5917	TPCMD MASSErase <b>command: Option byte memory mass erase error</b>
\$5918	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$5919	TPCMD ERASE <b>command: target start address parameter or length parameter out of range</b>
\$591A	TPCMD ERASE <b>command: Flash memory erase error</b>
\$591B	TPCMD ERASE <b>command: Flash device secured</b>
\$591C	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$591D	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$591E	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$591F	TPCMD BLANKCHECK <b>command: Option byte memory blankcheck error</b>
\$5920	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5921	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5922	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$5923	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$5924	TPCMD PROGRAM <b>command: Option byte memory program error</b>
\$5925	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5926	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5927	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5928	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$5929	TPCMD VERIFY <b>command: Option byte memory verify error</b>
\$592A	TPCMD READ <b>command: missing or invalid parameter</b>
\$592B	TPCMD READ <b>command: Flash read error</b>
\$592C	TPCMD WRITE <b>command: missing or invalid parameter</b>
\$592E	TPCMD WRITE <b>command: Flash write error</b>
\$592F	TPSTART <b>command: execution error</b>
\$5930	TPSTART <b>command: missing or invalid parameter</b>
\$5931	TPEND <b>command: execution error</b>
\$5932	TPCMD MASSErase <b>command: Eeprom memory mass erase error</b>
\$5933	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$5934	TPCMD UNPROTECT <b>command: memory unprotect error</b>
\$5935	TPCMD READ_ID <b>command: read chip ID error</b>
\$5936	TCSETPAR ENTRY_RETRIES <b>command: missing, invalid or out of range parameter</b>
\$5937	TCSETPAR RESTORE_BMI <b>command: missing, invalid or out of range parameter</b>
\$5938	TPCMD RESTORE_BMI <b>command: Change boot mode index error</b>
<b>ANALOGDEVICES ADP1048W ANDEV_A - Specific Errors</b>	
\$6B00	TCSETDEV <b>command: manufacturer not supported</b>
\$6B01	TCSETDEV <b>command: algorithm not found on card</b>
\$6B02	TCSETDEV <b>command: device not supported</b>
\$6B03	TCSETDEV <b>command: internal hardware configuration error</b>

Error #	Description
\$6B04	TCSETPAR <b>command: invalid command</b>
\$6B05	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$6B06	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$6B07	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$6B08	TPSTART <b>command: command not preceded by TCSETPAR PWUP command</b>
\$6B09	TCSETPAR PWDOWN <b>command: missing, invalid or out of range parameter</b>
\$6B0A	TPSTART <b>command: command not preceded by TCSETPAR PWDOWN command</b>
\$6B0B	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$6B0C	TPSTART <b>command: command not preceded by TCSETPAR RSTDOWN command</b>
\$6B0D	TCSETPAR SCLK <b>command: missing, invalid or out of range parameter</b>
\$6B0E	TPSTART <b>command: command not preceded by TCSETPAR SCLK command</b>
\$6B0F	TCSETPAR PMB_ADDR <b>command: missing, invalid or out of range parameter</b>
\$6B10	TPSTART <b>command: command not preceded by TCSETPAR PMB_ADDR command</b>
\$6B11	TCSETPAR VDD_AUX <b>command: missing, invalid or out of range parameter</b>
\$6B12	TPCMD <b>command: missing or invalid command</b>
\$6B13	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
\$6B14	TPCMD MASSERASE <b>command: Flash memory mass erase error</b>
\$6B15	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$6B16	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$6B17	TPCMD BLANKCHECK <b>command: Eeprom memory blankcheck error</b>
\$6B18	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$6B19	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$6B1A	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$6B1B	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$6B1C	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$6B1D	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$6B1E	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$6B1F	TPCMD VERIFY <b>command: Eeprom memory verify error</b>
\$6B20	TPCMD READ <b>command: missing or invalid parameter</b>
\$6B21	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$6B22	TPCMD READ <b>command: Eeprom read error</b>
\$6B23	TPCMD ERASE <b>command: missing or invalid parameter</b>
\$6B24	TPCMD ERASE <b>command: Flash erase error</b>
\$6B25	TPCMD UNLOCK <b>command: Unlock Eeprom error</b>
\$6B26	TPCMD UNLOCK <b>command: missing or invalid parameter</b>
\$6B27	TPSTART <b>command: execution error</b>
\$6B28	TPEND <b>command: execution error</b>
\$6B29	TPCMD ENTRY <b>command: execution error</b>

