

AN00140: Interfacing FlashRunner with Microchip PIC18 Devices

by Mirco Ambroset (mirco.ambroset@softecmicro.com)

FlashRunner is a Universal In-System Programmer which uses the principles of In-Circuit Serial Programming (better known as ICSP) to program Microchip PIC microcontrollers. This application note describes how to properly set up and use FlashRunner to program PIC18 Flash devices.

This Application Note assumes that you are familiar with FlashRunner and with the main features of the PIC 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 PIC microcontrollers is performed by entering the device's program/verify mode, which allows the programming of the MCU memory, through the ICSP protocol. This protocol accepts serial data and clock on two of the port pins and writes the information to the appropriate target memory space.

To use FlashRunner to perform in-system programming, you need to implement the ICSP hardware interface on your application board.

Important

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

SOFTEC MICROSYSTEMS 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

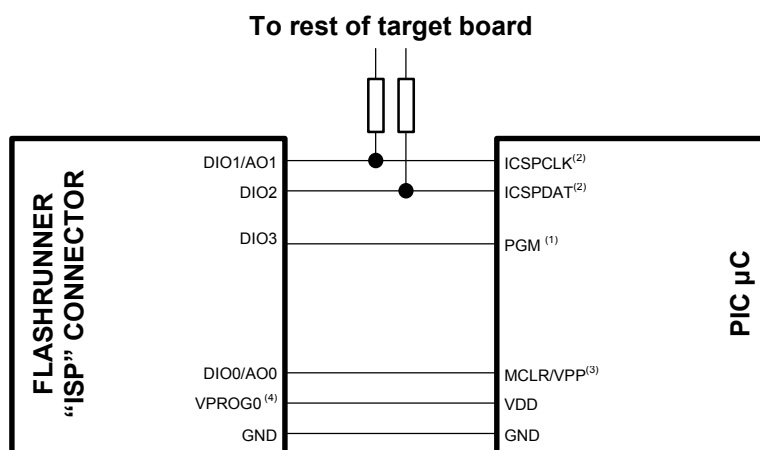
SofTec Microsystems™ and the SofTec Microsystems logo are trademarks of SofTec Microsystems.

Other products and company names listed are trademarks or trade names of their respective companies.

2. Hardware Configuration

The microcontroller lines needed to implement the ICSP protocol and program a PIC device are the following:

- **MCLR:** Program Mode select.
- **VDD:** Device power supply voltage. The range of programming voltage is typically 2.0–5.5V, depending on the MCU. This line must be connected if you want FlashRunner to automatically power the target device.
- **ICSPDAT (or DATA):** This pin is used for data input/output during serial operation.
- **GND:** Device power supply ground.
- **ICSPCLK (or CLOCK):** This pin is used for clock input during serial operation.
- **PGM:** This pin must be connected only for devices that feature the Low Voltage Programming (LVP) mode.



Notes

- (1) Connect this line only if you choose to enter program/verify mode with low voltage programming mode.
- (2) If the lines ICSPDAT and ICSPCLK needed to enter program/verify mode are used for other purposes in the application, series resistors should be implemented to avoid a conflict in case the rest of the target board forces the signal level. If these lines are used as outputs, these resistors are not necessary.
- (3) Special care should be taken in isolating the MCLR/VPP line from the rest of your target board because, in high-voltage programming mode, this signal could have a voltage level about 13V—which could damage the target board if the necessary measures are not taken. Note also that large capacitive loads on the MCLR/VPP line could increase the signal rise time, preventing the MCU from entering the Program/Verify mode.
- (4) Connect this line if you want FlashRunner to power the target board (recommended).

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 FlashRunner programming capabilities rely on the In-Circuit Serial Programming (ICSP) of the target device. In order to enter this special mode (which establishes a communication channel between the target device and FlashRunner) and configure it properly, 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):

- V_{DD} ;
- Power up time;
- Power down time;
- Serial clock frequency;
- Programming mode;
- Programming voltage (only for high-voltage programming mode).

