# System Controller

## User Manual

### Purpose

The purpose of this document is to present the characteristics of System Controller to the user, and to provide the user with a comprehensive guide to understanding and using the System Controller.

### Summary

This document first gives an overview of the System Controller followed by a detailed description of its features and configuration options.

| Product Information | Code | Name |
|---|---|---|
| Product | SC | System Controller |

| Document Information | Reference | Version | Date |
|---|---|---|---|
| Reference / Version / Date | D-0000-215-002 | 02 | 08.07.2021 |

| Approval Information | Name | Position | Date |
|---|---|---|---|
| Written by | ABUE | Design Engineer | 08.07.2021 |
| Verified by | GKOE | Design Expert | 08.07.2021 |
| Approved by | DIUN | Manager, BU SP | 08.07.2021 |

**Enclustra GmbH** – Räffelstrasse 28 – CH-8045 Zürich – Switzerland
Phone +41 43 343 39 43 – www.enclustra.com

## Copyright Reminder



## Document History

| Version | Date | Author | Comment |
|---|---|---|---|
| 01 | 31.01.2020 | ABUE | Version 01 |
| 02 | 08.07.2021 | ABUE | Minor fixes in documentation of register bank |

## Table of Contents

# 1 Design Description

## 1.1 Overview

Figure 1 shows a block diagram of the System Controller design. A description of each block can be found in the following sections.

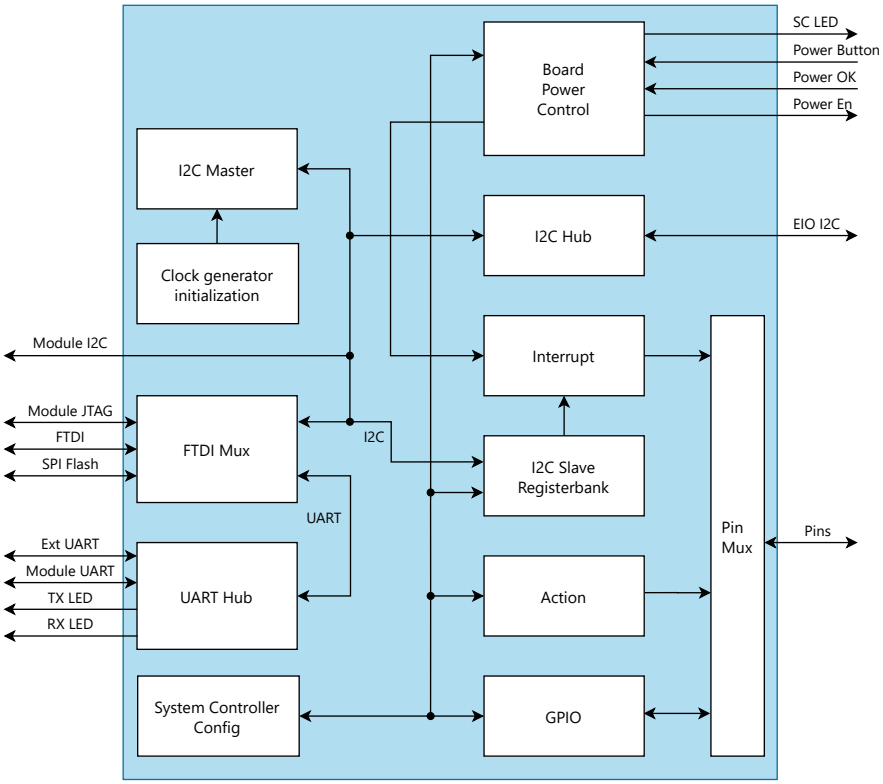


*Figure 1: System Controller Overview*

## 1.2 Board Power Control

Figure 2 shows the state machine to control the board power. If the DIP switch PWR_ON# is enabled, the board is always powered and the board power control state machine has no effect. If the DIP switch PWR_ON# is disabled, the board power can be enabled by pressing the power button. If the power button is pressed for longer than 3 seconds while the board is powered, the board power is turned off.

The button press duration until the power is turned off can be configured by BoardPowerButtonTime register (address 0x04) from disabling power immediately up to 12.75 seconds in steps of 50 milliseconds. If the button is pressed and released before the configured time has elapsed, the power is not turned off. If the power button interrupt is enabled (see section 1.6), the board power will not be turned off but instead an interrupt is generated to allow the user software to shut down the device in a controlled way. The board power can be disabled immediately by bit 0 in BoardPowerControl register (address 0x02) regardless if the power button interrupt is enabled or not. As alternative to pressing the power button, the power can also be enabled by writing 0x01 to BoardPowerControl register (address 0x02).

