SMH Technologies®

29/09/2023

Driver v. 5.02

Moreno Ortolan

Interfacing FlashRunner 2.0 with NXP SJA11

SMH Technologies S.r.l.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

SMH Technologies®

NXP SJA11 Introduction

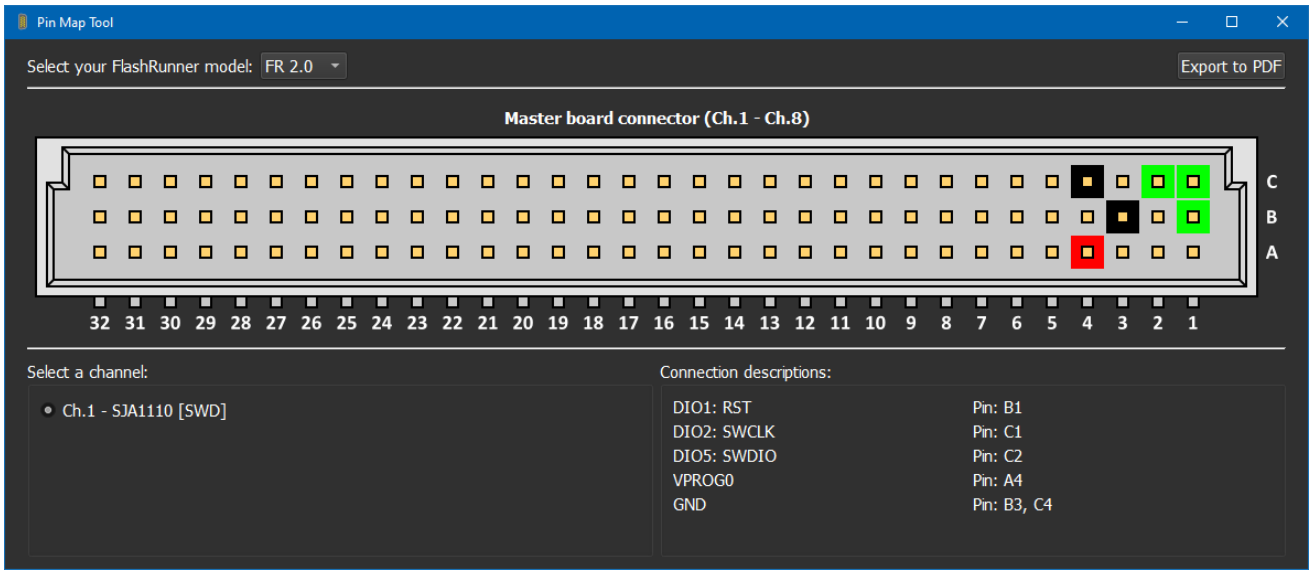
SJA1110 is an NXP Automotive Gigabit Ethernet Switch.

NXP SJA11 Protocol and PIN map

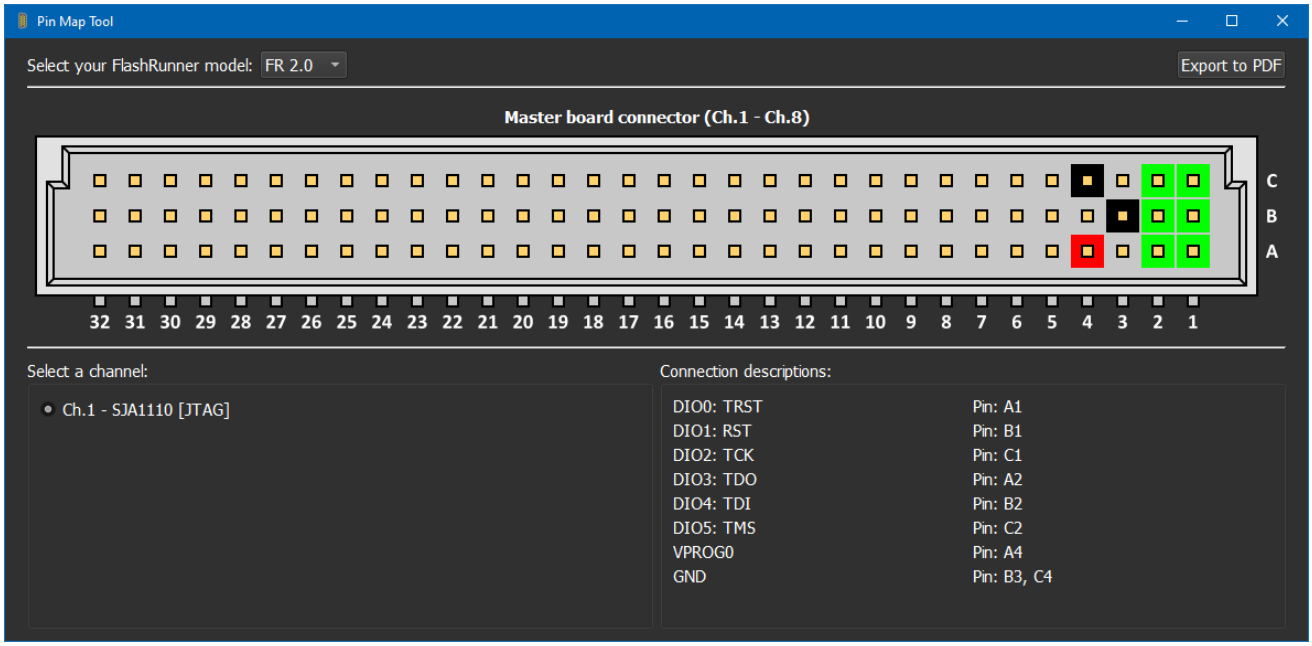
SJA11 devices support the SWD and JTAG protocol.

```
#TCSETPAR CMODE <SWD/JATG>
```