## Errors

Error #	Description
\$6B2A	<b>Can't verify data mismatch</b>
\$6B2B	TPCMD BLANKCHECK <b>command: Unlock blankcheck error</b>
\$6B2C	TPCMD VERIFY <b>command: Unlock memory verify error</b>
\$6B2D	TPCMD READ <b>command: Unlock memory read error</b>
\$6B2E	TPCMD LOCK <b>command: missing or invalid parameter</b>
\$6B2F	TPCMD LOCK <b>command: Eeprom lock error</b>
\$6B30	TPCMD CHANGE_PSW <b>command: missing or invalid parameter</b>
\$6B31	TPCMD CHANGE_PSW <b>command: change password error</b>
\$6B32	TPCMD EEPROM_CRC_CHECKSUM <b>command: missing or invalid parameter</b>
\$6B33	TPCMD EEPROM_CRC_CHECKSUM <b>command: Eeprom CRC checksum error</b>
\$6B34	<b>Eeprom CRC checksum execution error</b>
\$6B35	TPCMD RESTORE <b>command: missing or invalid parameter</b>
\$6B36	<b>Restore execution error</b>
\$6B37	TPCMD RESTORE <b>command: Restore execution error</b>
\$6B38	TPCMD SAVE_REGISTER <b>command: Restore execution error</b>
\$6B39	<b>Store execution error</b>
\$6B3B	TPCMD READ_INFO <b>command: Information read error</b>
\$6B3C	TPCMD READ_REG <b>command: Register read error</b>
<b>ATMEL AT89xx AT8051 - Specific Errors</b>	
\$5E00	TCSETDEV <b>command: manufacturer not supported</b>
\$5E01	TCSETDEV <b>command: algorithm not found on card</b>
\$5E02	TCSETDEV <b>command: device not supported</b>
\$5E03	TCSETDEV <b>command: internal hardware configuration error</b>
\$5E04	TCSETPAR <b>command: invalid command</b>
\$5E05	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$5E06	TPSTART <b>command: command not preceded by TCSETPAR FCPU command</b>
\$5E07	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$5E08	TPSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$5E09	TPCMD <b>command: missing or invalid command</b>
\$5E10	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$5E11	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$5E12	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$5E13	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$5E14	TPCMD VERIFY <b>command: Eeprom memory verify error</b>
\$5E15	TPCMD VERIFY <b>command: Fuse Bits verify error</b>
\$5E16	TPCMD VERIFY <b>command: Lock Bits verify error</b>
\$5E17	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$5E18	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$5E19	TPCMD PROGRAM <b>command: source offset parameter out of range</b>

Error #	Description
\$5E1A	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$5E1B	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$5E1C	TPCMD PROGRAM <b>command: Fuse Bits program error</b>
\$5E1D	TPCMD PROGRAM <b>command: Lock Bits program error</b>
\$5E1E	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$5E1F	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$5E20	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$5E21	TPCMD BLANKCHECK <b>command: Eeprom memory blankcheck error</b>
\$5E22	TPCMD MASSERASE <b>command: Memory mass erase error</b>
\$5E23	TPCMD WRITECALIBRVALUE <b>command: missing or invalid parameter</b>
\$5E24	TPCMD WRITECALIBRVALUE <b>command: write calibr value error</b>
\$5E25	TPCMD RUN <b>command: execution error</b>
\$5E26	TPEND <b>command: execution error</b>
\$5E27	TPSTART <b>command: execution error</b>
\$5E28	TPCMD ENTRY <b>command: execution error</b>
\$5E30	TPCMD READ <b>command: missing or invalid parameter</b>
\$5E31	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$5E32	TPCMD READ <b>command: read memory error</b>
\$5E33	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$5E34	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$5E35	TCSETPAR RSTPOL <b>command: missing, invalid or out of range parameter</b>
\$5E36	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$5E37	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$5E38	TPCMD SET_JTCLK <b>command: missing or invalid parameter</b>
\$5E39	TCSETPAR CMODE <b>command: invalid protocol</b>
\$5E3A	TPSTART <b>command: command not preceded by TCSETPAR BAUDRATE command</b>
\$5E3B	TPCMD PROGRAM_SIG <b>command: signature program error</b>
\$5E3C	TPCMD VERIFY_SIG <b>command: signature verify error</b>
\$5E3D	TPCMD SIGNEARSE <b>command: Sign erase error</b>
\$5E3E	TPCMD MASSERASE <b>command: missing or invalid parameter</b>
<b>ATMEL AT32UCxx AVR32 - Specific Errors</b>	
\$7100	TCSETDEV <b>command: manufacturer not supported</b>
\$7101	TCSETDEV <b>command: algorithm not found on card</b>
\$7102	TCSETDEV <b>command: device not supported</b>
\$7103	TCSETDEV <b>command: internal hardware configuration error</b>
\$7104	TCSETPAR <b>command: invalid command</b>
\$7105	TCSETPAR FCPU <b>command: missing, invalid or out of range parameter</b>
\$7106	TPSTART <b>command: command not preceded by TCSETPAR FCPU command</b>

