AN00154: Interfacing FlashRunner with Renesas RX62T Devices

FlashRunner is a Universal In-System Programmer, which uses the principles of In-Circuit Programming to program Renesas RX62T family microcontrollers. This Application Note describes how to properly set up and use FlashRunner to program RX62T Flash devices.

This Application Note assumes that you are familiar with both FlashRunner and the main features of the RX62T family. Full documentation about these topics is available in the FlashRunner user's manual and in device-specific datasheets.

1. Introduction

In-system programming of RX62T microcontrollers is performed through JTAG IEEE 1149.1 standard protocol.
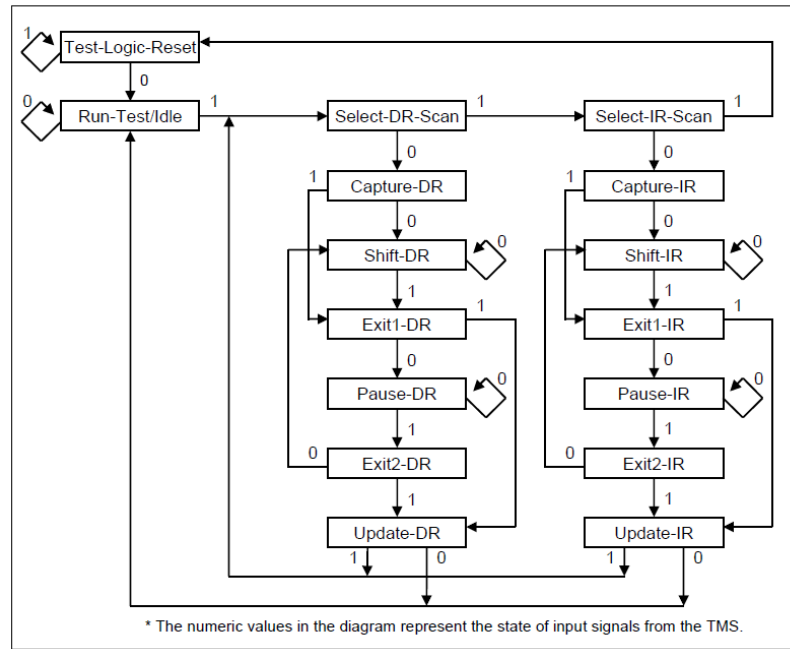
Figure 1: State Transition of the TAP Controller

In order to use FlashRunner to perform in-system programming, you need to implement the appropriate in-circuit programming hardware interface on your application board.

Thanks to its in-system programming capabilities, FlashRunner allows you to program or update the content of the Flash memory when the chip is already plugged on the application board.

2. Hardware Configuration

The microcontroller's lines needed to program a RX62T device are the following:

- **JTRST:** JTAG Module Reset/Enable Input

- **SRST:** Reset signal input pin. This LSI enters the reset state when this signal goes low.

- **JTCLK:** Test Clock.

- **JTDO:** Test Data Out

- **JTDI:** Test Data In.

- **JTMS:** Test mode select.

- **EMLE:** Input pin for the on-chip emulator enable signal. When the on-chip emulator is used, this pin should be driven high. When not used, it should be driven low.

- **VDD:** Device power supply voltage.

- **VSS:** Device power supply ground.

The lines mentioned above must be connected to the FlashRunner's "ISP" connector according to the following diagram:
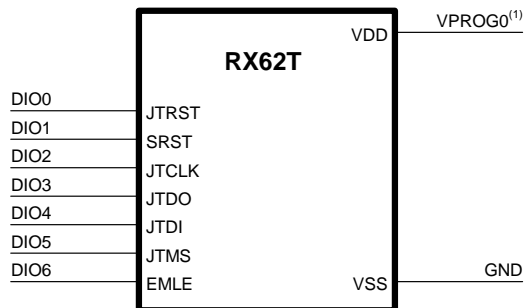
[1] Connect this line if you want FlashRunner to automatically power the target device

3. Specific TCSETPAR Programming Commands

Overview

`TCSETPAR` commands set device-specific and programming algorithm-specific parameters. These commands must be sent after the `TCSETDEV` command and before a `TPSTART` / `TPEND` command block.