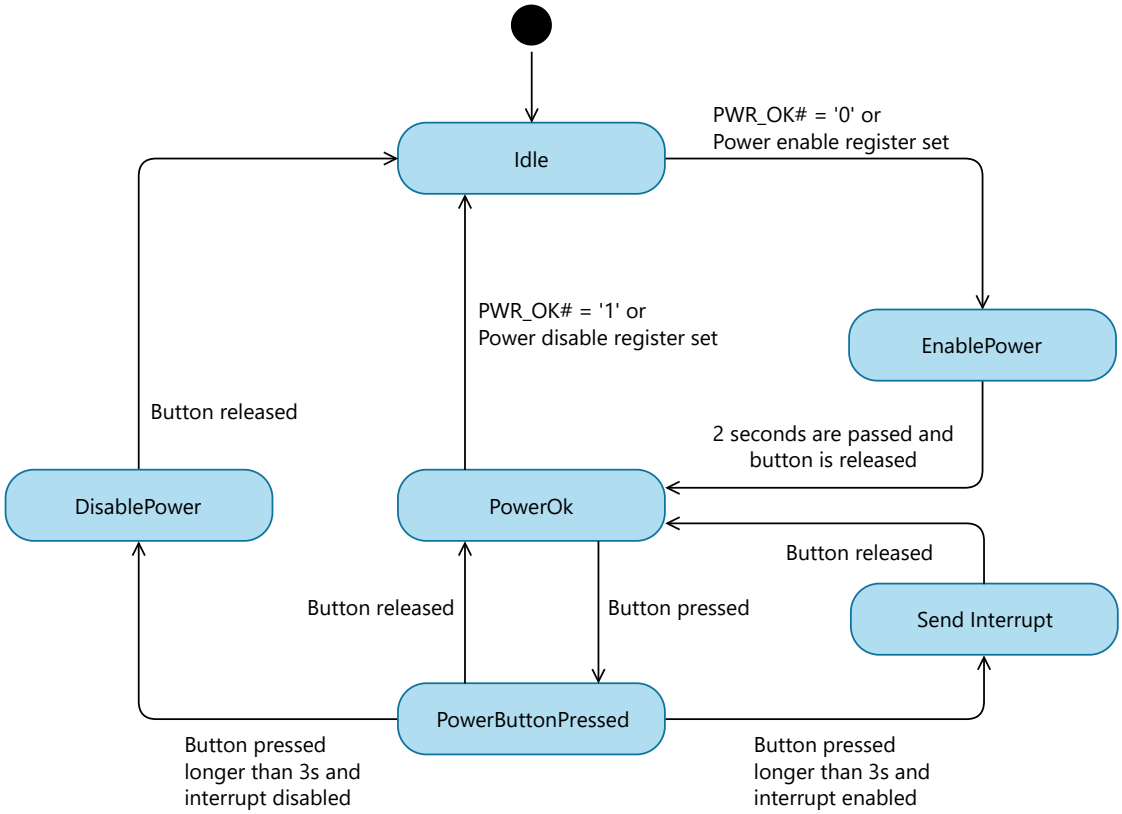


*Figure 2: Board Power Control FSM*

## 1.3 System Controller Config

The System Controller configuration unit generates the internal reset signal for the System Controller logic. This reset signal becomes active after powering the board and it is deactivated after 10ms. Before the reset is deactivated, the values of some specific pins are sampled and stored. Table 1 shows a list of all pins latched at startup and their usage. The stored value of each of these pins can be read using the registers listed in table 1.

| Pin | Register | Function |
|---|---|---|
| VMON_SEL_SCMODE0# | ScStatus (0x01) bit 0 | JTAG select (0=Xilinx, 1=Intel), Intel JTAG currently not supported |
| IOE_SEL_SCMODE1# | ScStatus (0x01) bit 1 | Default output value for USB_MODE pin |
| FTDI_BCBUS1_DONE_HWID0# | HwId (0x28) bit 0 | Hardware ID 0 |
| FTDI_BCBUS2_SRSTRDY#_HWID1# | HwId (0x28) bit 1 | Hardware ID 1 |
| FTDI_BCBUS3_PORLOAD#_HWID2# | HwId (0x28) bit 2 | Hardware ID 2 |
| FTDI_BCBUS0_LED_HWID3# | HwId (0x28) bit 3 | Hardware ID 3 |
| USB_SEL | ScStatus (0x01) bit 2 | Default output value for RSVD_BOOTMODE1 pin |

*Table 1: Pins latched at startup*

The configuration unit also controls the System Controller LED (see section 1.4) and generates a pulse to trigger the initialization of the registerbank depending on the detected hardware ID.

## 1.4 LEDs

### 1.4.1 System Controller LED

The System Controller LED is constantly blinking at an interval of one second. The brightness is reduced when the power good signal (pin PWR_OK#) is inactive. The blinking frequency is twice as fast while the module is held in reset (pin SRST#_RDY# = low or POR#_LOAD# = low).

### 1.4.2 UART LEDs

The RX and TX LEDs are flashing when activity on the UART bus is detected. For better visibility, each of the LEDs are enabled for at least 100ms after the UART event.

