AN204471

# FM4 S6E2CC Series External Memory Programmer

**Targeted Product: FM4 MCU S6E2CC**

This user manual describes how to use the FMx MCU Universal Programmer (named as PGM hereinafter) to act as an off-line programmer for Quad SPI flash memory programming on the Orion SK.

## Contents


1 Introduction....1
1.1 About Document....1
1.2 About Off-line Programmer....1
1.3 About FMx MCU....1
1.4 Programmer Feature....1
2 Component....2
2.1 PGM Board....2
2.2 Configuration File Format....5
3 Operation Process....7
3.1 SWD Programming Process....7
4 Maintenance....10
5 Reference Documents....10
6 Document History....11


# 1 Introduction

## 1.1 About Document

This user manual describes how to use the FMx MCU Universal Programmer (named as PGM hereinafter) to act as an off-line programmer for Quad SPI flash memory programming on the Orion SK.

## 1.2 About Off-line Programmer

With the off-line programmer, the user can update the external memory on the board under the software control without removing the mounted memory chip from the actual end product.

## 1.3 About FMx MCU

Spansion's FM microcontrollers incorporate the latest ARM® Cortex™ standard cores (M4, M3, M0+ and R4), offering customers the optimal product for a wide range of industrial and consumer applications. The scalable platform ranges from low-pin-count, low-power microcontrollers to high-performance products with a rich set of peripherals.

- Outstanding performance
- Functional safety
- High-performance flash memory
- Advanced peripherals

Now there are 13 different types of MCU in FM3 family. For FM4, the type number is 4, and FM0+ is 1. For all these different types of MCUs, there are 3 types of flash structure: Main, Dual, and Main + Work. This enables the customers to choose the appropriate MCU for the application according to the flash structure.

The Orion is one type FM4 MCU of FMx MCU.

## 1.4 Programmer Feature

The main features of this programmer are as follows:

1. Support the Quad SPI Flash programming on the Orion SK.
2. Program interface: SWD.

3. Storage media: SD Card.
4. One key operation.
5. Power supply optional: USB/DC In/Battery.
6. Status display by 3 LEDs.
7. Programming file operation is controlled by a configuration file output by PC.

# 2 Component

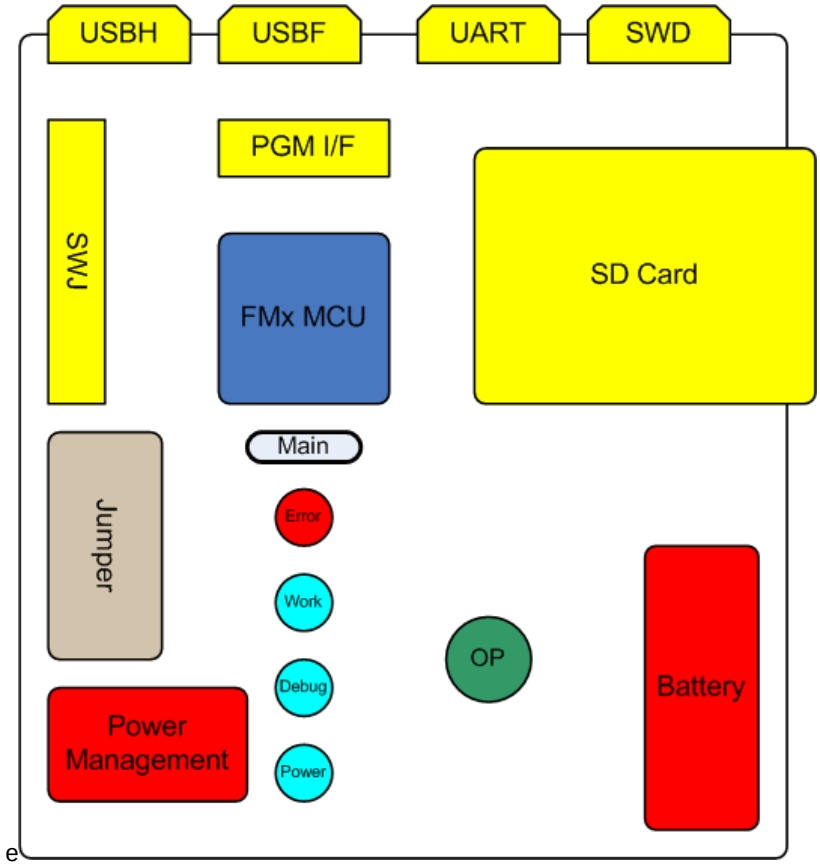
The whole PGM includes: the board and the PC tool (notepad).