NXP SJA11 PIN MAP SWD



NXP SJA11 PIN MAP JTAG



HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

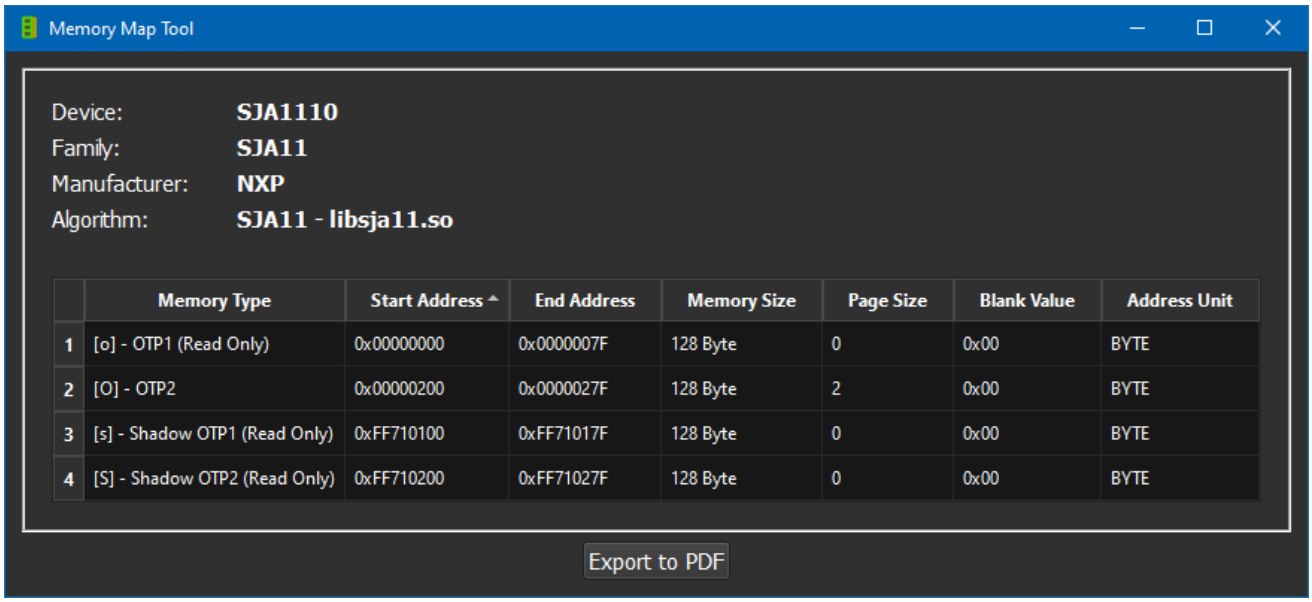
D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

SMH Technologies®

NXP SJA11 Memory Map

Memory Type	Start Address	End Address	Memory Size	Page Size	Blank Value	Address Unit
[o] - OTP1 (Read Only)	0x00000000	0x0000007F	128 Byte	0	0x00	BYTE
[O] - OTP2	0x00000200	0x0000027F	128 Byte	2	0x00	BYTE
[s] - Shadow OTP1 (Read Only)	0xFF710100	0xFF71017F	128 Byte	0	0x00	BYTE
[S] - Shadow OTP2 (Read Only)	0xFF710200	0xFF71027F	128 Byte	0	0x00	BYTE



	Memory Type	Start Address	End Address	Memory Size	Page Size	Blank Value	Address Unit
1	[o] - OTP1 (Read Only)	0x00000000	0x0000007F	128 Byte	0	0x00	BYTE
2	[O] - OTP2	0x00000200	0x0000027F	128 Byte	2	0x00	BYTE
3	[s] - Shadow OTP1 (Read Only)	0xFF710100	0xFF71017F	128 Byte	0	0x00	BYTE
4	[S] - Shadow OTP2 (Read Only)	0xFF710200	0xFF71027F	128 Byte	0	0x00	BYTE

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

NXP SJA11 One Time Programmable Bits

This is an OTP area (One-Time-Programmable) and this means that once a bit is blown, so it has the '1' state, it cannot return to the '0' state.

