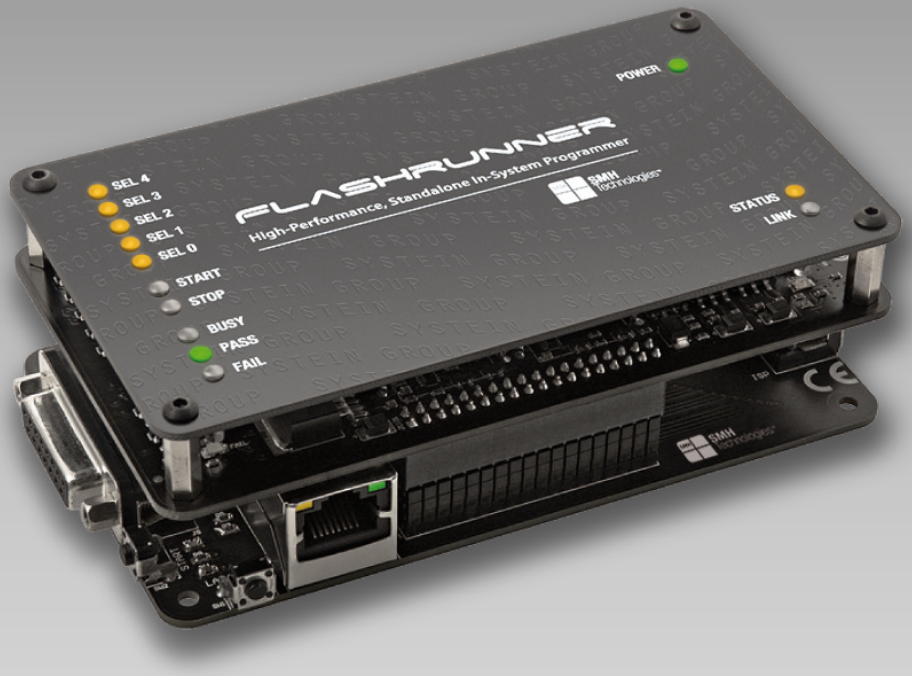
FLASHRUNNER

High-Performance, Standalone In-System Programmer

FR01LAN

User's Manual



SMH Technologies™

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

SMH Technologies™

FlashRunner FR01LAN

High-Performance, Standalone, In-System Programmer

User's Manual

DC 10729

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.

CE

Our policy at SMH Technologies is to comply with all applicable worldwide safety and EMC/EMI regulations.
This product is certified to comply with the 2004/108/EC Directives and is in conformity with the EN6100-6-2 and the EN61000-6-3 standards.
In a domestic environment, this product may cause radio interference in which case the user may be required to take adequate prevention measures.
Attaching additional wiring to this product or modifying the product operation from the factory default as shipped may effect its performance and cause interference with other apparatus in the immediate vicinity. If such interference is detected, suitable mitigating measures should be taken.

Disposal of Waste Electrical & Electronic Equipment (WEEE).
In the European Union, this label indicates that this product should not be disposed of with household waste. It must be deposited in an appropriate facility to allow for recovery and recycling. For more detailed information about the recycling of this product, please contact your local city office, household waste disposal service or the retail store where you purchased this product.

SMH Technologies

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

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

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

Contents

Index of Figures

Index of Tables

0 Before Starting

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

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 Safety

FlashRunner is a low-voltage device. However, when integrating it inside an automatic test equipment or when interfacing it with other systems, take all precautions in order to avoid electrical shocks due to, for example, different ground references.

Make all connections to the target system before applying power to the instrument.

To protect FlashRunner against electrostatic discharge (ESD), always connect yourself to ground (e.g. via wrist straps) when handling the instrument.

Always store FlashRunner inside an antistatic bag when not in use.

0.3 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*

0.4 Additional Documentation

This user's manual provides information about how to setup FlashRunner FR01LAN and its hardware characteristics.

For information about FlashRunner commands and their syntax, including specific commands for specific family of microcontrollers, please refer to the FlashRunner Programmer's Manual, included (in PDF format) in the FlashRunner CD-ROM.

1 Overview

1.1 What is FlashRunner FR01LAN?

FlashRunner FR01LAN is a member of the FlashRunner series of a high-performance, standalone In-System Programmers specific for Flash-based microcontrollers and serial memories.

FlashRunner FR01LAN is targeted at production environments, easily interfaces to your programming system or Automatic Test Equipment (ATE) and can work either in full standalone mode or controlled by a host system.