## Errors

Error #	Description
\$7107	TCSETPAR VDD <b>command: missing, invalid or out of range parameter</b>
\$7108	TFSTART <b>command: command not preceded by TCSETPAR VDD command</b>
\$7109	TPCMD <b>command: missing or invalid command</b>
\$7110	TPCMD VERIFY <b>command: missing or invalid parameter</b>
\$7111	TPCMD VERIFY <b>command: target start address parameter or length parameter out of range</b>
\$7112	TPCMD VERIFY <b>command: source offset parameter out of range</b>
\$7113	TPCMD VERIFY <b>command: Flash memory verify error</b>
\$7114	TPCMD VERIFY <b>command: Eeprom memory verify error</b>
\$7115	TPCMD VERIFY <b>command: Fuse Bits verify error</b>
\$7116	TPCMD VERIFY <b>command: Lock Bits verify error</b>
\$7117	TPCMD PROGRAM <b>command: missing or invalid parameter</b>
\$7118	TPCMD PROGRAM <b>command: target start address parameter or length parameter out of range</b>
\$7119	TPCMD PROGRAM <b>command: source offset parameter out of range</b>
\$711A	TPCMD PROGRAM <b>command: Flash memory program error</b>
\$711B	TPCMD PROGRAM <b>command: Eeprom memory program error</b>
\$711C	TPCMD PROGRAM <b>command: Fuse Bits program error</b>
\$711D	TPCMD PROGRAM <b>command: Lock Bits program error</b>
\$711E	TPCMD BLANKCHECK <b>command: missing or invalid parameter</b>
\$711F	TPCMD BLANKCHECK <b>command: target start address parameter or length parameter out of range</b>
\$7120	TPCMD BLANKCHECK <b>command: Flash memory blankcheck error</b>
\$7121	TPCMD BLANKCHECK <b>command: Eeprom memory blankcheck error</b>
\$7122	TPCMD MASERASE <b>command: Memory mass erase error</b>
\$7123	TPCMD WRITECALIBRVALUE <b>command: missing or invalid parameter</b>
\$7124	TPCMD WRITECALIBRVALUE <b>command: write calibr value error</b>
\$7125	TPCMD RUN <b>command: execution error</b>
\$7126	TFEND <b>command: execution error</b>
\$7127	TFSTART <b>command: execution error</b>
\$7128	TPCMD ENTRY <b>command: execution error</b>
\$7129	TPCMD MASERASE <b>command: missing or invalid parameter</b>
\$7130	TPCMD READ <b>command: missing or invalid parameter</b>
\$7131	TPCMD READ <b>command: target start address parameter or length parameter out of range</b>
\$7132	TPCMD READ <b>command: read memory error</b>
\$7133	TCSETPAR RSTDOWN <b>command: missing, invalid or out of range parameter</b>
\$7134	TCSETPAR RSTUP <b>command: missing, invalid or out of range parameter</b>
\$7135	TCSETPAR RSTPOL <b>command: missing, invalid or out of range parameter</b>
\$7136	TCSETPAR PWUP <b>command: missing, invalid or out of range parameter</b>
\$7137	TCSETPAR CMODE <b>command: missing, invalid or out of range parameter</b>
\$7138	TPCMD SET_JTCLK <b>command: missing or invalid parameter</b>



Error #	Description
\$7139	TCSETPAR CMODE <b>command: invalid protocol</b>
\$713A	TPSTART <b>command: command not preceded by</b> TCSETPAR BAUDRATE <b>command</b>
\$713B	TPCMD SET_DAISS_CHAIN <b>command: missing or invalid parameter</b>
\$713C	TPCMD SET_SECURITY <b>command: missing or invalid parameter</b>

# 12 Approved SD Cards

FlashRunner has been successfully tested with the SD cards listed in the table below, which are therefore recommended.

Table 106: Recommended SD Cards

Manufacturer	Model	Size
Kingston Technology	1GB Elite Pro	1GB
Kingston Technology	2GB Elite Pro	2GB
Transcend	SD Card	2GB
Transcend	SD Card	2GB
Transcend	High-Speed SD Card	512MB
Transcend	High-Speed SD Card	1GB
Transcend	SD Card	1GB
Memorex	SD Travelcard	1GB
Memorex	SD Travelcard	512MB
SwissBit	S-200 Series	2GB
SwissBit	S-300u Series	2GB

FlashRunner also supports SD HC cards via a small setup process, which will reduce the SD card partition up to 2 GB regardless to SD card size.

To setup the SD HC cards:

Install a partition manager software, like "Partition Master" from EaseUS (<http://www.easeus.com/>)

Delete the main SD card partition.

Create a new 2GB partition.

Format it with FAT16 file system technology.

Table 107: Recommended SD Cards

Manufacturer	Model	Size
Transcend	4Gb 10 microsd HC	4GB
Kingston	4GB MicroSD HC SDC10/4GB N0352-003.A00LF	4GB
Swissbit	4GB microsd HC	4GB