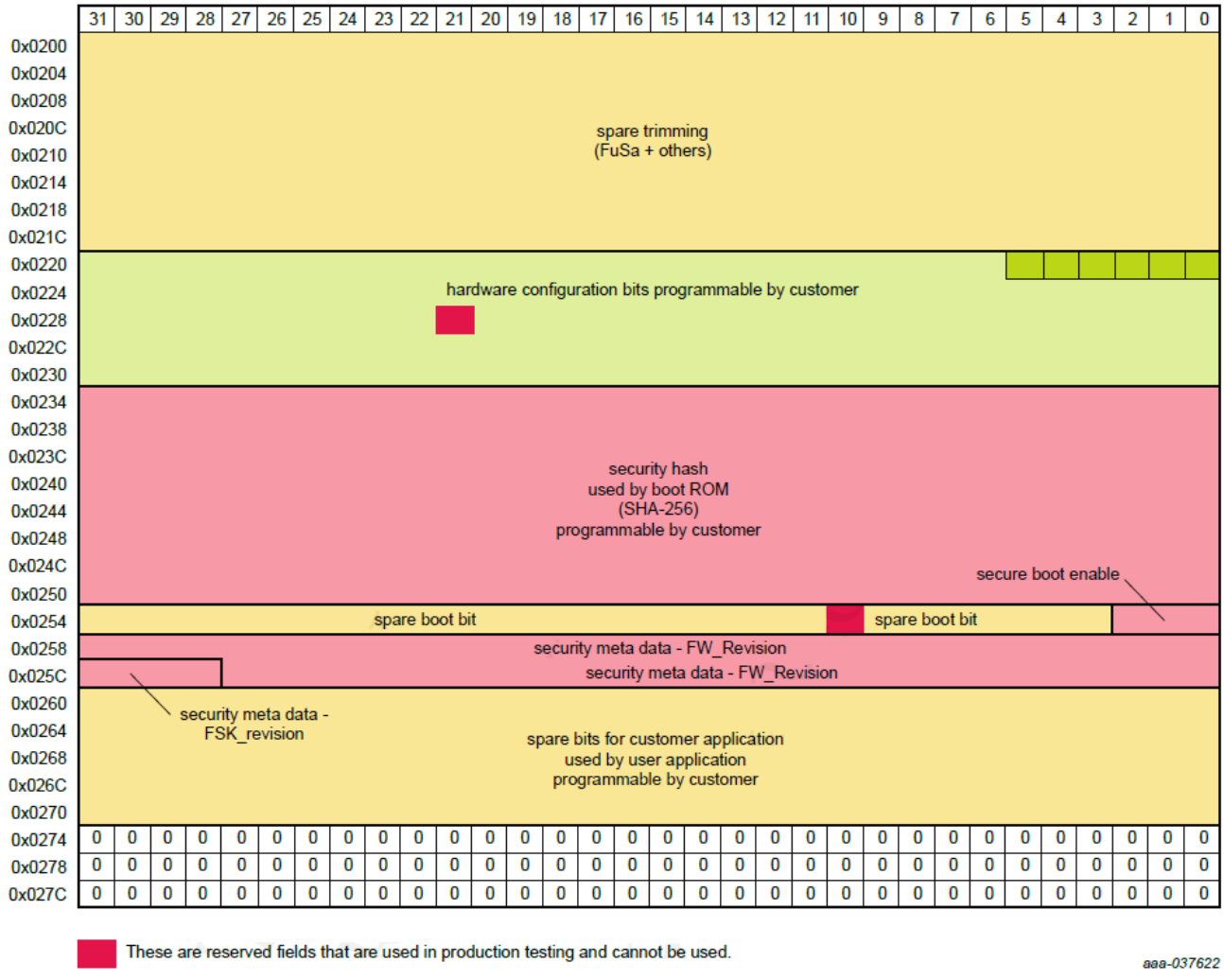
Through the OTP controller, the OTP memory instances are addressed consecutively with bit-addresses.

The OTP1 instance starts at address ***0x0000***.
The OTP1-R redundant instance starts at address ***0x0400***.

The OTP2 instance starts at address ***0x0800***.
The OTP2-R redundant instance starts at address ***0x0C00***.

If you have data with OTP memory addresses expressed like this, they must be converted.

In the previous chapter you can see the memory map of SJA1110 device.