### 1.4.3 CPU LED

The CPU LED can be manually enabled by ScConfig register (address 0x00).

## 1.5 Action

The action unit allows to pull following signals low for 10ms:

- FTDI_RESET#
- PWR_EN
- POR#_LOAD#
- SRST#_RDY#

This functionality is controlled by the Action register (address 0x0A). Writing 1 to one of these bits triggers an action to pull the corresponding signal low for 10 milliseconds. After the 10 milliseconds passed, the signals are released to high impedance state. All signals are independent of each other. They can be triggered simultaneously or one by one.

## 1.6 Interrupt

An edge triggered interrupt can be generated by following sources:

| Interrupt Source | Trigger |
|---|---|
| VBUS_DETECT | Rising or falling edge on pin VBUS_DETECT |
| PCIE_PERST# | Rising or falling edge on pin PCIE_PERST# |
| PCIE_WAKE# | Rising or falling edge on pin PCIE_WAKE# |
| SDCARD_CD# | Rising or falling edge on pin SDCARD_CD# |
| MINI_PERST# | Rising or falling edge on pin MINI_PERST# |
| MINI_WAKE# | Rising or falling edge on pin MINI_WAKE# |
| Power button | Power button is pressed longer than configured time in BoardPowerButtonTime register (address 0x04) |

*Table 2: Possible interrupt sources*

The active edge can be selected by InterruptEdge register (address 0x0D) for all interrupts except power button interrupt. The power button interrupt generation is explained in detail in section 1.2.

When an interrupt occurs, the I2C_INT# pin is driven low until the interrupt flag is cleared. The interrupt flags are cleared by writing any value to the InterruptStatus register (address 0x0B). Note that all flags are

cleared simultaneously, it is not possible to clear single flags.

As alternative to the I2C_INT# pin, the interrupt signal can be routed to following pins, controlled by Gpio2Mux0 register (address 0x23) and Gpio2Mux1 register (address 0x24):

- SIO1_CPULED#
- SIO0_SCINT#

## 1.7 GPIO

Some pins of the System Controller can be used as GPIO. The GPIO functionality can be controlled using three registers: enable, write and read.

If the GPIO pin is used as input, the enable register needs to be set to 0, to disable the output driver. The signal value can be read using the read register. By default, all GPIO pins are configured as inputs.

If the GPIO pin is used as output, the enable register needs to be set to 1. The output value of the pin can be set by the write register.

## 1.8 I2C

The System Controller's internal I2C bus is always connected to the I2C bus of the SoC/FPGA. Optionally it is connected to the I2C bus on the PCIe edge connector when the pin EIO_EN# is driven low.
Only standard mode is supported with a transfer speed of 100kHz. There is no arbitration logic for multi master and clock stretching is not supported.

### 1.8.1 I2C Register Bank

The I2C register bank is located at bus address 0x0D. Figure 3 shows how to access the register bank by I2C. All bytes are transfered MSB first.

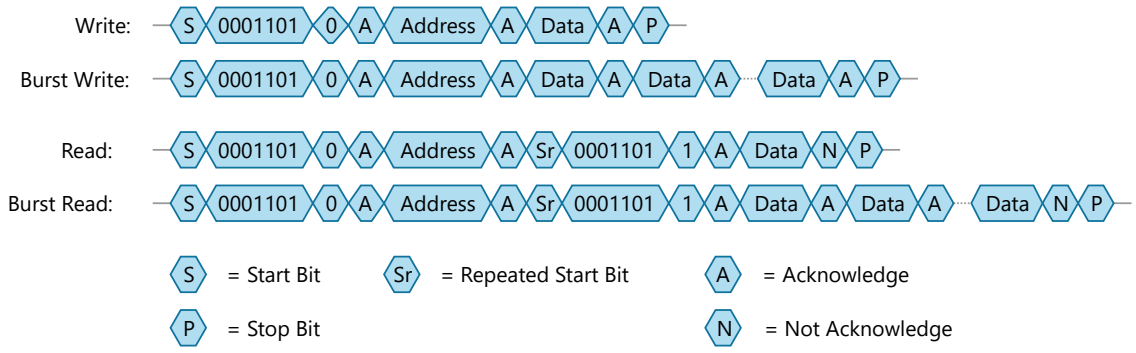


*Figure 3: Register Bank Access by I2C bus*

For consecutive read and write transactions, the address is incremented after each written/read byte. To test read/write access Register Debug (address 0x07) can be used, because its value is not used internally.

### 1.8.2 I2C Master

The I2C master is only active while the FPGA/SoC is in reset (POR#_LOAD# or SRST#_RDY# asserted). It is used by the clock generator initialization unit to configure the clock generator device every time after pin PWR_OK# is driven low.

### 1.8.3 I2C Hub

The I2C hub connects the EIO I2C bus to the regular I2C bus. It is enabled when EIO_EN# pin is driven low.