All of the following parameters must be correctly specified through the relative `TCSETPAR` commands (although the order with which these parameters are set is not important):

- VDD voltage;
- Power Up time;
- Power Down time;
- Reset Up time;
- Reset Down time;
- FOSC external oscillator frequency;
- KEY CODE settings.

TCSETPAR VDD

Command syntax:

TCSETPAR VDD <voltage mV>

Parameters:

voltage mV: Target device supply voltage, expressed in millivolts.

Description:

This command is used to properly generate the voltage level of the ISP lines. Additionally, the specified voltage is routed to the VPROG0 line of the FlashRunner's "ISP" connector, which can be used as a supply voltage for the target board.

TCSETPAR PWUP

Command syntax:

TCSETPAR PWUP <time ms>

Parameters:

time ms: Power rising time, expressed in milliseconds.

Description:

This command is necessary because, to enter the programming mode, FlashRunner must properly drive the V_{DD} line during the power-on reset.

The V_{DD} rising time (PWUP) is expressed in milliseconds and depends on the features of your target board. Make sure to choose a value large enough to ensure that the V_{DD} signal reaches the high logic level within the specified time. Note that, if the V_{DD} line has a high load, a longer time is required for the V_{DD} signal to reach the high logic level. If PWUP is not long enough, FlashRunner could not be able to enter the programming mode.

TCSETPAR PWDOWN

Command syntax:

TCSETPAR PWDOWN <time ms>

Parameters:

time ms: Power falling time, expressed in milliseconds.

Description:

The V_{DD} falling time (PWDOWN) is expressed in milliseconds and depends on the features of your target board. Make sure to choose a value large enough to ensure that the V_{DD} signal reaches the low logic level within the specified time. Note that, if the V_{DD} line has a high load, a longer time is required for the V_{DD} signal to reach the low logic level.

TCSETPAR RSTUP

Command syntax:

TCSETPAR RSTUP <time µs>

Parameters:

time µs: Reset rising time, expressed in microseconds.

Description:

The Reset rising time (RSTUP) is expressed in microseconds and depends on the features of your target board. Make sure to choose a value large enough to ensure that the Reset signal reaches the high logic level within the specified time. Note that, if the Reset line has a high load, a longer time is required for the Reset signal to reach the high logic level. If RSTUP isn't long enough, FlashRunner could not be able to enter the JTAG programming mode.

TCSETPAR RSTDOWN

Command syntax:

`TCSETPAR RSTDOWN <time µs>`

Parameters:

`time µs`: Reset falling time, expressed in microseconds.

Description:

The Reset falling time (RSTDOWN) is expressed in microseconds and depends on the features of your target board. Make sure to choose a value large enough to ensure that the Reset signal reaches the low logic level within the specified time. Note that, if the Reset line has a high load, a longer time is required for the Reset signal to reach the low logic level.

TCSETPAR FOSC

Command syntax:

`TCSETPAR FOSC <frequency Hz>`

Parameters:

`frequency Hz`: External oscillator frequency, expressed in Hertz.

Description:

This command is used to set up the frequency of the external oscillator mounted on the target board.

TCSETPAR KEY_CODE CONST

Command syntax:

```
TCSETPAR KEY_CODE CONST <value1> <value2> <value3> <value4>
<value5> <value6> <value7> <value8> <value9> <valueA> <valueB>
<valueC> <valueD> <valueE> <valueF> <value10>
```

Parameters:

valueX: value of the 16 bytes considered as password. X = 0x1 ÷ 0x10.

Description:

This command is used to set 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. Programming algorithm considers them as password desired for the comparison with the password already programmed in the device.

TCSETPAR KEY_CODE SRC

Command syntax:

```
TCSETPAR KEY_CODE SRC <address>
```

Parameters:

address: Address of the source file where the password is stored.

Description:

This command is used to set up the address of the FRB source file where the password is stored. Programming algorithm reads by itself from the address specified the following 16 bytes and it considers them as password desired for the comparison with the password already programmed in the device.

Note: 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.

4. Specific TPCMD Programming Commands

Overview

TPCMD commands perform a programming operation (i.e. mass erase, program, verify, etc.) These command must be sent within a **TPSTART** / **TPEND** command block.

Renesas RX62T-specific target programming commands are the following:

- **TPCMD SETJTCLK**;

- **TPCMD MASSERASE**;

- **TPCMD ERASE**;

- **TPCMD BLANKCHECK**;

- **TPCMD PROGRAM**;

- **TPCMD VERIFY**;

- **TPCMD READ**;

- **TPCMD READ32**;

- **TPCMD RUN**.