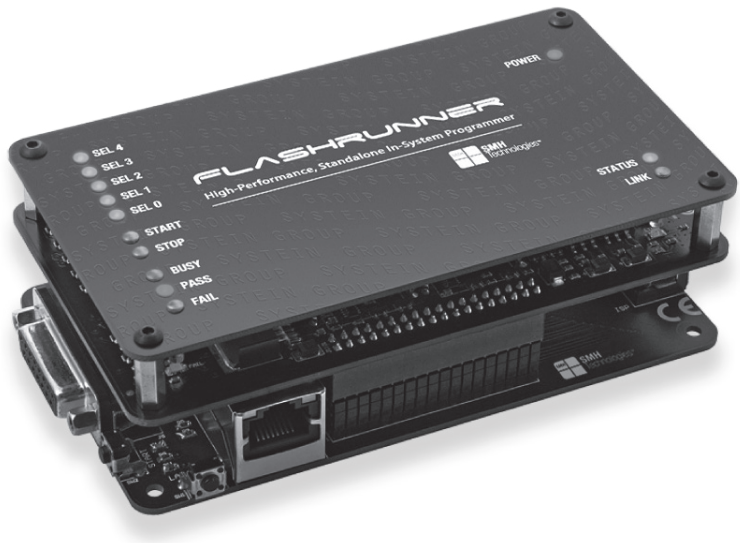


Figure 1.1: FlashRunner FR01LAN

1.1.1 General features

- Fastest programming algorithms (as fast as target device's memory technology limit), approved by silicon manufacturers;
- Easy ATE integration;

- Standalone operations (projects and code images stored on a memory card);
- Also controllable by any host system via RS-232 or Ethernet;
- Supports most ISP protocols (BDM, JTAG, SPI, I2C, MON, ICC, SCI, etc.);
- Flexible, fully configurable;
- Compact and robust design for production environments;
- Data integrity guaranteed (every data transfer to/from the host system or Secure Digital card is CRC tagged).

1.1.2 Hardware features

- 9 to 24V power supply input;
- Five digital I/O lines;
- Two digital I/O or analog output lines;
- Two programmable output voltages (0 to 15V, 0.25A and 0 to 5V, 0.5A);
- One analog input line;
- One programmable clock output;
- Secure Digital memory card (up to 2 GB);
- 512 bytes on-board dynamic memory;
- On-board timekeeper and calendar;
- I/O protection;
- Optoisolated inputs for project selection;
- Two optoisolated command inputs (START and STOP);
- Three optoisolated status outputs (FAIL, PASS, BUSY);
- One optoisolated RS-232 channel;
- One optoisolated Ethernet channel.

1.1.3 Software features

- Fully autonomous standalone mode thanks to its SD memory card (FAT16);
- Controllable by any host system through a terminal utility and simple ASCII protocol;

- Up to 32 hardware-selectable projects (scripts), unlimited software-selectable projects;
- Log files;
- Erase, blank check, program, read, verify, oscillator trimming, etc.

1.2 Package Checklist

The FlashRunner FR01LAN package includes the following items:

- FlashRunner FR01LAN unit, including an SD card already pre-installed with the programming algorithm(s) you specified at the time of purchase;
- An Ethernet cross cable;
- A serial cable;
- FlashRunner “System Software” CD-ROM, containing the FlashRunner Control Panel utility and the FlashRunner Programmer’s Manual in PDF format;
- This user’s manual;
- A registration card.

1.3 Hardware Overview

FlashRunner FR01LAN is composed of three layers. From bottom to top:

- **Connection Layer.** Provides D-Sub connectors to interface to your programming/testing system. Includes optoisolation circuitry and a LAN and RS-232 connectors to interface to a host system.
- **Programming Engine Layer.** Contains the FlashRunner programming engine, the core of the instrument.
- **Cover Layer.** The cover layer has the function of protecting the underlying layers and replicating the programming engine’s status LEDs. If space is an issue when integrating FlashRunner in your programming/testing system, the cover layer can be easily removed.

1

The figures below illustrate the various FlashRunner FR01LAN layers.