The logic detects which bus starts a transaction and switches the direction of the pins to allow a proper communication. The internal state machine always keeps track of the current transmitted bit and switches the direction of the pins depending on which bus the master is located or depending on if a acknowledge bit or a read or write byte is transmitted.

## 1.9 FTDI

Pin USB_SEL selects to which USB connector the FTDI USB port is connected. This signal is set by DIP switch USB_SEL. It can be manually overridden by UsbGpioEn (address 0x19) and UsbGpioWrite (address 0x1B) registers.

The signals from the FTDI FT2232 can be multiplexed to support multiple serial protocols. The selection is done using the FTDI driven signals FTDI_BCBUS6_MODE1 (Mode 1) and FTDI_BCBUS5_MODE0 (Mode 0). Following modes are supported:

| Mode 1 | Mode 0 | FTDI A | FTDI B |
|---|---|---|---|
| 0 | 0 | UART | Passive serial/Slave serial FPGA configuration |
| 0 | 1 | UART | I2C |
| 1 | 0 | UART | SPI for serial flash programming |
| 1 | 1 | JTAG | UART |

*Table 3: FTDI Mux modes*

### 1.9.1 Passive serial/Slave Serial Configuration

This mode is used to configure the FPGA logic of the inserted module in passive serial/slave serial mode using Enclustra MCT.

### 1.9.2 SPI for Serial Flash Programming

This mode is used to program the QSPI flash of the inserted module. Programming the flash is done using Enclustra MCT.

### 1.9.3 I2C

The FTDI device can be used as I2C master. It is connected to the System Controller's internal I2C bus.

### 1.9.4 JTAG

The FTDI device can be used as Xilinx compatible JTAG adapter. The EEPROM of the FTDI device must be configured first using the Enclustra MCT tool, otherwise the Xilinx tools will not recognize the FTDI as programming adapter.

### 1.9.5 UART

The UART bus of the module is connected to either FTDI port A or port B depending on the selected mode (see section 1.9).

## 1.10 UART Hub

The UART hub connects the three available UART buses in a manner that the TX line of one UART bus drives the RX lines of the other two UART buses. The individual ports of the UART hub can be enabled or disabled by ScConfig (address 0x00) register. By default only the FTDI UART bus is connected to the module UART bus. If all UART buses are enabled simultaneously, it is up to the user to ensure that only one bus is active to avoid data collision.

Following table shows the signals connected to the UART hub:

| Port | TX pin | RX pin | Description |
|---|---|---|---|
| 0 | FTDI_ADBUS0 or FTDI_BDBUS0_SCTCK_TXD | FTDI_ADBUS1 or FTDI_BDBUS1_SCTDI_RXD | FTDI UART bus, always enabled. The connected signals depend on the configured FTDI mode. |
| 1 | UART_TX | UART_RX | UART bus, enabled by default |
| 2 | UART_TX_EXT | UART_RX_EXT | UART_EXT bus, disabled by default |

*Table 4: UART hub connection*

## 1.11 Clock Generator Initialization

Every time the PWR_OK# pin is driven low, the clock generator is initialized to route the input clock coming from the PCIe edge connector to the clock output 0 of the clock generator device. Before the initialization is started, the initialization unit waits until the I2C lines are stable high for at least 500us. Following settings are used to configure the SI5338 device:

| Step | Device Address | Address | Data | Description |
|---|---|---|---|---|
| 1 | 0x70 | 0x1C | 0x00 | Set Input 1/2 as reference clock source |
| 2 | 0x70 | 0x1D | 0xA0 | Set Input 1/2 as reference clock source |
| 3 | 0x70 | 0x1F | 0x22 | Set reference clock as output clock |
| 4 | 0x70 | 0x23 | 0x00 | Set output voltage to 3.3V |
| 5 | 0x70 | 0x24 | 0x07 | Set output driver to HCSL |

*Table 5: SI5338 Clock generator initialization*

## 1.12 Pin Multiplexer

Following pins can be multiplexed internally to switch between different functions:

- The PCIe EIO SPI pins can be connected to the SC0 to SC3 signals by configuring the EioGpioMux0 (address 0x14) and EioGpioMux1 (address 0x15) register. By default the SPI multiplexer is disabled and the pins are set to high impedance.
- The SIO0_SCINT# pin can be configured to output the PCIE_PERST# or Interrupt signal by Gpio2Mux0 (address 0x23) and Gpio2Mux1 (address 0x24) register. By default the SIO0_SCINT# pin outputs the value of PCIE_PERST#.
- The SIO1_CPULED# pin can be configured to output the Interrupt signal by Gpio2Mux0 register (address 0x23). By default the SIO1_CPULED# is in GPIO mode set to high impedance.

- The SIO2_SDCD# pin can be configured to output the SDCARD_CD# signal by Gpio2Mux0 register (address 0x23). By default the SIO2_SDCD# pin outputs the value of SDCARD_CD#.