TPCMD SETJTCLK

Command syntax:

TPCMD SETJTCLK <frequency Hz>

Command options:

frequency Hz: Specifies the JTAG clock frequency; the value is expressed in Hertz.

Description:

It selects the JTAG clock frequency in Hertz; the maximum value is 12500000Hz; it must be lower than the external oscillator frequency.

TPCMD MASSERASE

Command syntax:

TPCMD MASSERASE F|E

Command options:

F|E: Specifies that this command refer to the whole Flash memory (**F**) or to the EEPROM memory (**E**).

Description:

It erases all the Flash memory (**F**) or EEPROM memory (**E**).

TPCMD ERASE

Command syntax:

TPCMD ERASE F|E <tgt start addr> <len>

Command options:

F|E: Specifies that this command refer to the Flash memory (**F**) or to the EEPROM memory (**E**).

tgt start address: Device memory location from where the block erase operation will start.

len: Number of locations to be block erased.

Description:

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. The start address is the logical address specified in each single device datasheet or User's Guide under "Memory" chapter.

TPCMD BLANKCHECK

Command syntax:

`TPCMD BLANKCHECK F|E <tgt start addr> <len>`

Command parameters and options:

`F|E`:   Specifies Flash (`F`) memory or EEPROM (`E`) memory.

`tgt start address`:   Device memory location from where the blankcheck operation will start.

`len`:   Number of locations to be blankchecked.

Description:

It blankchecks Flash memory or EEPROM memory. Blankchecks `len` locations starting from the address specified by `tgt start address`. The start address is the logical address specified in each single device datasheet or User's Guide under "Memory" chapter.

TPCMD PROGRAM

Command syntax:

`TPCMD PROGRAM F│E <src offset> <tgt start addr> <len>`

Command parameters and options:

`F│E`:   Specifies Flash (`F`) memory or EEPROM (`E`) memory.

`src offset`:   Offset from the beginning of the source memory.

`tgt start addr`:   Device memory location from where the program operation will start.

`len`:   Number of locations to be programmed.

Description:

It programs `len` locations of Flash memory or EEPROM memory starting from the `tgt start addr` address.

`len` specifies the number of locations to be programmed. `tgt start addr` is the logical address specified in each single device datasheet or User's Guide under "Memory" chapter.

TPCMD VERIFY

Command syntax:

`TPCMD VERIFY F│E R <src offset> <tgt start addr> <len>`

Command parameters and options:

`F│E`:   Specifies Flash (`F`) memory or EEPROM (`E`) memory.

`R`   Specifies Readout (`R`).

`src offset`:   Offset from the beginning of the source memory.

tgt start addr: Device memory location from where the verify operation
  will start.

len: Number of locations to be verified

Description:

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 READ

Command syntax:

TPCMD READ F│E <tgt start addr> <len>

TPCMD READ IDCODE

Command parameters and options:

F│E: Specifies Flash (**F**) memory or EEPROM (**E**) memory.

IDCODE: Specifies ID code value to be read.

tgt start addr: Device memory location from where the read operation will
  start.

len: Number of locations to be read.

Description:

It reads **len** locations of Flash memory or EEPROM memory starting from the **tgt start addr** address. **len** specifies the number of locations to be read. If **IDCODE** is selected, the programming algorithm reads the IDCODE value of the device.

TPCMD READ32

Command syntax:

`TPCMD READ32 <tgt start addr>`

Command parameters and options:

`tgt start addr`: Device memory location from where the read operation will start.

Description:

It reads a sequence of 32 bits of Flash or EEPROM memory starting from the `tgt start addr` address. The programming algorithm returns a value formed by 4 bytes (32 bits) together.

TPCMD RUN

Command syntax:

`TPCMD RUN`

Command parameters:

None.

Description:

It runs the target application.