So, to convert addresses from OTP controller notation to OTP direct memory region, you need to follow the formula below:

(((OTP Controller Target Address – OTP Controller Add Start Address) \ 0x10) * 2) + OTP Region Base Address

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

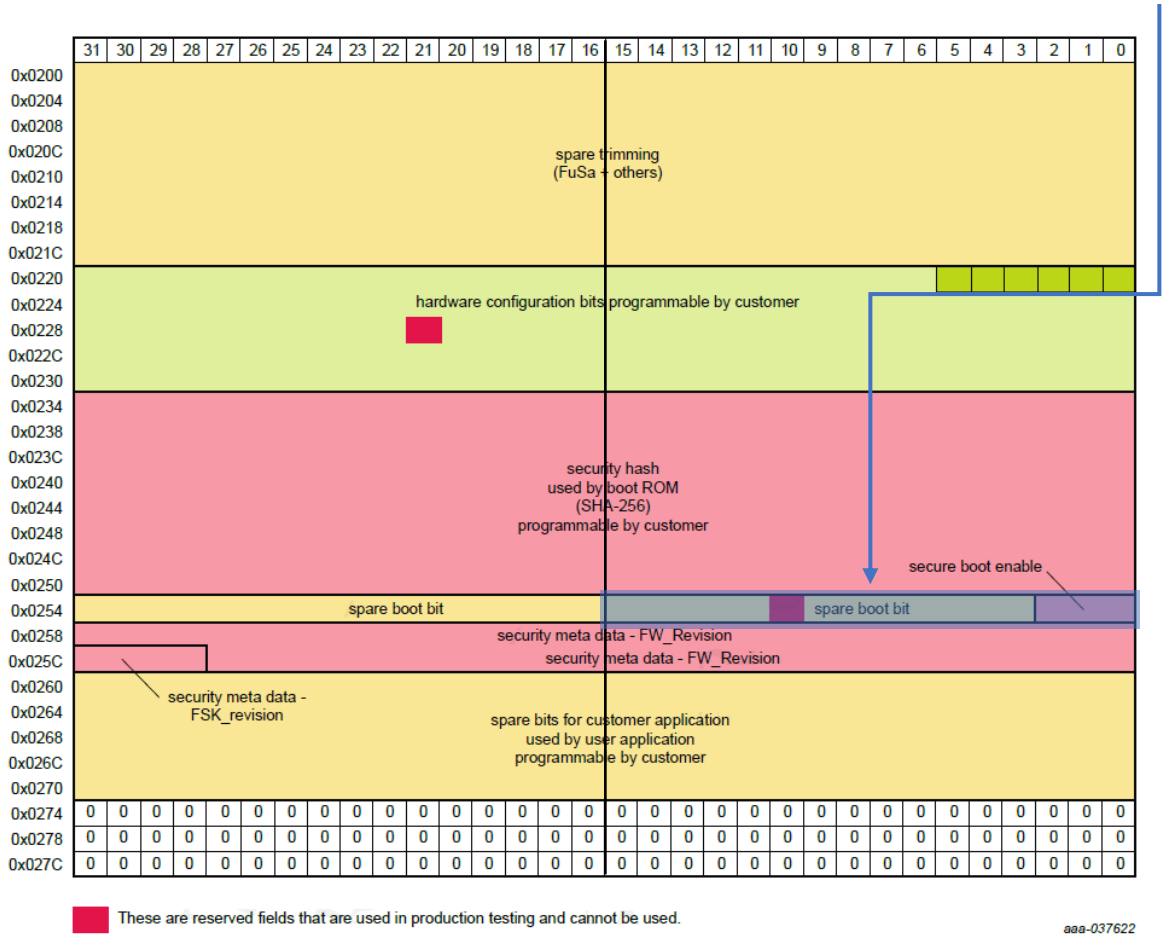
NXP SJA11 OTP Example 1

Consider you want to program the following data: Write **0x0007** data to address **0xAA0**. Register size is 16bit.

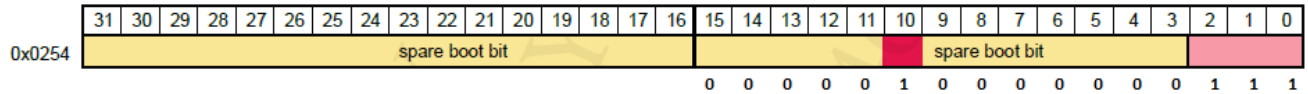
OTP Controller Target Address = 0xAA0
OTP Controller Add Start Address = 0x0800 (Base Address of OTP2 memory controller region)
OTP Region Base Address = 0x200 (Base Address of OTP2 memory region)

So, we need to apply the formula above:

= (((0xAA0 – 0x800) \ 0x10) * 2) + 0x200 = (((0x2A0) \ 0x10) * 2) + 0x200 = ((0x2A) * 2) + 0x200 = 0x54 + 0x200 = 0x254



At this point you can easily create the Dynamic Command for FlashRunner 2;
Here for example by using **#DYNMEMSET2** command: **#DYNMEMSET2** 0x0254 0x2 0700



In this specific case we need to consider the Reserved Bit (Highlighted in Red) from NXP because this bit is set to 1.
So, if you try to program and verify the value 0x0007, you will receive a failure into Verify command.