## 1.13 USB

Host or device mode is selected by USB_ID pin. By default the signal is set by DIP switch USB_ID. The default value can be overridden by UsbGpioEn (address 0x19) and UsbGpioWrite (address 0x1B) registers.

By default USB_MODE is set to the startup value of pin IOE_SEL_SCMODE1#. The default value can be overridden by UsbGpioEn (address 0x19) and UsbGpioWrite (address 0x1B) registers.

In host mode (pin USB_ID is 0): PWR_EN_VBUS# is enabled (0) when USB_MODE is 0 and USBMOD_CPEN is 1 or USB_MODE is 1 to enable USB 5V VBUS output.

In device mode (pin USB_ID is 1): PWR_EN_VBUS# is enabled (0) when USB_MODE is 0 and VBUS_DETECT is 1 or USB_MODE is 1 to route 5V to the USB PHY for device detection.

VBUS_DETECT is routed to RSVD_CLKEXT_SC which is routed on the hardware further to the VBUS_DETECT signal of module.

## 1.14 Boot Mode

By default the BOOT_MODE0 pin is configured to high impedance and the value of BOOT_MODE0 signal is set by DIP switch BOOT_MODE0. BOOT_MODE0 pin can be pulled low by BootConfigEn (address 0x0E) register.

By default the RSVD_BOOTMODE1 pin outputs the startup value of USB_SEL signal which is set by DIP switch USB_SEL. The default value can be overridden by BootConfigEn (address 0x0E) and BootConfigWrite (address 0x10) registers.

## 1.15 Version

The build version is stored in the following four registers: VersionYearMsb, VersionYearLsb, VersionMonth, VersionDay (address 0x2B to 0x2E).

# 2 I2C Register Bank

## 2.1 Register Field Types

| Field Type | Description |
|---|---|
| R | Read only. This bit can only be modified by hardware, writing to this bit has no effect. |
| R0 | Read only (always reads as zero). Writing to this bit has no effect. |
| R1 | Read only (always reads as one). Writing to this bit has no effect. |
| RW | Standard read/write bit. This bit can be modified by hardware and software. |
| RW0 | Read only. This bit can only be set by hardware, writing any value to this bit clears it to zero. |

*Table 6: Register Field Types*

## 2.2 Register Overview

| Address | Name | Description |
|---|---|---|
| 0x00 | ScConfig | System Controller configuration |
| 0x01 | ScStatus | System Controller configuration readback |
| 0x02 | BoardPowerControl | Power control of base board |
| 0x03 | BoardPowerStatus | Power status of base board |
| 0x04 | BoardPowerButtonTime | Configures power button press time |
| 0x05 | ModulePowerControl | Power control for module |
| 0x06 | ModulePowerStatus | Module power status |
| 0x07 | Debug | Debug register |
| 0x08 | Error | Error register |
| 0x0A | Action | Configures action unit |
| 0x0B | InterruptStatus | Interrupt status |
| 0x0C | InterruptEn | Interrupt enable |
| 0x0D | InterruptEdge | Interrupt edge selection |
| 0x0E | BootConfigEn | Boot config pin tristate select |
| 0x0F | BootConfigRead | Boot config pin read value |
| 0x10 | BootConfigWrite | Boot config write value |
| 0x11 | MiscGpioEn | Miscellaneous GPIO tristate select |
| 0x12 | MiscGpioRead | Miscellaneous GPIO read value |
| 0x13 | MiscGpioWrite | Miscellaneous GPIO write value |
| 0x14 | EioGpioMux0 | EIO GPIO multiplexer select bit 0 |
| 0x15 | EioGpioMux1 | EIO GPIO multiplexer select bit 1 |
| 0x16 | EioGpioEn | EIO GPIO tristate select |
| 0x17 | EioGpioRead | EIO GPIO read value |
| 0x18 | EioGpioWrite | EIO GPIO write value |

*Continued on next page...*

| | | |
|---|---|---|
| 0x19 | UsbGpioEn | USB GPIO tristate select |
| 0x1A | UsbGpioRead | USB GPIO read value |
| 0x1B | UsbGpioWrite | USB GPIO write value |
| 0x1C | Gpo | GPO write value |
| 0x1D | Gpio0En | GPIO0 tristate select |
| 0x1E | Gpio0Read | GPIO0 read value |
| 0x1F | Gpio0Write | GPIO0 write value |
| 0x20 | Gpio1En | GPIO1 tristate select |
| 0x21 | Gpio1Read | GPIO1 read value |
| 0x22 | Gpio1Write | GPIO1 write value |
| 0x23 | Gpio2Mux0 | GPIO2 multiplexer select bit 0 |
| 0x24 | Gpio2Mux1 | GPIO2 multiplexer select bit 1 |
| 0x25 | Gpio2En | GPIO2 tristate select |
| 0x26 | Gpio2Read | GPIO2 read value |
| 0x27 | Gpio2Write | GPIO2 write value |
| 0x28 | HwId | Hardware ID of base board |
| 0x2B | VersionYearMsb | System Controller build year MSB |
| 0x2C | VersionYearLsb | System Controller build year LSB |
| 0x2D | VersionMonth | System Controller build month |
| 0x2E | VersionDay | System Controller build day |