5. Typical Programming Flow

The following flow chart illustrates typical steps to help you write your own script file.

```
┌─────────────────────────────┐
│         TCSETDEV            │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│       TCSETPAR VDD          │
│       TCSETPAR PWUP         │
│       TCSETPAR PWDOWN       │
│       TCSETPAR RSTUP        │
│       TCSETPAR RSTDOWN      │
│       TCSETPAR FOSC         │
│       TCSETPAR KEY_CODE     │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│          TPSETSRC           │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│          TPSTART            │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│       TPCMD SETJTCLK        │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│       TPCMD MASSERASE       │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│       TPCMD BLANKCHECK      │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│        TPCMD PROGRAM        │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│        TPCMD VERIFY         │
└─────────────────────────────┘
              │
              ▼
┌─────────────────────────────┐
│           TPEND            │
└─────────────────────────────┘
```
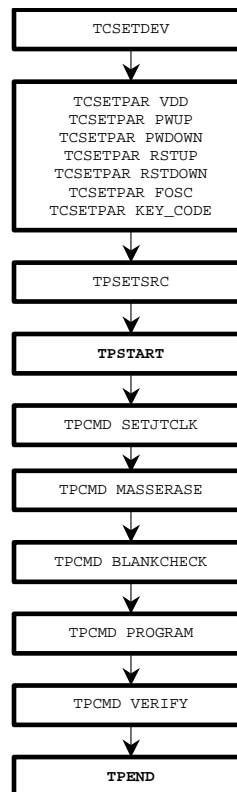
6. Script Example

The example below shows a typical programming flow for a Renesas RX62T device.

```
;
; FLASHRUNNER SCRIPT EXAMPLE FOR RENESAS R5F562TADDFP
;
; Use this example as a starting point for your specific programming needs
;
; ----------
;
; Hardware connections
;
; DIO0 (JTRST)
; DIO1 (SRST)
; DIO2 (JTCLK)
; DIO3 (JTDO)
; DIO4 (JTDI)
; DIO5 (JTMS)
; DIO6 (EMLE)
;

; Turns off logging
#LOG_ON 2
; Halt on errors
#HALT_ON FAIL

; Sets device
TCSETDEV RENESAS R5F562TADDFP RNSS_E

; Target voltage, mV (change as needed)
TCSETPAR VDD 4750

; VDD Rise-Time, ms (change as needed)
TCSETPAR PWUP 1

; VDD Fall-Time, ms (change as needed)
TCSETPAR PWDOWN 1

; RESET Rise-Time, us (change as needed)
TCSETPAR RSTUP 1000

; RESET Fall-Time, us (change as needed)
```

SMH Technologies S.r.l.
Società unipersonale

Capitale Sociale: € 10.000 i.v. - P. IVA: 01607790936 - Rea: PN-90085
via Giovanni Agnelli 1, 33083 Villotta di Chions (PN) Italy
Phone +39 0434 421111 – Fax +39 0434 639021 - info@smh-tech.com - www.smh-tech.com