## 2.1 PGM Board

The board provides the hardware method to operate the target board.

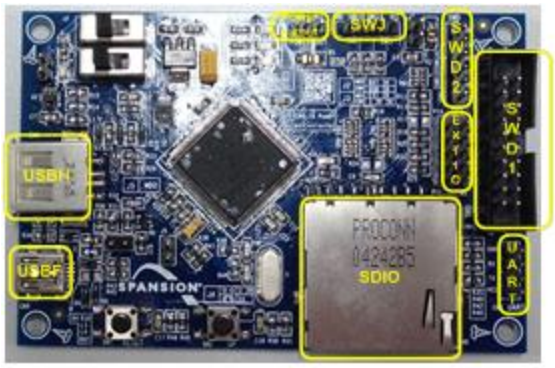
The architecture of the whole system is shown below

Figure 1. PGM architecture



The interfaces are shown below.

Figure 2. PGM interface




**UART:**

4-pin interface, this interface can be used to act as the communication interface to program the target board through asynchronous protocol.

**SWD1:**

20-pin interface, this interface can be used to act as the communication interface to program the target board through SWD protocol.

**SWD2:**

6-pin interface, the function is same as SWD1.

**USBH:**

USB host socket, USB-Disk with the target files can be connected with the PGM through this interface.

**USBF:**

USB device socket, used for power supply or acted as the debug interface for CMSIS-DAP (not supported yet).

**SWJ:**

System debug interface, used to debug the PGM.

**SDIF:**

SD card socket, SD card with the target files can be connected with the PGM through this interface; support SDSC and SDHC cards.

**Power output:**

Provide DC power output.

The jumpers are described in the table below.

**Table 1. Jumper table**

| Jumper | Function | Setting |
|---|---|---|
| J1 | JTAG power output | Open: No power output from CN4-19<br>Close: Power output from CN4-19 |
| J2 | Battery/USB input selection | Right: Battery input<br>Left: USB input |
| J3 | Ext power voltage selection | Right: Direct external power input<br>Left: Adjust the external input power to 3.3V |
| J4 | USB input voltage adjusting | Open: If input voltage is 3.3V<br>Close: If input voltage is 5V |
| J5 | Mode pin | Open: Normal run mode<br>Close: BI code run mode |
| J7 | Function pin | Open: UART program mode<br>Close: SWD program mode |

Power supply

The PGM can be powered by:

USB: J2 → Right

Battery: J2 → Left

Voltage selection:

1. Use 3.3V

USB (5V)/ Battery (4.5V) input

J2 → Right/Left

J3 → Left

J4 → Close

J5 → Open

2. Use 5V

USB (5V) input

J2 → Left

J3 → Right

J4 → Open

J5 → Open

## 2.2 Configuration File Format

The configuration file is use by the PGM as the guideline to operate the Quad SPI flash on the Orion SK.

The format is:

File Number

File1 name

File1 size

File1 start location (block number)

File1 block count

File1 checksum

File2 name

File2 size

File2 start location (block number)

File2 block count

File2 checksum

Please check the sample file as following content:

```
4
421Main1.bin
1055493
1
17
0x55
421Main2.bin
1055494
18
17
0x66
421Main3.bin
1055495
35
17
0x77
421Main4.bin
1055492
52
17
0x88
```

Please note:

- The maximum number of the programming file is 5.
- The block 0 can't be used as the programming data area (don't use 0 as the file start location).
- The block 127 can't be used as the programming data area (don't make the file content overlap the block 127).
- Please use some checksum tools to generate the checksum of the file. (Please input as the 0xaa mode)
- Please press the enter key after the final line of the configuration file.
- The file name should follow the 8.3 file naming rule.

The maximum file size should be less than 8257536.