Figure 1.2: FR01LAN Top Layer

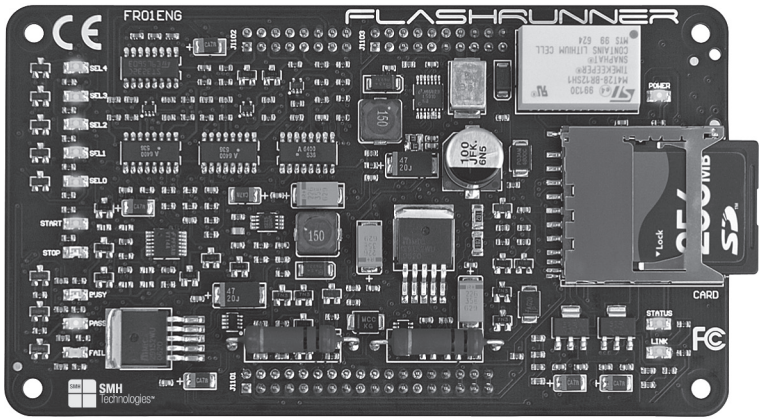


Figure 1.3: FR01LAN Programming Engine Layer

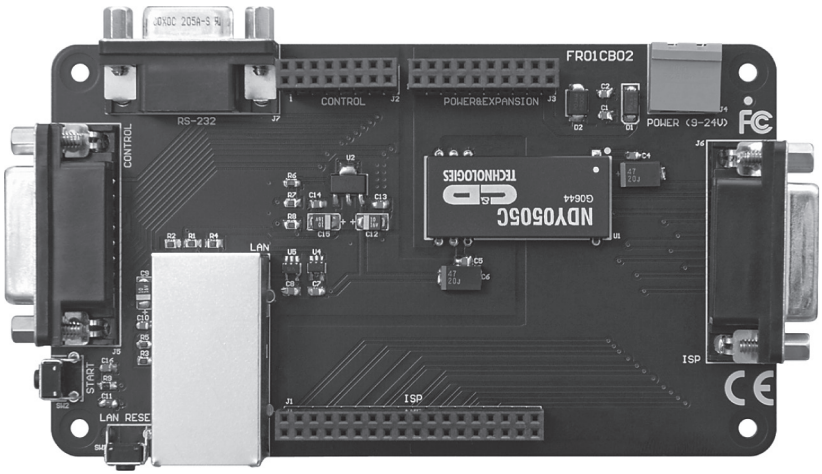


Figure 1.4: FR01LAN Connection Layer

The Connection layer includes target connectors, optoisolated LAN and RS-232 ports, a power connector, and a “START” push-button.

1.3.1 Power Supply

FlashRunner FR01LAN is powered through a 9-24V DC terminal block connector.

1.3.2 LAN Connector

The LAN connector is used for communication with the host PC system. Use the provided Ethernet cross cable to connect FlashRunner with your PC.

1.3.3 RS-232 Connector

Alternatively, communication with the host PC can be done with the RS-232 connector. Use the provided serial cable to connect FlashRunner with your PC.

1.3.4 Target Connectors

The “ISP” and “CONTROL” D-Sub connectors are used to interface to an external target system/programming equipment.

Note: *for the pinout of the “ISP” and “CONTROL” connector, see “Connectors” on page 29.*

1.3.5 Start Push-Button

The “START” push-button is directly connected to the FlashRunner START line in the “CONTROL” D-Sub connector.

1.3.6 Optoisolation

All signals in the “CONTROL”, “LAN” and “RS-232” connectors are optoisolated. See “Connectors” on page 29 for more information.

1.4 Programming Algorithms and Licenses

FlashRunner FR01LAN 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 FR01LAN 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 (see the FlashRunner Programmer’s Manual for more information).

1.4.1 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 that will let you use the programming algorithm. A license file enables the use of a specific programming algorithm 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 the FlashRunner Programmer's Manual).

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 **`FSSENDFILE`** command (for more information on FlashRunner commands, see the FlashRunner Programmer's Manual).

Alternatively, you can use the FlashRunner Control Panel utility to install new programming algorithms and licenses. For more information on the FlashRunner Control Panel please refer to the FlashRunner Programmer's Manual.

1.5 Upgrading the Firmware

The FlashRunner firmware can be easily upgraded using the provided Control Panel utility. For more information, please refer to the FlashRunner Programmer's Manual.

1

2 System Setup

2.1 Overview

Note: *the example shows how to set up the system for programming a Freescale MC68HC908QY4 microcontroller. For how to connect to other target devices, please refer to the FlashRunner Programmer's Manual.*

This chapter will explain how to set up FlashRunner FR01LAN for the first time. Although FlashRunner is typically used for standalone operations (Standalone mode), the examples in this chapter will use the host system to send commands to FlashRunner (Host mode).

When moving FlashRunner to the production environment, you can take full advantage of the instrument's SD card to make the instrument work without being controlled by the host system.

For more information about Standalone mode and Host mode, see the FlashRunner Programmer's Manual.

2.2 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;
- 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.*

2.3 Hardware Setup

To set up FlashRunner FR01LAN, you must follow the steps below, in the indicated order:

1. Interface FlashRunner with your test/programming equipment;
2. Connect FlashRunner to the host PC system;
3. Power up FlashRunner;
4. Set up LAN settings (if you use the Ethernet connection);
5. Send FlashRunner commands via the FlashRunner Control Panel utility.

2.3.1 Interfacing FlashRunner with your Test/Programming Equipment

Build an ISP cable to connect from the FlashRunner's 15-way, D-Sub "ISP" connector (located in the Connection layer) to your target board. Make all the required connections (power, oscillator, ISP signals) to the target microcontroller, by wiring the required lines from the "ISP" connector to your target microcontroller.

Typical connections for all the device families supported by FlashRunner are shown in the FlashRunner Programmer's Manual.

2.3.2 Connecting FlashRunner to the Host PC System

You can connect FlashRunner to the host system through either the RS-232 or LAN port. Both the serial and LAN connectors are located in the Connection layer.

FlashRunner FR01LAN comes with a serial cable and an Ethernet cross cable to connect directly to a host PC.

2.3.3 Powering Up FlashRunner

Power up FlashRunner by connecting the output of a power supply to the terminal block connector located in the Connection layer. FlashRunner accepts any DC voltage between 9V and 24V.