Therefore, we need to modify the **#DYNMEMSET2** command this way: **#DYNMEMSET2** 0x0254 0x2 0704

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

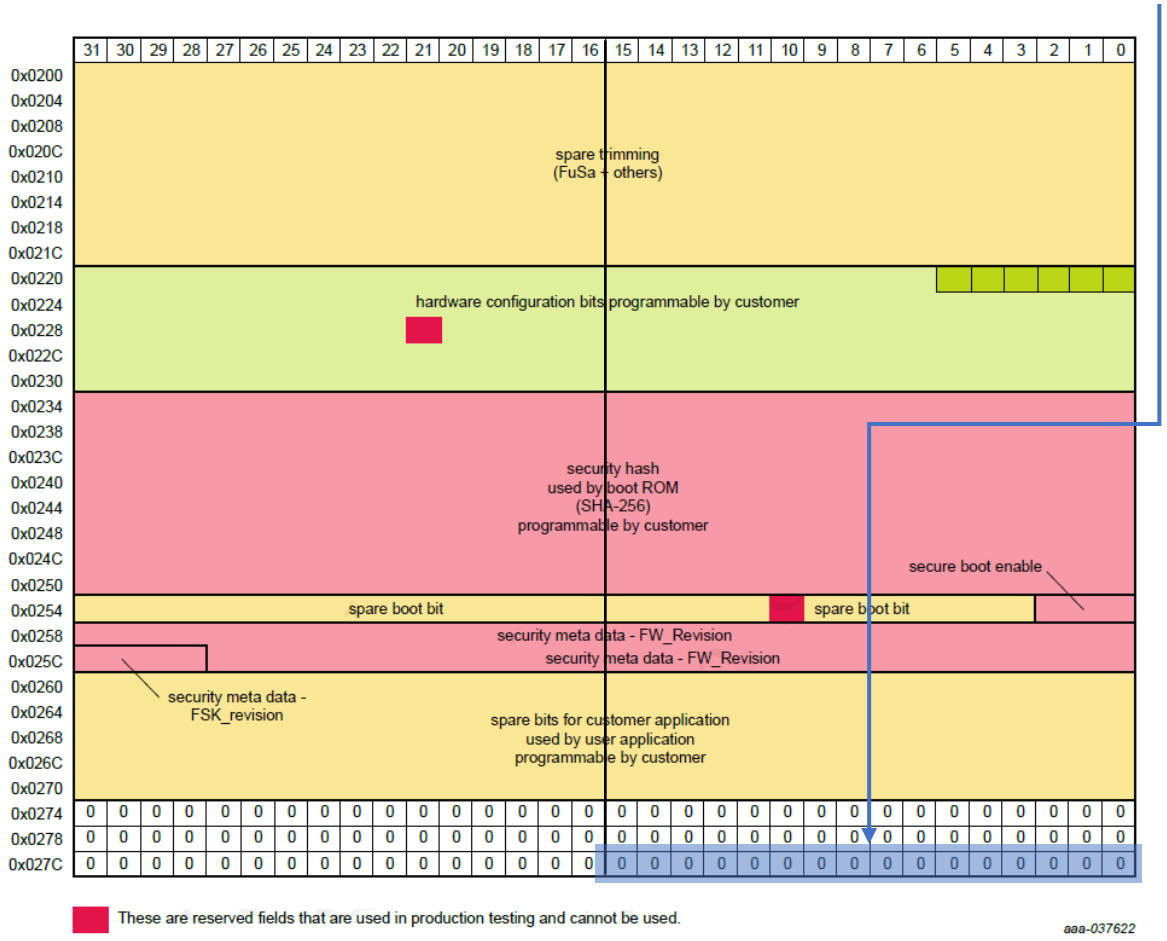
NXP SJA11 OTP Example 2

Consider you want to program the following data: Write **0xE000** data to address **0xBE0**. Register size is 16bit.