# 3 Operation Process

## 3.1 SWD Programming Process

The user can program the target board through SWD communication interface. The operation steps are as follows:

1. Prepare the downloading files

The user can prepare at most 5 files to download and copy to the SD card.

2. Write the program.ini file

Please follow the 2.2 Configuration file format and copy to the SD card.

3. Hardware connection and setting

Power and jumper setting

| USB (5V) power input | Battery (4.5V) power input |
|---|---|
| J1: Close | J1: Close |
| J2: Left | J2: Right |
| J3: Left | J3: Left |
| J4: Open | J4: Open |
| J5: Open | J5: Open |
| J7: Close | J7: Close |

Communication line (SWD) connection

SWD1 (pin number):

Pin 2 (Universal PGM) ←→ GND (Target Board)

Pin 7 (Universal PGM) ←→ SWDIO (Target Board)

Pin 9 (Universal PGM) ←→ SWDCLK (Target Board)

Pin 19 (Universal PGM) ←→ Vcc (Target Board)

SWD 2 (pin function):

GND (Universal PGM) ←→ GND (Target Board)

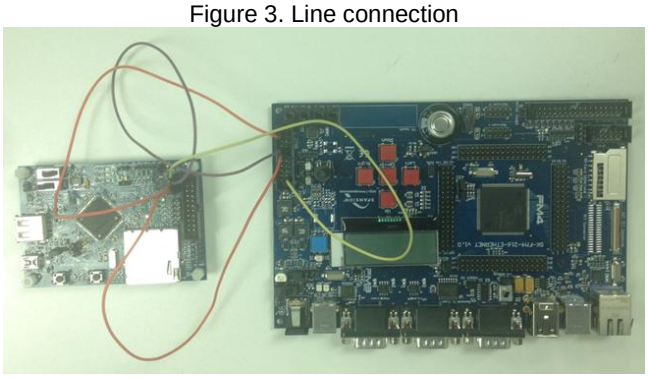
TMS (Universal PGM) ←→ SWDIO (Target Board)

TCK (Universal PGM) ←→ SWDCLK (Target Board)

VCC (Universal PGM) ←→ Vcc (Target Board)

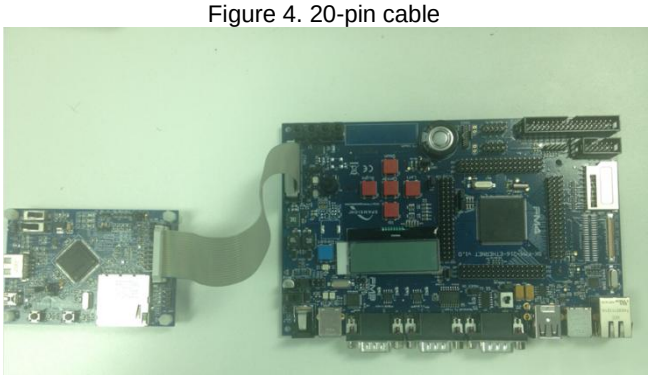
The connection is shown in the following figure. Connect the target board with the PGM through lines (VCC/SWDIO/SWCLK/GND):

Figure 3. Line connection



Or the user can use the standard 20-pin cable as shown in the following figure:

Figure 4. 20-pin cable



SD Card insertion

Insert the SD card into the SD socket correctly.

Operation

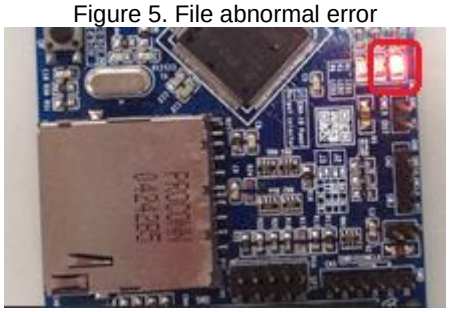
Please set the Mode 0 pin of the target board to high and set Mode 1 pin to low.

Power on the PGM, if there is no SD card in the socket; no LED will be turned on.

If there is an SD card in the socket and there is correct program.ini file in the SD card, 3 LEDs will be turned on.