SofTec Microsystems

SMH Technologies is the licensee of the SofTec Microsystems trademark

```
TCSETPAR RSTDOWN 1000


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


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


; Key-code settings (change as needed).
;TCSETPAR KEY_CODE CONST $FF $FF $FF $FF $FF $FF $FF $FF $FF $FF $FF $FF $FF $FF $FF $FF
; The Key-code data can be read from binary file.
TCSETPAR KEY_CODE SRC $0003FFA0                         ; key-code file address


; Starts programming block
TPSTART


; JTAG clock frequency, Hz (max 12500000)
; The JTAG clock frequency must be lower than the external oscillator frequency.
TPCMD SETJTCLK 6250000


; Mass erases Flash memory
TPCMD MASSERASE F


; Block erases Flash memory.
;TPCMD ERASE F $FFFC0000 $00040000                ; Erase all blocks
;TPCMD ERASE F $FFFC0000 $00004000                ; Erase EB00
;TPCMD ERASE F $FFFC4000 $00004000                ; Erase EB01
;TPCMD ERASE F $FFFC8000 $00004000                ; Erase EB02
;TPCMD ERASE F $FFFCC000 $00004000                ; Erase EB03
;TPCMD ERASE F $FFFD0000 $00004000                ; Erase EB04
;TPCMD ERASE F $FFFD4000 $00004000                ; Erase EB05
;TPCMD ERASE F $FFFD8000 $00004000                ; Erase EB06
;TPCMD ERASE F $FFFDC000 $00004000                ; Erase EB07
;TPCMD ERASE F $FFFE0000 $00004000                ; Erase EB08
;TPCMD ERASE F $FFFE4000 $00004000                ; Erase EB09
;TPCMD ERASE F $FFFE8000 $00004000                ; Erase EB10
;TPCMD ERASE F $FFFEC000 $00004000                ; Erase EB11
;TPCMD ERASE F $FFFF0000 $00004000                ; Erase EB12
;TPCMD ERASE F $FFFF4000 $00004000                ; Erase EB13
;TPCMD ERASE F $FFFF8000 $00001000                ; Erase EB14
;TPCMD ERASE F $FFFF9000 $00001000                ; Erase EB15
;TPCMD ERASE F $FFFFA000 $00001000                ; Erase EB16
;TPCMD ERASE F $FFFFB000 $00001000                ; Erase EB17
;TPCMD ERASE F $FFFFC000 $00001000                ; Erase EB18
;TPCMD ERASE F $FFFFD000 $00001000                ; Erase EB19
```

```
;TPCMD ERASE F $FFFFE000 $00001000              ; Erase EB20
;TPCMD ERASE F $FFFFF000 $00001000              ; Erase EB21


; Blank checks Flash memory (change address and length as needed).
TPCMD BLANKCHECK F $FFFC0000 $00040000


; Programs Flash memory (change address and length as needed).
TPCMD PROGRAM F $FFFC0000 $FFFC0000 $00040000


; Verifies Flash memory, read-out method (change address and length as needed).
TPCMD VERIFY F R $FFFC0000 $FFFC0000 $00040000


; ** EEPROM MEMORY ***
; Image file to be programmed (must be placed in the \BINARIES directory)
; For a correct file conversion, specify the 100000 offset in the Control Panel
TPSETSRC FILE EEPROM.FRB


; Mass erases EEPROM memory
;TPCMD MASSERASE E


; Block erases EEPROM memory.
TPCMD ERASE E $00100000 $00008000              ; Erase all blocks
;TPCMD ERASE E $00100000 $00000800              ; Erase DB00
;TPCMD ERASE E $00100800 $00000800              ; Erase DB01
;TPCMD ERASE E $00101000 $00000800              ; Erase DB02
;TPCMD ERASE E $00101800 $00000800              ; Erase DB03
;TPCMD ERASE E $00102000 $00000800              ; Erase DB04
;TPCMD ERASE E $00102800 $00000800              ; Erase DB05
;TPCMD ERASE E $00103000 $00000800              ; Erase DB06
;TPCMD ERASE E $00103800 $00000800              ; Erase DB07
;TPCMD ERASE E $00104000 $00000800              ; Erase DB08
;TPCMD ERASE E $00104800 $00000800              ; Erase DB09
;TPCMD ERASE E $00105000 $00000800              ; Erase DB10
;TPCMD ERASE E $00105800 $00000800              ; Erase DB11
;TPCMD ERASE E $00106000 $00000800              ; Erase DB12
;TPCMD ERASE E $00106800 $00000800              ; Erase DB13
;TPCMD ERASE E $00107000 $00000800              ; Erase DB14
;TPCMD ERASE E $00107800 $00000800              ; Erase DB15


; Blank checks EEPROM memory (change address and length as needed).
TPCMD BLANKCHECK E $00100000 $00008000


; Programs EEPROM memory (change address and length as needed).
TPCMD PROGRAM E $00000000 $00100000 $00008000
```

```
; Verifies EEPROM memory, read-out method (change address and length as needed).
;TPCMD VERIFY E R $00000000 $00100000 $00008000


; Ends programming block
TPEND
```

The FlashRunner's system software setup will install script examples specific for each device of the RX62T family on your PC.

7. Programming Times

The following table shows programming times for selected Renesas RX62T devices.

Device	Mem. Size	Conditions	Operations	Time
R5F562TADDFP	256KB Flash + 32KB EEPROM	FR01LAN, SCLK=6.25MHz	Erase + Blankcheck + Program + Verify	28.55 s
R5F562TADDFP	256KB Flash + 32KB EEPROM	FR04A08, SCLK=6.25MHz	Erase + Blankcheck + Program + Verify	27.05 s

Programming times depend on Programming Algorithm version, target board connections, communication mode, target microcontroller mask, and other conditions. Programming times for your actual system may therefore be different than the ones listed here. SMH Technologies reserves the right to modify Programming Algorithms at any time.

8. References

FlashRunner User's Manual.

Microcontroller-specific Datasheets and User's Guides.

SMH Technologies is the licensee of the SofTec Microsystems trademark