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 signals. Additionally, the specified voltage is routed to the VPROG0 line of the FlashRunner "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 Program/Verify 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 Program/Verify 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 SCLK

Command syntax:

```
TCSETPAR SCLK <frequency Hz>
```

Parameters:

frequency Hz: serial communication clock frequency, expressed in Hertz.

Description:

This command sets the serial communication clock frequency. This frequency should not exceed the device's internal RC clock frequency, which is typically 4MHz.

When powering the device with a low voltage (less than 4.5V) or when the serial communication lines have high capacitive loads, the serial communication clock frequency should be lowered in order to avoid communication errors.

TCSETPAR PROGMODE**Command syntax:**

```
TCSETPAR PROGMODE HVP|LVP
```

Command options:

HVP: uses the High-Voltage Programming mode.

LVP: uses the Low-Voltage Programming mode.

Description:

Sets the programming mode.

The High-Voltage Programming (HVP) mode is supported by all devices. Some devices also support the Low-Voltage Programming (LVP) mode.

When using the HVP mode, a high programming voltage (typically in the range 10-13V) must be supplied to the device's MCLR/VPP pin.

When using the LVP mode, no high voltage needs to be supplied to the MCLR/VPP pin, but an additional line (PGM) must be connected to FlashRunner. Additionally, the LVP mode is only available if the LVP bit in the configuration word is set to 1 (default value for blank devices).

On PIC18J series devices, the **TCSETPAR** command is not mandatory. If used, however, it must specify the LVP mode.

TCSETPAR VPP**Command syntax:**

```
TCSETPAR VPP <voltage mV>
```

Parameters:

voltage mV: Target device programming voltage, expressed in millivolts. Typically in the 10-13V range.

Description:

This command is only needed for the High-Voltage Programming (HVP) mode, and is used to properly generate the appropriate programming voltage in the MCLR/VPP line.

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.

The PIC18-specific target programming commands are the following:

- **TPCMD BLANKCHECK;**
- **TPCMD MASSERASE;**
- **TPCMD PROGRAM;**
- **TPCMD VERIFY;**
- **TPCMD RUN.**

TPCMD BLANKCHECK

Command syntax:

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

Command options and parameters:

F E:	Memory type to be blankchecked. F stands for Flash memory and E stands for EEPROM memory.
tgt start address:	Device memory location from where the blankcheck operation will start.
len:	Number of locations to be blankchecked.

Description:

Blankchecks Flash or EEPROM memory. Blankchecks **len** locations starting from the address specified by **tgt start address**.

TPCMD MASSERASE

Command syntax:

```
TPCMD MASSERASE F <tgt row start addr> <len>
```

```
TPCMD MASSERASE E|C
```

Command options:

F E C:	Memory type to be mass erased (Flash, EEPROM, Complete).
tgt row start addr:	Row start address.

len: Length of the memory block to mass erase.

Description:

Mass erases Flash memory, EEPROM memory, or both.

TPCMD MASSERASE F <tgt row start addr> <len> mass erases one or more Flash memory rows (program memory). The **tgt row start addr** parameter must be the first location in a row, and **len** must be a multiple of a row length. Note: this command only works with devices that support row erasing.

TPCMD MASSERASE E mass erases the EEPROM memory.

TPCMD MASSERASE C performs a complete mass erase (program memory, EEPROM, user ID locations and configuration words). **TPCMD MASSERASE C** is also needed for mass erasing protected devices.

TPCMD PROGRAM

Command syntax:

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

Command options and parameters:

F: Flash memory (program memory, user ID locations and configuration words).

E: EEPROM 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:

Programs **len** locations in the Flash or EEPROM memory starting from the **tgt start addr** address.

Please note that program memory, user ID locations and configuration words (both in Flash) must be programmed separately.

Note for Devices NOT Belonging to the PIC18J Series

Please note that an Intel-Hex file generated by Microchip development tools contains both Flash memory (including user ID locations and configuration words) and EEPROM memory. Every memory zone to be programmed to the target device must be separately converted to a FlashRunner binary file.