OTP Controller Target Address = 0xBE0
OTP Controller Add Start Address = 0x0800 (Base Address of OTP2 memory controller region)
OTP Region Base Address = 0x200 (Base Address of OTP2 memory region)

So, we need to apply the formula above:

= (((0xBE0 – 0x800) \ 0x10) * 2) + 0x200 = (((0x3E0) \ 0x10) * 2) + 0x200 = ((0x3E) * 2) + 0x200 = 0x7C + 0x200 = 0x27C



At this point you can easily create the Dynamic Command for FlashRunner 2;
Here for example by using **#DYNMEMSET2** command: **#DYNMEMSET2** 0x027C 0x2 00E0

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

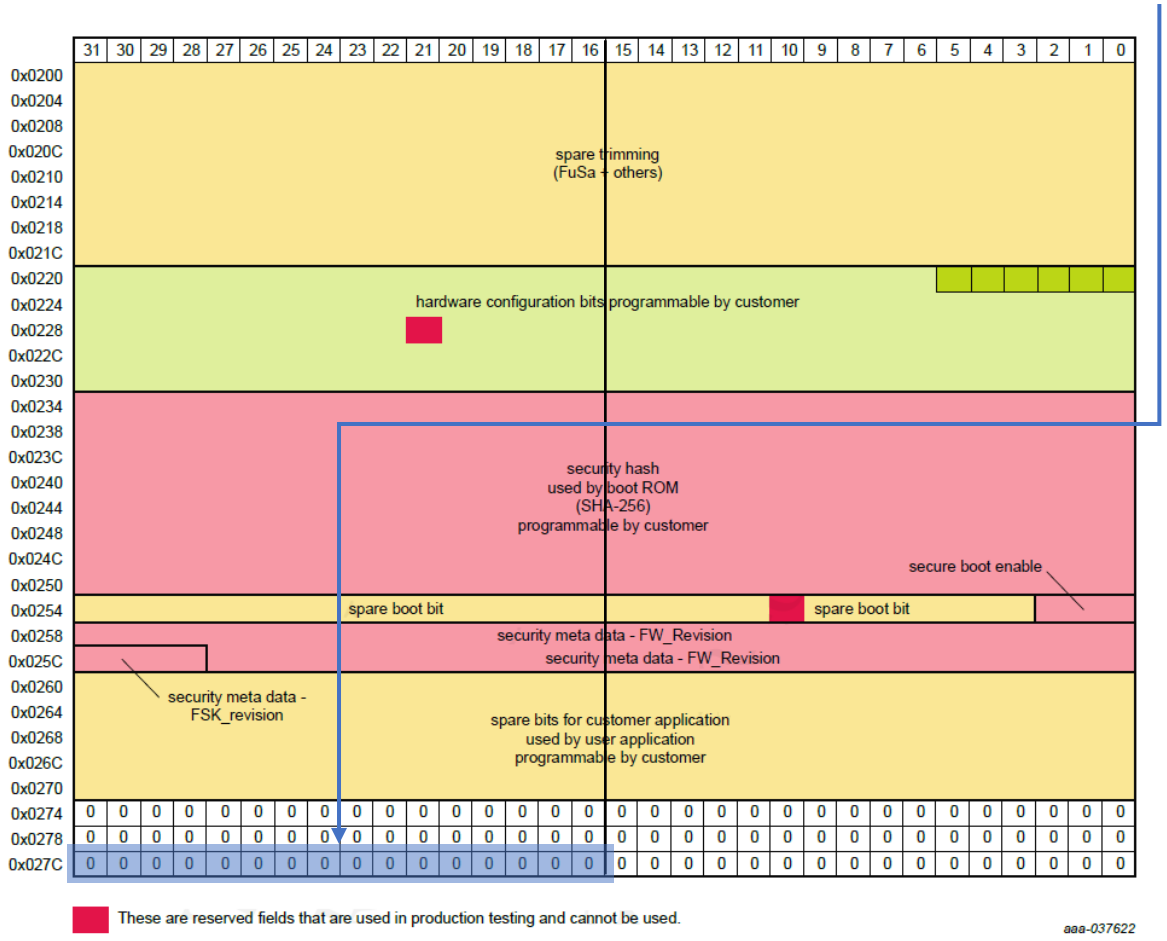
NXP SJA11 OTP Example 3

Consider you want to program the following data: Write **0x001F** data to address **0xBF0**. Register size is 16bit.

OTP Controller Target Address = 0xBF0
OTP Controller Add Start Address = 0x0800 (Base Address of OTP2 memory controller region)
OTP Region Base Address = 0x200 (Base Address of OTP2 memory region)

So, we need to apply the formula above:

= (((0xBF0 – 0x800) \ 0x10) * 2) + 0x200 = (((0x3F0) \ 0x10) * 2) + 0x200 = ((0x3F) * 2) + 0x200 = 0x7E + 0x200 = 0x27E



At this point you can easily create the Dynamic Command for FlashRunner 2;
Here for example by using **#DYNMEMSET2** command: **#DYNMEMSET2** 0x027E 0x2 1F00

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

NXP SJA11 Driver Parameters

The standard parameters are used to configure some specific options inside SJA11 driver.