*Table 7: I2C Register Bank Overview*

## 2.3 Detailed Register Description

### 2.3.1 System Controller Configuration Register (0x00)

General System Controller configuration.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | CpuLedOn | RW | 0x00 | Writing 1 to this register enables the CPU LED permanently |
| 1 | 1 | UartEn | RW | 0x01 | Writing 1 to this register enables the UART port |
| 2 | 1 | ExtUartEn | RW | 0x00 | Writing 1 to this register enables the EXT_UART port |
| 7:3 | 5 | Unused | R0 | 0x00 | - |

*Table 8: ScConfig Register (0x00)*

### 2.3.2 System Controller Status Register (0x01)

The configuration signals which are read and stored at startup can be seen in this register.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | ScMode0 | R | 0x00 | Inverted value of pin VMON_SEL_SCMODE0# at startup |

*Continued on next page...*

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | ScMode1 | R | 0x00 | Inverted value of pin IOE_SEL_SCMODE1# at startup |
| 2 | 1 | BootMode1 | R | 0x00 | Value of pin USB_SEL at startup |
| 7:3 | 5 | Unused | R0 | 0x00 | - |

*Table 9: ScStatus Register (0x01)*

### 2.3.3 Board Power Control Register (0x02)

This register controls power of the base board. Writing to this register triggers to enable or disable the board power. Read returns the last written value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | PwrDis | RW | 0x00 | Writing 1 to this register disables the power of the base board |
| 1 | 1 | PwrEn | RW | 0x00 | Writing 1 to this register enables the power of the base board |
| 7:2 | 6 | Unused | R0 | 0x00 | - |

*Table 10: BoardPowerControl Register (0x02)*

### 2.3.4 Board Power Status Register (0x03)

Base board power status register.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | ButtonState | R | 0x00 | Read value is 1 if the power button is pressed |
| 1 | 1 | PwrCtrl | R | 0x00 | Read value is 1 if the board power control FSM enabled the power |
| 7:2 | 6 | Unused | R0 | 0x00 | - |

*Table 11: BoardPowerStatus Register (0x03)*

### 2.3.5 Board Power-Button Time Register (0x04)

Configures the time the power button needs to be pressed until base board power is turned off or an interrupt is generated. The value is set to 3 seconds by default.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 7:0 | 8 | ButtonTime | RW | 0x3C | Value*50ms = button press time |

*Table 12: BoardPowerButtonTime Register (0x04)*

### 2.3.6 Module Power Control Register (0x05)

This register is used to control power for the attached module.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | ForceSepicEn | RW | 0x00 | Not used on this product |

*Continued on next page...*

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | UsbCPwrFsmEn | RW | 0x00 | Not used on this product |
| 2 | 1 | DisableModulePower | RW | 0x00 | A value of 1 disables the power to the module |
| 7:3 | 5 | Unused | R0 | 0x00 | - |

*Table 13: ModulePowerControl Register (0x05)*

### 2.3.7 Module Power Status Register (0x06)

Module power status register.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | PwrEn | R | 0x00 | Read value is 1 if module power is enabled (PWR_EN = 1) |
| 1 | 1 | PwrOk | R | 0x00 | Read value is 1 if module power is ok (PWR_OK# = 0) |
| 2 | 1 | SepicEnabled | R | 0x00 | Always 0 |
| 3 | 1 | UsbC500mA | R | 0x00 | Always 0 |
| 4 | 1 | UsbC1500mA | R | 0x00 | Always 0 |
| 5 | 1 | UsbC3000mA | R | 0x00 | Always 0 |
| 7:6 | 2 | Unused | R0 | 0x00 | - |

*Table 14: ModulePowerStatus Register (0x06)*

### 2.3.8 Debug Register (0x07)

This register is not used internally. It can be used for read/write access test. The written value is stored and returned on read.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 7:0 | 8 | Debug | RW | 0x00 | Debug |

*Table 15: Debug Register (0x07)*

### 2.3.9 Error Register (0x08)

Error register, contains read only error flags.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | ClockGenError | R | 0x00 | Clock generator initialization error |
| 1 | 1 | MgtMuxError | R | 0x00 | Always 0 |
| 2 | 1 | ClockGenRetry | R | 0x00 | Clock generator I2C acknowledge was missing |
| 3 | 1 | MgtMuxRetry | R | 0x00 | Always 0 |
| 7:4 | 4 | Unused | R0 | 0x00 | - |