2.3.4 Setting Up LAN Settings

If you connected FlashRunner to the host PC using the Ethernet connection, you need to set up the FlashRunner IP address. For learning how to set up the FlashRunner IP address, please refer to the FlashRunner Programmer's Manual.

2.4 Step-by-Step Tutorial: Sending Commands to FlashRunner

After setting up the hardware, you are ready to send commands to the instrument. The following steps will guide you through the process of launching your first FlashRunner commands using the provided FlashRunner Control Panel utility. For detailed information about the FlashRunner Control Panel utility, see the FlashRunner Programmer's Manual.

Note: *the following steps show how to program a Freescale MC68HC908QY4 microcontroller, and the details are therefore specific for that microcontroller. However, the procedures shown are general and will allow you get a feel of how FlashRunner works.*

1. Launch the FlashRunner Control Panel utility. Select **Start > Programs > SMH Technologies > FlashRunner > Control Panel**. The Control Panel utility will open.
2. 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
   - **"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, please refer to the FlashRunner Programmer's Manual.

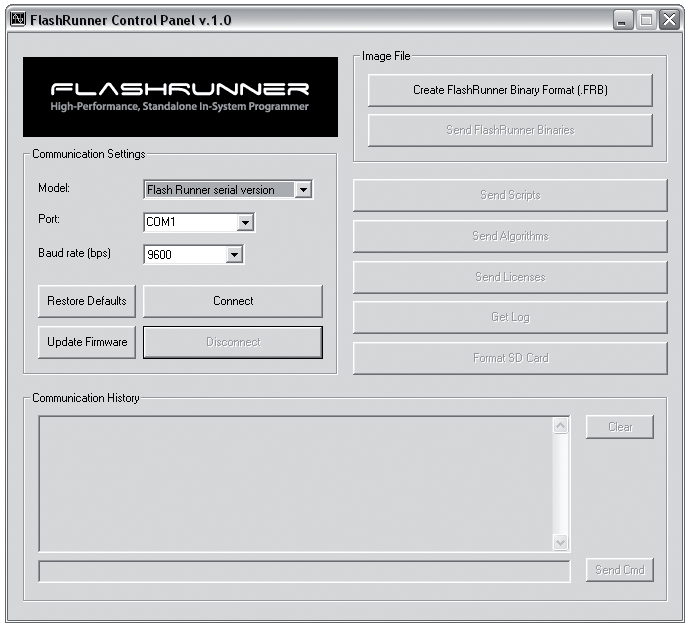


Figure 2.1: FlashRunner Control Panel, Communication Settings

3. Click the **"Connect"** button. On the **"Communication History"** section, note the commands that have been sent and received. In this case, the **SPING** command is automatically sent to FlashRunner, which replies with the **PONG>** string.
4. In the edit box below the communication history, type the following commands (each followed by Return):

```
TCSETDEV FREESCALE MC68HC908QY4 HC08
TCSETPAR FOSC 16000000
TCSETPAR FDIV 4
TCSETPAR VDD 5000
```

These commands set, respectively, the target microcontroller, the oscillator frequency, the internal divisor and the VDD voltage. In this example, we used a 16 MHz oscillator, the internal divisor for MC68HC908QY4 devices is fixed to 4, and the VDD is 5 V.

FlashRunner will respond to each command with the > string, indicating that the command has been successfully executed. After sending these commands, the Control Panel will look like the figure below.

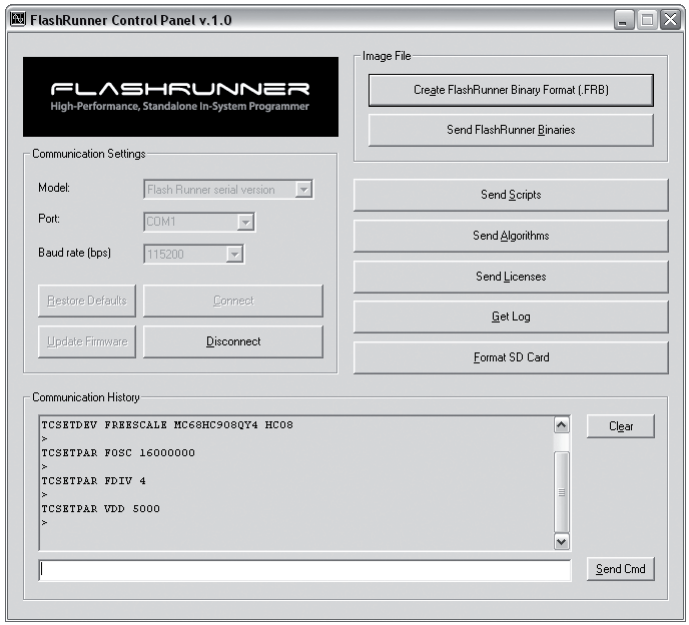


Figure 2.2: FlashRunner Control Panel, Target Device Configured

5. When working with Freescale HC08 devices, FlashRunner requires you to specify the power up and power down times, in milliseconds. Send the following two commands:

```
TCSETPAR PWDOWN 10
TCSETPAR PWUP 10
```

6. After specifying the target device settings, we are ready to transfer to FlashRunner the binary image to be programmed into the target device. 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.