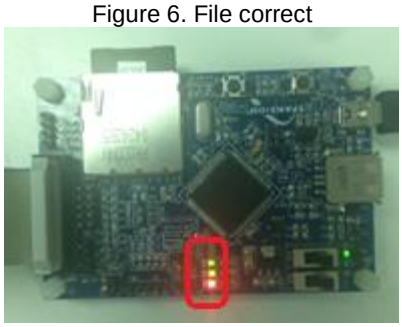
If there is an SD card in the socket but the program.ini file is not correct, LED1 will be on as shown in the following figure:

Figure 5. File abnormal error



If the SD card is inserted and the target file is correct, the LED status will be shown as the following figure:

Figure 6. File correct

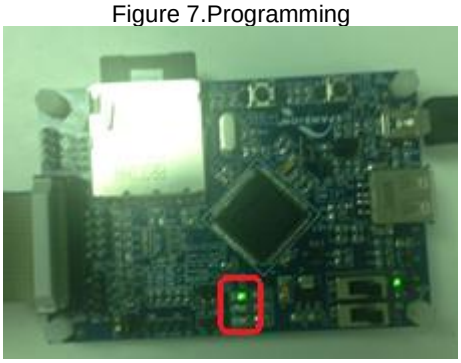


If the LED status is as shown above (all LED on), the user can press the ‘Op’ key to start the programming of the target board.

Result and status check

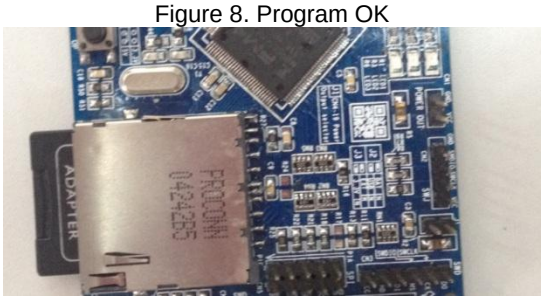
During the programming process, the status of LED3 is as shown in the following figure (LED3 on).

Figure 7.Programming



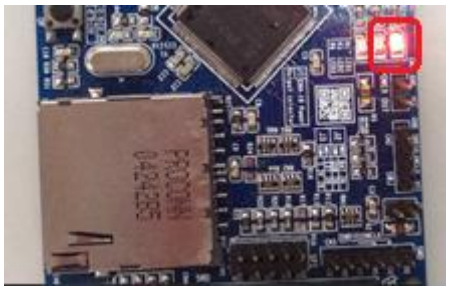
If the program is successful, the LED status is shown as the following figure (all off):

Figure 8. Program OK



Or the LED status is shown as the following figure (LED1 on):

Figure 9. Program error



If any error occurs, the user can check the program.ini to confirm the format (checksum, size or unit number).

# 4 Maintenance

None

# 5 Reference Documents

**SWD related:**

IHI0031A_ARM_debug_interface_v5.pdf

**FM4 related:**

4_08-3_HS-QSPIcontroller_E_fromMB91590.pdf

FM4F_S6E2CC Series_DS_E_Preliminary-.pdf

# 6 Document History

Document Title: AN204471 - FM4 S6E2CC Series External Memory Programmer

Document Number: 002-04471

| Revision | ECN | Orig. of Change | Submission Date | Description of Change |
|---|---|---|---|---|
| ** | _ | CPQI | 03/20/2015 | Initial Release |
| *A | 5028495 | CPIQ | 12/02/2015 | Migrated Spansion Application Note “S6E2CC_AN709-00018-1v0-E” to Cypress format |
| *B | 5874988 | AESATMP9 | 09/06/2017 | Updated logo and copyright. |

## Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

## Products

| | |
|---|---|
| ARM® Cortex® Microcontrollers | cypress.com/arm |
| Automotive | cypress.com/automotive |
| Clocks & Buffers | cypress.com/clocks |
| Interface | cypress.com/interface |
| Internet of Things | cypress.com/iot |
| Memory | cypress.com/memory |
| Microcontrollers | cypress.com/mcu |
| PSoC | cypress.com/psoc |
| Power Management ICs | cypress.com/pmic |
| Touch Sensing | cypress.com/touch |
| USB Controllers | cypress.com/usb |
| Wireless Connectivity | cypress.com/wireless |

## PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

## Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

## Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2015-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.