Conversion is done by using the `fr_bin2frb` utility, installed in the local in the local **UTILITIES** folder. The syntax of the `fr_bin2frb` utility is the following:

```
fr_bin2frb -i <hex file> -t hex -s <start addr> -e <end addr> -u FF -o
          <output file>
```

where `<start addr>` and `<end addr>` are the start and end address, respectively, of the memory zone under consideration, as logically mapped in the hex file.

The following examples show how to correctly generate FlashRunner binary files for a PIC18F4520 device.

Example of program memory code conversion:

```
fr_bin2frb -i source.hex -t hex -s 0 -e 7FFF -u FF -o flash.frb
```

Example of ID locations conversion:

```
fr_bin2frb -i source.hex -t hex -s 200000 -e 200007 -u FF -o usid.frb
```

Example of configuration words conversion:

```
fr_bin2frb -i source.hex -t hex -s 300000 -e 30000D -u FF -o cw.frb
```

Example of EEPROM conversion:

```
fr_bin2frb -i source.hex -t hex -s F00000 -e F000FF -u FF -o eeprom.frb
```

(Please note that, on Intel-Hex files generated by Microchip development tools, EEPROM locations are logically mapped starting from address 0xF00000)

The portion of script that programs and verifies all of the four memory zones will look like the following:

```
; Programs and verifies Flash memory
TPSETSRC FILE FLASH.FRB
TPCMD PROGRAM F $0 $0 32768
TPCMD VERIFY F $0 $0 32768

; Programs and verifies EEPROM memory
TPSETSRC FILE EEPROM.FRB
TPCMD PROGRAM E $0 $0 256
TPCMD VERIFY E $0 $0 256

; Programs and verifies UserId Locations
TPSETSRC FILE USID.FRB
TPCMD PROGRAM F $0 $200000 8
TPCMD VERIFY F $0 $200000 8

; Programs and verifies configuration words
TPSETSRC FILE CW.FRB
TPCMD PROGRAM F $0 $300000 14
TPCMD VERIFY F $0 $300000 14
```


TPCMD VERIFY

Command syntax:

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

Command options and parameters:

F:	Flash memory (program memory, user ID locations and configuration words).
E:	EEPROM memory.
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:

Verifies **len** locations in the Flash or EEPROM memory starting from the **tgt start addr** address.

Please note that program memory, user ID locations and configuration words (both in Flash) must be verified separately.

TPCMD RUN

Command syntax:

```
TPCMD RUN
```

Command parameters:

None.

Description:

Runs the target application.

Typical

5. Script Examples

The examples provided below will help you understand how the commands discussed above should be used for two typical PIC devices: a PIC18F4520 device and a PIC18F67J50 device.

```

;
; FLASHRUNNER SCRIPT EXAMPLE FOR MICROCHIP PIC18F4520
;
; Use this example as a starting point for your specific programming needs
;
; -----
;
; Hardware connections
;
; DIO0 (MCLR)
; DIO1 (ICSPCLK)
; DIO2 (ICSPDAT)
; DIO3 (PGM) only for low voltage programming mode
;
; Turns off logging
#LOG_OFF
; Halt on errors
#HALT_ON FAIL
; Sets device
TCSETDEV MICROCHIP PIC18F4520 PIC18
; Oscillator frequency, Hz (change as needed)
TCSETPAR SCLK 4000000
; Power Supply Voltage, mV (change as needed)
TCSETPAR VDD 5000
; Sets High Voltage Program Mode
TCSETPAR PROGMODE HVP
; Programming Voltage value, mV (change as needed)
TCSETPAR VPP 13250
; VDD rise-time, ms (change as needed)
TCSETPAR PWUP 1
; VDD fall-time, ms (change as needed)
TCSETPAR PWDOWN 1
; Starts programming block
TPSTART
; Erases FLASH, EEPROM, Configuration Words and User Id Locations also if code protected
TPCMD MASSERASE C
; Blank checks Flash memory, UserId and Configuration Words (change address and length as needed)
TPCMD BLANKCHECK F $0 3145742
; Blank Checks EEPROM memory (change address and length as needed)
TPCMD BLANKCHECK E $0 256
; Image file to be programmed (must be placed in the \BINARIES directory)
TPSETSRC FILE FLASH.FRB
; Programs Flash memory (change addresses and length as needed)
TPCMD PROGRAM F $0 $0 32768
; Verifies Flash memory (change addresses and length as needed)
TPCMD VERIFY F $0 $0 32768
; Image file to be programmed into the EEPROM(must be placed in the \BINARIES directory)
TPSETSRC FILE EEPROM.FRB
; Programs EEPROM memory
TPCMD PROGRAM E $0 $0 256
; Verifies EEPROM memory
TPCMD VERIFY E $0 $0 256
; Image file to be programmed into the User Id Locations (must be placed in the \BINARIES directory)
TPSETSRC FILE USID.FRB
; Programs UserId Locations
TPCMD PROGRAM F $0 $200000 8
; Verifies UserId Locations
TPCMD VERIFY F $0 $200000 8
; Image file to be programmed into the Configuration Words (must be placed in the \BINARIES directory)
TPSETSRC FILE CW.FRB
; Programs Configuration Words (change addresses and length as needed)
TPCMD PROGRAM F $0 $300000 14
; Verifies Configuration Words (change addresses and length as needed)
TPCMD VERIFY F $0 $300000 14
; Ends programming session
TPEND