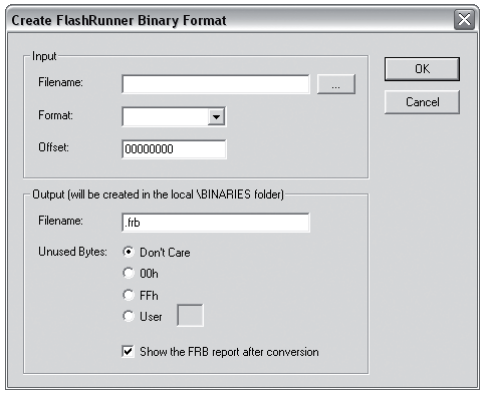


Figure 2.3: FlashRunner Control Panel, Binary File Conversion

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.

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

7. To transfer the created image to FlashRunner, send the following command:

```
TPSENDFILE YMODEM DEMO.FRB
```

In this example, the image file is called **`DEMO.FRB`**. The following dialog box will appear.

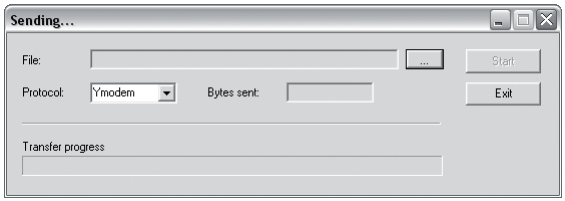


Figure 2.4: FlashRunner Control Panel, File Transfer

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

8. We are now ready to start the actual programming part. Send the following commands:

```
TPSETSRC FILE DEMO.FRB
TPSTART
TPCMD SETPWD CONST $FF $FF $FF $FF $FF $FF $FF $FF
TPCMD MASSERASE F
TPCMD BLANKCHECK F $EE00 4608
TPCMD PROGRAM F $EE00 $EE00 4608
TPCMD VERIFY F S $EE00 $EE00 4608
TPEND
```

The data to be programmed is taken from the image file starting at $EE00 (offset from the beginning of the file), is programmed to the target microcontroller starting from the location $EE00 and is 4608 bytes long.

The **`TPSETSRC`** command specifies the source file for the **`TPCMD PROGRAM`** e **`TPCMD VERIFY`** commands that come next. All the actual programming operations are sent between a **`TPSTART`** and **`TPEND`** command. The **`TPCMD SETPWD`** command sets the security bytes needed to perform subsequent operations.

After sending these commands, the Control Panel will look like the figure below.

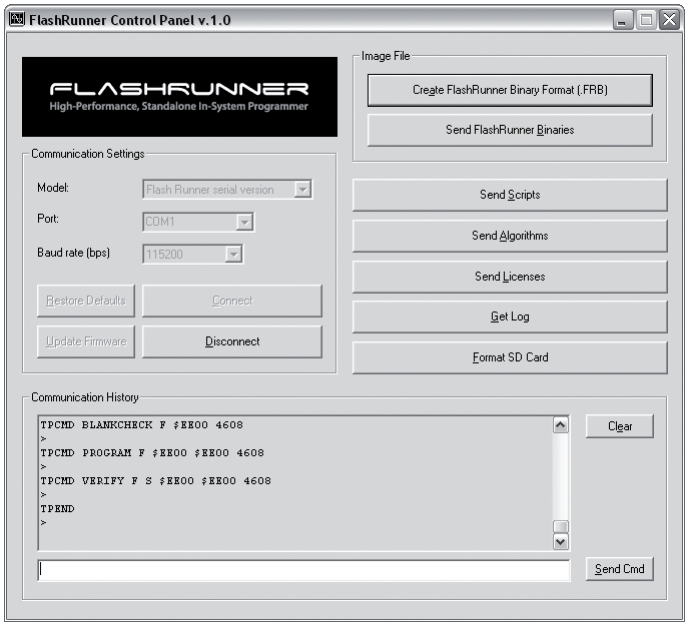


Figure 2.5: FlashRunner Control Panel, Target Device Programmed

9. We are now done with programming the target device. Click the **"Disconnect"** button to free the serial port resource.

For detailed information on all of the FlashRunner commands and their syntax, including specific commands for specific family of microcontrollers, please refer to the FlashRunner Programmer's Manual, included (in PDF format) in the FlashRunner CD-ROM.

Programming can be automated by creating "scripts". Scripts are text files, stored in the SD card, which contain a sequence of FlashRunner commands. See the FlashRunner Programmer's Manual for more information about scripts.

3 Connectors

3.1 Overview

FlashRunner FR01LAN connects to your programming/testing system through two D-Sub connectors, the “ISP” connector and the “CONTROL” connector.