#TCSETPAR ENTRY_CLOCK

Syntax: `#TCSETPAR ENTRY_CLOCK <Frequency>`

`<Frequency>` Accepted parameters 4000000, 2000000, 1000000, 500000, 100000 Hz

Description: Set the JTAG/SWD frequency used in the Connect procedure before raising the PLL of the device, if the device PLL is available

Note: Default value 4.00 MHz

#TCSETPAR SAMPLING_POINT

Syntax: `#TCSETPAR SAMPLING_POINT <Value>`

`<Value>` Accepted values are in the range 1-15

Description: Use this parameter to permanently set the sampling point of the FPGA
It is recommended to leave this parameter with the default value

Note: Default value 17

NXP SJA11 Driver Commands

Here you can find the complete list of all available commands for SJA11 driver.

```
o → OTP1 (Read Only)
O → OTP2
s → Shadow OTP1 (Read Only)
S → Shadow OTP2 (Read Only)
```

#TPCMD CONNECT

`#TPCMD CONNECT`

This function performs the entry and is the first command to be executed when starting the communication with the device.
Here you can the log of a standard connect procedure:

```
---|TPCMD CONNECT
Protocol selected SWD.
Entry Clock is 4.00 MHz.
Hot Plug connect procedure.
IDCODE: 0x0BD11477.
Designer: 0x23B, Part Number: 0xBD11, Version: 0x0.
ID-Code read correctly at 4.00 MHz.
JTAG-SWD Debug Port enabled.
Scanning AP map to find all APs.
AP[0] IDR: 0x04770041, Type: AMBA AHB3 bus.
AP[0] ROM table base address 0xE00FD000.
CPUID: 0x411FC271.
Implementer Code: 0x41 - [ARM].
Found Cortex M7 revision r1p1.
Cortex M7 Core halted [0.002 s].
Requested Clock is 25.00 MHz.
Generated Clock is 25.00 MHz.
Good samples: 6 [Range 3-8].
IDCODE: 0x0BD11477.
Designer: 0x23B, Part Number: 0xBD11, Version: 0x0.
ID-Code read correctly at 25.00 MHz.
Time for Connect: 0.105 s.
```


UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

#TPCMD PROGRAM

`#TPCMD PROGRAM <O>`
Program available only for OTP2 Memory.
Programs all memory of the selected type based on the data in the FRB file.

`#TPCMD PROGRAM <O> <start address> <size>`
Program available only for OTP2 Memory.
Programs selected part of memory of the selected type based on the data in the FRB file.
Enter the Start Address and Size in hexadecimal format.

#TPCMD VERIFY

`#TPCMD VERIFY <O> <R>`
R: Readout Mode.
Verify Readout available only for OTP2 Memory.
Verify all memory of the selected type based on the data in the FRB file.

`#TPCMD VERIFY <O> <R> <start address> <size>`
R: Readout Mode.
Verify Readout available only for OTP2 Memory.
Verify selected part of memory of the selected type based on the data in the FRB file.
Enter the Start Address and Size in hexadecimal format.

#TPCMD READ

`#TPCMD READ <o|O|s|S>`
`#TPCMD READ <o|O|s|S> <start address> <size>`
Read function for OTP1, OTP2, Shadow OTP1 and Shadow OTP2 memories.
The result of the read command will be visible into the Terminal.

#TPCMD DUMP

`#TPCMD DUMP <o|O|s|S>`
`#TPCMD DUMP <o|O|s|S> <start address> <size>`
Dump command for OTP1, OTP2, Shadow OTP1 and Shadow OTP2 memories.
The result of the dump command will be stored in the FlashRunner 2.0 internal memory.

#TPCMD OVERVIEW_OTP

Syntax: `#TPCMD OVERVIEW_SUPERVISORY_FLASH <OTP2/SHADOW_OTP2>`

`<OTP2/SHADOW_OTP2>` OTP2 or Shadow OTP2 memory

Note: This command prints into Real Time Log

Examples: Correct command execution: 😊

```
---#TPCMD OVERVIEW_OTP OTP2
Spare Trimming Area bits:
 Word 0: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 1: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 2: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 3: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 4: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 5: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 6: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 7: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
Hardware Configuration bits:
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com
info@smh-tech.com