*Table 16: Error Register (0x07)*

### 2.3.10 Action Register (0x0A)

Control register for action unit. Writing 1 to these bits triggers immediately an action for 10 milliseconds. Read returns the last written value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | FtdiReset | RW | 0x00 | Writing 1 to this bit resets the FTDI on the base board |
| 1 | 1 | PwrDis | RW | 0x00 | Writing 1 to this bit disables the module power |
| 2 | 1 | PorLoad | RW | 0x00 | Writing 1 to this bit resets the module by POR#_LOAD# pin |
| 3 | 1 | SrstRdy | RW | 0x00 | Writing 1 to this bit resets the module by SRST#_RDY# pin |
| 7:4 | 4 | Unused | R0 | 0x00 | - |

*Table 17: Action Register (0x0A)*

### 2.3.11 Interrupt Status Register (0x0B)

When an interrupt occurs, the corresponding flag in this register is set. Writing any value to this register clears all flags.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | PowerButton | RW0 | 0x00 | Set when a power button interrupt occurred |
| 1 | 1 | VBusDetect | RW0 | 0x00 | Set when an VBUS_DETECT interrupt occurred |
| 2 | 1 | PciePerstN | RW0 | 0x00 | Set when a PCIE_PERST# interrupt occurred |
| 3 | 1 | PcieWakeN | RW0 | 0x00 | Set when a PCIE_WAKE# interrupt occurred |
| 4 | 1 | SdCardCdN | RW0 | 0x00 | Set when a SDCARD_CD# interrupt occurred |
| 5 | 1 | Gpio1 | RW0 | 0x00 | Set when a MINI_PERST# interrupt occurred |
| 6 | 1 | Gpio2 | RW0 | 0x00 | Set when a MINI_WAKE# interrupt occurred |
| 7 | 1 | Unused | R0 | 0x00 | - |

*Table 18: InterruptStatus Register (0x0B)*

### 2.3.12 Interrupt Enable Register (0x0C)

Interrupt enable register. Writing 1 to these bits enables interrupt generation for the chosen signal.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | PowerButton | RW | 0x00 | Enables interrupt when power button is pressed longer than configured time |
| 1 | 1 | VBusDetect | RW | 0x00 | Enables interrupt when signal on pin VBUS_DETECT changes |
| 2 | 1 | PciePerstN | RW | 0x00 | Enables interrupt when signal on pin PCIE_PERST# changes |
| 3 | 1 | PcieWakeN | RW | 0x00 | Enables interrupt when signal on pin PCIE_WAKE# changes |
| 4 | 1 | SdCardCdN | RW | 0x00 | Enables interrupt when signal on pin SDCARD_CD# changes |
| 5 | 1 | Gpio1 | RW | 0x00 | Enables interrupt when signal on pin MINI_PERST# changes |
| 6 | 1 | Gpio2 | RW | 0x00 | Enables interrupt when signal on pin MINI_WAKE# changes |
| 7 | 1 | Unused | R0 | 0x00 | - |

*Table 19: InterruptEn Register (0x0C)*

### 2.3.13 Interrupt Edge Select Register (0x0D)

Selects the edge that generates an interrupt. 1 selects rising edge, 0 selects falling edge.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Unused | R0 | 0x00 | - |
| 1 | 1 | VBusDetect | RW | 0x00 | Selects the active edge to generate a VBUS_DETECT interrupt |
| 2 | 1 | PciePerstN | RW | 0x00 | Selects the active edge to generate a PCIE_PERST# interrupt |
| 3 | 1 | PcieWakeN | RW | 0x00 | Selects the active edge to generate a PCIE_WAKE# interrupt |
| 4 | 1 | SdCardCdN | RW | 0x00 | Selects the active edge to generate a SDCARD_CD# interrupt |
| 5 | 1 | Gpio1 | RW | 0x00 | Selects the active edge to generate a MINI_PERST# interrupt |
| 6 | 1 | Gpio2 | RW | 0x00 | Selects the active edge to generate a MINI_WAKE# interrupt |
| 7 | 1 | Unused | R0 | 0x00 | - |

*Table 20: InterruptEdge Register (0x0D)*

### 2.3.14 Boot Configuration Enable Register (0x0E)

Writing 1 to these bits allows to override the signal value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | BootMode1 | RW | 0x00 | Pin: RSVD_BOOTMODE1<br>0 = Output startup value of pin USB_SEL<br>1 = Output value is set by BootConfigWrite register (0x10) bit 0 |
| 1 | 1 | BootMode0 | RW | 0x00 | Pin: BOOT_MODE0<br>0 = High impedance<br>1 = Drive low |
| 2 | 1 | PorLoadN | RW | 0x00 | Pin: POR#_LOAD#<br>0 = High impedance<br>1 = Drive low |
| 3 | 1 | SrstRdyN | RW | 0x00 | Pin: SRST#_RDY#<br>0 = High impedance<br>1 = Drive low |
| 7:4 | 4 | Unused | R0 | 0x00 | - |