3.2 ISP Connector

The “ISP” D-Sub connector groups all of the signals needed to program the target device. This connector has several input/output lines, both digital and analog, that are automatically configured by FlashRunner depending on the specific target device to be programmed (see the FlashRunner Programmer’s Manual to learn how to connect these lines to your specific target device).

Note: *ISP signals are not optoisolated and are referenced to GND (the power supply ground).*

Additionally, in order to avoid undesired current loops between the FlashRunner power supply and the target board, a power supply with a floating output (ground not referenced to the earth potential) should be used.

Note: *when FlashRunner is powered off, ISP signals are not HiZ. As a result, ISP signals have a low impedance that could be intrusive when other tools perform in-circuit testing on the target board.*

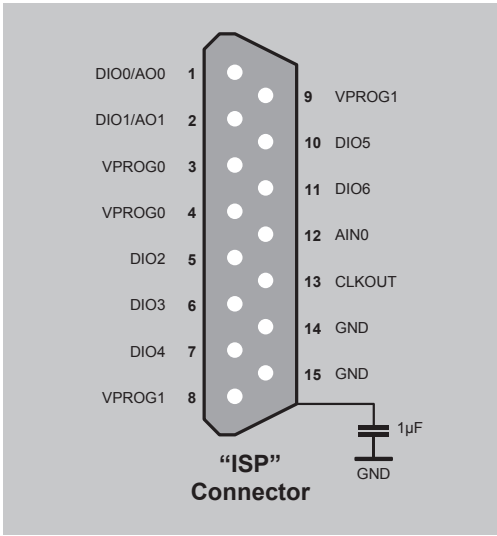


Figure 3.1: ISP Connector

Table 3.1: ISP Connector Signals

Pin #	Signal Name	Description
1	DIO0/AO0	Digital input/output 0 or analog output 0
2	DIO1/AO1	Digital input/output 1 or analog output 1
3	VPROG0	Programmable voltage 0 (max 5.5V, 500mA)
4	VPROG0	Programmable voltage 0 (max 5.5V, 500mA)
5	DIO2	Digital input/output 2
6	DIO3	Digital input/output 3
7	DIO4	Digital input/output 4
8	VPROG1	Programmable voltage 1 (max 14.5V, 250mA)
9	VPROG1	Programmable voltage 1 (max 14.5V, 250mA)
10	DIO5	Digital input/output 5
11	DIO6	Digital input/output 6
12	AIN0	Analog input 0 (max 28.5V)
13	CLKOUT	Clock output
14	GND	Ground
15	GND	Ground

3.3 Control Connector

The “CONTROL” D-Sub connector is used to communicate with the host system and for integration with an automatic programming/testing equipment.

Note: *all control signals are optoisolated and are referenced to OPTO_GND.*

This allows a host system to safely communicate with FlashRunner FR01LAN even when the target board has a different ground reference than the host system’s (and it’s not possible to connect them together).

Additionally, in order to avoid undesired current loops between the FlashRunner power supply and the target board, a power supply with a floating output (ground not referenced to the earth potential) should be used.

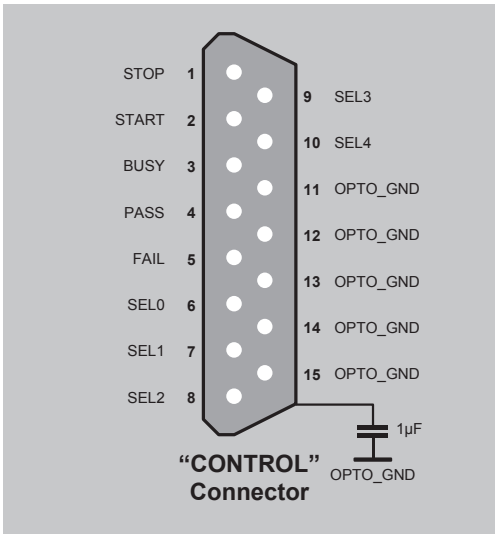


Figure 3.2: Control Connector

Table 3.2: Control Connector Signals

Pin #	Signal Name	Description
1	STOP	STOP (input , optoisolated, active low)
2	START	START (input , optoisolated, active low)
3	BUSY	BUSY (output, open-drain, optoisolated, active low)
4	PASS	PASS (output, open-drain, optoisolated, active low)
5	FAIL	FAIL (output, open-drain, optoisolated, active low)
6	SEL0	Script selection 0 (input, optoisolated)
7	SEL1	Script selection 1 (input, optoisolated)
8	SEL2	Script selection 2 (input, optoisolated)
9	SEL3	Script selection 3 (input, optoisolated)
10	SEL4	Script selection 4 (input, optoisolated)
11	OPTO_GND	Optoisolation ground
12	OPTO_GND	Optoisolation ground
13	OPTO_GND	Optoisolation ground
14	OPTO_GND	Optoisolation ground
15	OPTO_GND	Optoisolation ground

3.3.1 RS-232 Connector

The “RS-232” D-Sub connector can be used to communicate with a host system.