```
 Word 0: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 1: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 2: Hex: 0x00200000 - Bin: 0000|0000|0010|0000|0000|0000|0000|0000
 Word 3: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 4: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
Security Hash used by boot ROM (SHA-256) bits:
 Word 0: Hex: 0xC5DF688C - Bin: 1100|0101|1101|1111|0110|1000|1000|1100
 Word 1: Hex: 0x1D0CAABA - Bin: 0001|1101|0000|1100|1010|1010|1011|1010
 Word 2: Hex: 0x62C3D41A - Bin: 0110|0010|1100|0011|1101|0100|0001|1010
 Word 3: Hex: 0x75063FA6 - Bin: 0111|0101|0000|0110|0011|1111|1010|0110
 Word 4: Hex: 0x8F56FFD8 - Bin: 1000|1111|0101|0110|1111|1111|1101|1000
 Word 5: Hex: 0x3332253B - Bin: 0011|0011|0011|0010|0010|0101|0011|1011
 Word 6: Hex: 0xD1C98064 - Bin: 1101|0001|1100|1001|1000|0000|0110|0100
 Word 7: Hex: 0xB61FC6F0 - Bin: 1011|0110|0001|1111|1100|0110|1111|0000
Boot bits:
 Word 0: Hex: 0x00000407 - Bin: 0000|0000|0000|0000|0000|0100|0000|0111
Security Meta Data bits:
 Word 0: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 1: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
Customer Application Spare Data bits:
 Word 0: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 1: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 2: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 3: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 4: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
Read/Write Fuse Protection bits:
 Word 0: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 1: Hex: 0x00000000 - Bin: 0000|0000|0000|0000|0000|0000|0000|0000
 Word 2: Hex: 0x001FE000 - Bin: 0000|0000|0001|1111|1110|0000|0000|0000
Time for Overview OTP2: 0.005 s
>|
```

#TPCMD RUN

Syntax: `#TPCMD RUN <Time [s]>`

<Time [s]>	Time in seconds (i.e., 2 s). This time is an optional parameter.

Prerequisites: none

Description: Move the Reset line up and down quickly if no parameter <Time [s]> is inserted.
`#TPCMD RUN <Time [s]>` instead moves the Reset line down, waits for the entered time and then sets the Reset line high.
This command typically can be used to execute the firmware programmed in the device.

#TPCMD READ_MEM8

Syntax: `#TPCMD READ_MEM8 <Address> <Byte Count>`

<Address>	Address in HEX format (i.e., 0x52002020)
<Byte Count>	Byte count in decimal format (i.e., 8 -> eight bytes)

Prerequisites: none

Description: Read memory byte per byte from target SJA11 device

Note: This command prints into Terminal and Real Time Log

Examples: Correct command execution: 😊

```
---#TPCMD READ_MEM8 0x52002020 8
Read[0x52002020]: 0xF0
Read[0x52002021]: 0xAA
Read[0x52002022]: 0x16
Read[0x52002023]: 0x14
Read[0x52002024]: 0x00
Read[0x52002025]: 0x00
Read[0x52002026]: 0x00
Read[0x52002027]: 0x00
Time for Read Mem: 0.002 s
```


HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

#TPCMD READ_MEM16

Syntax: `#TPCMD READ_MEM16 <Address> <16-bit Word Count>`

<Address>	Address in HEX format (i.e., 0x52002020)
<16-bit Word Count>	16-bit Word count in decimal format (i.e., 4 -> four 16-bit words)

Prerequisites: none

Description: Read memory 16-bit word per 16-bit word from target SJA11 device

Note: This command prints into Terminal and Real Time Log

Examples: Correct command execution: 😊

```
---#TPCMD READ_MEM16 0x52002020 4
Read[0x52002020]: 0xAAF0
Read[0x52002022]: 0x1416
Read[0x52002024]: 0x0000
Read[0x52002026]: 0x0000
Time for Read Mem: 0.002 s
```

#TPCMD READ_MEM32

Syntax: `#TPCMD READ_MEM32 <Address> <32-bit Word Count>`

<Address>	Address in HEX format (i.e., 0x52002020)
<32-bit Word Count>	32-bit Word count in decimal format (i.e., 2 -> two 32-bit words)

Prerequisites: none

Description: Read memory 32-bit word per 32-bit word from target SJA11 device

Note: This command prints into Terminal and Real Time Log

Examples: Correct command execution: 😊

```
---#TPCMD READ_MEM32 0x52002020 2
Read[0x52002020]: 0x1416AAF0
Read[0x52002024]: 0x00000000
Time for Read Mem: 0.002 s
```

#TPCMD DISCONNECT

`#TPCMD DISCONNECT`

Disconnect function. Power off and exit.

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com

NXP SJA11 Driver Changelog

Info about driver version 5.00 - 11/05/2023
Supported SJA1110 device.

Info about driver version 5.01 - 03/07/2023
Fixed wrong print into verify readout for SJA1110 device.

Info about driver version 5.02 - 29/09/2023
Internal driver update.

SMH Technologies S.r.l.

UNIVERSAL PRODUCTION IN-SYSTEM PROGRAMMING

HQ and Registered Office
Via Giovanni Agnelli 1
33083 Villotta di Chions (PN) Italy
Società Unipersonale

Capitale sociale €102.040
P.I. 01697470936
C.F. 01697470936
REA PN-97255

D-U-N-S® 51-724-9350
T + 39 0434 421 111
F + 39 0434 639 021

→smh-tech.com

info@smh-tech.com