```

```
;
;
; FLASHRUNNER SCRIPT EXAMPLE FOR MICROCHIP PIC18F67J50
;
;
; Use this example as a starting point for your specific programming needs
;
; -----
;
; Hardware connections
;
; DIO0 (MCLR)
; DIO1 (ICSPCLK)
; DIO2 (ICSPDAT)
;
; Turns off logging
#LOG_OFF
; Halt on errors
#HALT_ON_FAIL
; Sets device
TCSETDEV MICROCHIP PIC18F67J50 PIC18
; Oscillator frequency, Hz (change as needed)
TCSETPAR SCLK 4000000
; Power Supply Voltage, mV (change as needed)
TCSETPAR VDD 3300
; VDD rise-time, ms (change as needed)
TCSETPAR PWUP 1
; VDD fall-time, ms (change as needed)
TCSETPAR PWDOWN 1
; Starts programming block
TPSTART
; Erases FLASH and Configuration Words also if code protected
TPCMD MASSERASE C
; Blank checks Flash memory and Configuration Words (change address and length as needed)
TPCMD BLANKCHECK F $0 131070
; Image file to be programmed (must be placed in the \BINARIES directory)
TPSETSRC FILE FLASH.FRB
; Programs Flash memory and Configuration Words (change addresses and length as needed)
TPCMD PROGRAM F $0 $0 131070
; Verifies Flash memory and Configuration Words (change addresses and length as needed)
TPCMD VERIFY F $0 $0 131070
; Ends programming session
TPEND
```

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

6. Programming Times

The following table shows programming times for several devices representative of the PIC18 families.

Device	Memory Size	Comm. Frequency	Operations	Time
PIC18F2520	32 KB Flash	4 MHz	Program	1.70 s
PIC18F2520	32 KB Flash	4 MHz	Erase + Program + Verify	2.50 s
PIC18F6520	32 KB Flash	8 MHz	Program	1.80 s
PIC18F6520	32 KB Flash	8 MHz	Erase + Program + Verify	2.80 s
PIC18F6621	64 KB Flash	8 MHz	Program	2.40 s
PIC18F6621	64 KB Flash	8 MHz	Erase + Program + Verify	4.10 s
PIC18F66J15	96 KB Flash	8 MHz	Program	5.70 s
PIC18F66J15	96 KB Flash	8 MHz	Erase + Program + Verify	8.50 s
PIC18F65J90	32 KB Flash	8 MHz	Program	2.00 s
PIC18F65J90	32 KB Flash	8 MHz	Erase + Program + Verify	3.20 s

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

7. References

FlashRunner user's manual
Microcontroller-specific datasheets