Note: *the RS-232 signals are optoisolated.*

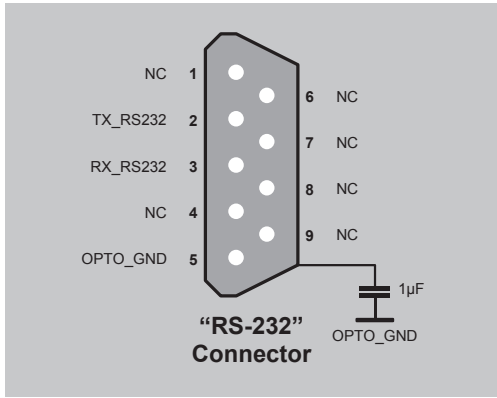


Figure 3.3: RS-232 Connector

Table 3.3: RS-232 Connector Signals

Pin #	Signal Name	Description
1	NC	Not connected
2	TX_RS232	TX (output, optoisolated, RS-232 levels)
3	RX_RS232	RX (input, optoisolated, RS-232 levels)
4	NC	Not connected
5	OPTO_GND	Optoisolation ground
6	NC	Not connected
7	NC	Not connected
8	NC	Not connected
9	NC	Not connected

4 Technical Specifications

4.1 Absolute Maximum Ratings

Table 4.1: Absolute Maximum Ratings

Parameter	Value
"CONTROL" Connector (all signals are referenced to OPTO_GND)	
Maximum input voltage on lines START, STOP, SEL[4..0]	-2V to +9V
Maximum current on lines BUSY, PASS,FAIL	-50mA to 1.5mA
"RS-232" Connector (all signals are referenced to OPTO_GND)	
Maximum input voltage on line RX_RS232	-25V to +25V
Maximum current on line TX_RS232	±60mA
"ISP" Connector (all signals are referenced to GND)	
Maximum input voltage on lines DIO/AO[1..0], DIO[6..2], CLKOUT	-1V to +7V
Maximum input voltage on line AIN0	-12V to +40V
Maximum current on lines DIO/AO[1..0], DIO[6..2], CLKOUT	±50mA
Maximum current on line VPROG0	500mA
Maximum current on line VPROG1	250mA
"POWER" Connector	
Maximum supply voltage on line POWER (reference GND)	-20V to +30V

4.2 DC Characteristics and Functional Operating Range

Table 4.2: DC Characteristics and Functional Operating Range

Parameter	Condition	Value		
		Min	Typ	Max
"CONTROL" Connector				
V_{IL} (input low voltage) on lines START, STOP, SEL[4..0]	The driver must be able to provide at least 5mA	0V	-	2V
V_{IH} (input high voltage) on lines START, STOP, SEL[4..0]		3V	-	5V
V_{OL} (output low voltage) on lines BUSY, FAIL, PASS	I_{OL} = 4.5mA	-	-	450mV
V_{OH} (output high voltage) on lines BUSY, FAIL, PASS		4.5V	-	5V
V_{OL} (output low voltage) on line TX_RS232	R_{LOAD} = 3KΩ	-	-	-5V
V_{OH} (output high voltage) on line TX_RS232	R_{LOAD} = 3KΩ	+5V	-	-
"RS-232" Connector				
V_{IL} (input low voltage) on line RX_RS232		-	-	1.2V
V_{IH} (input high voltage) on line RX_RS232		2.4V	-	-
"ISP" Connector				
V_{IL} (input low voltage) on lines DIO[6..2], DIO[1..0]	Configured as digital lines	-	-	$0.3V_{PROG0}$
V_{IH} (input high voltage) on lines DIO[6..2], DIO[1..0]	Configured as digital lines	$0.7V_{PROG0}$	-	V_{PROG0}
V_{OL} (output low voltage) on lines DIO[6..2], DIO[1..0], CLKOUT	Configured as digital lines, V_{PROG0} = 3V, I_{OL} = 12mA	-	-	0.36V
V_{OH} (output high voltage) on lines DIO[6..2], DIO[1..0], CLKOUT	Configured as digital lines, V_{PROG0} = 3V, I_{OH} = 12mA	2.56V	-	-
V_{OL} (output low voltage) on lines DIO[6..2], DIO[1..0], CLKOUT	Configured as digital lines, V_{PROG0} = 5.5V, I_{OL} = 24mA	-	-	0.36V
V_{OH} (output high voltage) on lines DIO[6..2], DIO[1..0], CLKOUT	Configured as digital lines, V_{PROG0} = 5.5V, I_{OH} = 24mA	4.86V	-	-
I_{OH} current (source) on lines DIO[6..2], DIO[1..0]	Configured as input with active pull-ups	-	3.4mA	-
DIO/AO[1..0] voltage	Configured as analog output	3V	-	14.5V
DIO/AO[1..0] IO current (sink and source)	Configured as analog output	-	-	±40mA
I_{OH} current (source) on lines DIO/AO[1..0]	Configured as analog lines with active pull-ups	-	5.5mA	-
I_L (input leakage current) on line AIN0	V_{AIN0} = 25V	-	-	4.3mA
AIN0 line input voltage		0V	-	28.5V
VPROG0 line output voltage		1.6V	-	5.5V
VPROG0 current (source)		-	-	500mA
VPROG1 line output voltage		3V	-	14.5V

Parameter	Condition	Value		
		Min	Typ	Max
VPROG1 current (source)		-	-	250mA
"POWER" Connector				
Supply voltage		9V	-	24V
Power consumption		-	-	1.5A

4.3 AC Characteristics

Table 4.3: AC Characteristics

Parameter	Condition		Value		
			Min	Typ	Max
"ISP" Connector					
t_{RISE} on lines DIO[6..2], DIO[1..0], CLKOUT when configured as digital output push-pull	V_{PROG0} = 1.8V	Load: 470Ω//100pF (see figure 4.1a)	-	40ns	-
	V_{PROG0} = 3.3V		-	30ns	-
	V_{PROG0} = 5V		-	25ns	-
t_{FALL} on lines DIO[6..2], DIO[1..0], CLKOUT when configured as digital output push-pull	V_{PROG0} = 1.8V	Load: 470Ω//100pF (see figure 4.1a)	-	35ns	-
	V_{PROG0} = 3.3V		-	25ns	-
	V_{PROG0} = 5V		-	25ns	-
t_{RISE} on lines DIO/AO[1..0] configured as analog output	V_{PROG1} = 3V	Load: 4.7KΩ//100pF (see figure 4.1a)	-	7µs	-
	V_{PROG1} = 12V		-	11µs	-
	V_{PROG1} = 14.5V		-	12µs	-
t_{FALL} on lines DIO/AO[1..0] configured as analog output	V_{PROG1} = 3V	Load: 100pF (see figure 4.1b)	-	8µs	-
	V_{PROG1} = 12V		-	20µs	-
	V_{PROG1} = 14.5V		-	30µs	-
t_{RISE} on line VPROG0	V_{PROG0} = 0-1.8V	Load: 15Ω//10mF (see figure 4.1a)	-	10ms	-
	V_{PROG0} = 0-3.3V	Load: 22Ω//10mF (see figure 4.1a)	-	15ms	-
	V_{PROG0} = 0-5.5V	Load: 22Ω//10mF (see figure 4.1a)	-	20ms	-
t_{FALL} on line VPROG0	V_{PROG0} = 1.8-0V	Load: 10mF (see figure 4.1b)	-	300ms	-
	V_{PROG0} = 3.3-0V		-	350ms	-
	V_{PROG0} = 5.5-0V		-	350ms	-
t_{RISE} on line VPROG1	V_{PROG1} = 0-3V	Load: 10Ω//1mF (see figure 4.1a)	-	1.3ms	-
	V_{PROG1} = 0-5V	Load: 47Ω//1mF (see figure 4.1a)	-	1.8ms	-
	V_{PROG1} = 0-14.5V	Load: 94Ω//1mF (see figure 4.1a)	-	13ms	-
t_{FALL} on line VPROG1	V_{PROG1} = 3-0V	Load: 1mF (see figure 4.1b)	-	18ms	-
	V_{PROG1} = 5-0V		-	30ms	-
	V_{PROG1} = 14.5-0V		-	45ms	-
CLKOUT frequency			0MHz	-	50MHz

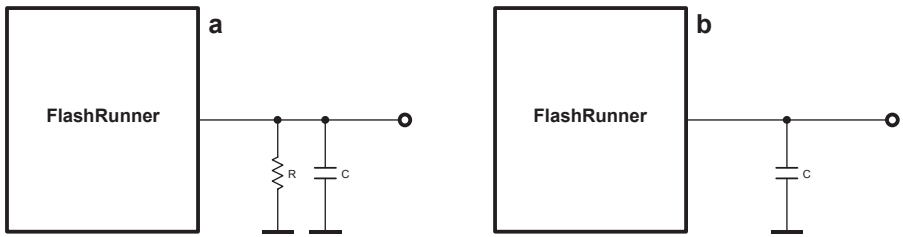


Figure 4.1: Load Conditions

4.4 Physical and Environmental Specifications

Table 4.4: Physical and Environmental Specifications

Parameter	Value
Dimensions (with top panel)	130 x 74 x 42 mm
Dimensions (without top panel)	130 x 74 x 27 mm
Weight (with top panel)	290 g
Weight (without top panel)	250 g
“ISP” connector type	15-pin D-Sub female
“CONTROL” connector type	15-pin D-Sub female
“RS-232” connector type	9-pin D-Sub female
“LAN” connector type	RJ-45 connector
“POWER” connector type	Terminal block connector, pitch = 2.54 mm
Operating temperature	0-50°C
Operating humidity	90% max (without condensation)
Storage temperature	0-70°C
Storage humidity	90% max (without condensation)

SMH Technologies S.r.l.
Via Giovanni Agnelli, 1
33083 Villotta di Chions (PN) Italy
info@smh-tech.com
www.smh-tech.com