*Table 21: BootConfigEn Register (0x0E)*

### 2.3.15 Boot Configuration Read Register (0x0F)

This register contains the read values of the boot configuration pins.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | BootMode1 | R | 0x00 | Read value of pin RSVD_BOOTMODE1 |
| 1 | 1 | BootMode0 | R | 0x00 | Read value of pin BOOT_MODE0 |
| 2 | 1 | PorLoadN | R | 0x00 | Read value of pin POR#_LOAD# |
| 3 | 1 | SrstRdyN | R | 0x00 | Read value of pin SRST#_RDY# |
| 4 | 1 | FpgaDone | R | 0x00 | Read value of pin FPGA_DONE |
| 5 | 1 | SdCardCdN | R | 0x00 | Read value of pin SDCARD_CD# |

*Continued on next page...*

| | | | | | |
|---|---|---|---|---|---|
| 7:6 | 2 | Unused | R0 | 0x00 | - |

*Table 22: BootConfigRead Register (0x0F)*

### 2.3.16 Boot Configuration Write Register (0x10)

If the corresponding bit in BootConfigEn register (0x0E) is set, this register allows to drive the pin low or high depending on the selected value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | BootMode1 | RW | 0x00 | Output value of pin RSVD_BOOTMODE1 when bit 0 in BootConfigEn register (0x0E) is set |
| 7:1 | 7 | Unused | R0 | 0x00 | - |

*Table 23: BootConfigWrite Register (0x10)*

### 2.3.17 Miscellaneous GPIO Enable Register (0x11)

Setting these bits to 1 allows to override the pin values.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | PciePerstN | RW | 0x00 | Pin: PCIE_PERST#<br>0 = High impedance<br>1 = Output value is set by MiscGpioWrite register (0x13) bit 0<br>Note that PCIE_PERST# is set to high impedance if Gpio2Mux0 register (0x23) bit 0 is set, independent of the value of this bit. |
| 1 | 1 | PcieWakeN | RW | 0x00 | Pin: PCIE_WAKE#<br>0 = High impedance<br>1 = Output value is set by MiscGpioWrite register (0x13) bit 1 |
| 2 | 1 | ScMode0 | RW | 0x00 | Pin: VMON_SEL_SCMODE0#<br>0 = High impedance<br>1 = Output value is set by MiscGpioWrite register (0x13) bit 2 |
| 3 | 1 | ScMode1 | RW | 0x00 | Pin: IOE_SEL_SCMODE1#<br>0 = High impedance<br>1 = Output value is set by MiscGpioWrite register (0x13) bit 3 |
| 4 | 1 | FtdiReset | RW | 0x00 | Pin: FTDI_RESET#<br>0 = High impedance<br>1 = Output value is 0 |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 24: MiscGpioEn Register (0x11)*

### 2.3.18 Miscellaneous GPIO Read Register (0x12)

This register contains the read values of the GPIO pins.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | PciePerstN | R | 0x00 | Read value of pin PCIE_PERST# |
| 1 | 1 | PcieWakeN | R | 0x00 | Read value of pin PCIE_WAKE# |
| 2 | 1 | ScMode0 | R | 0x00 | Read value of pin VMON_SEL_SCMODE0# |

*Continued on next page...*

| | | | | | |
|---|---|---|---|---|---|
| 3 | 1 | ScMode1 | R | 0x00 | Read value of pin IOE_SEL_SCMODE1# |
| 4 | 1 | FtdiReset | R | 0x00 | Read value of pin FTDI_RESET# |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 25: MiscGpioRead Register (0x12)*

### 2.3.19 Miscellaneous GPIO Write Register (0x13)

If the corresponding bit in MiscGpioEn register (0x11) is set, this register allows to drive the GPIO pin low or high depending on the selected value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | PciePerstN | RW | 0x00 | Output value of pin PCIE_PERST# when bit 0 in MiscGpioEn register (0x11) is set and bit 0 of Gpio2Mux0 register (0x23) is cleared |
| 1 | 1 | PcieWakeN | RW | 0x00 | Output value of pin PCIE_WAKE# when bit 1 in MiscGpioEn register (0x11) is set |
| 2 | 1 | ScMode0 | RW | 0x00 | Output value of pin VMON_SEL_SCMODE0# when bit 2 in MiscGpioEn register (0x11) is set |
| 3 | 1 | ScMode1 | RW | 0x00 | Output value of pin IOE_SEL_SCMODE1# when bit 3 in MiscGpioEn register (0x11) is set |
| 7:4 | 4 | Unused | R0 | 0x00 | - |

*Table 26: MiscGpioWrite Register (0x13)*

### 2.3.20 EIO GPIO Multiplexer 0 Register (0x14)

This register is not used on this product. Do not write any value other than zero to this register.

This register is used together with EioGpioMux1 register (0x15) to control the EIO SPI multiplexer. It allows to route the EIO SPI signals to the FPGA pins.

Multiplexer select bit 1 = 0 and bit 0 = 0: EIO SPI pin is used as GPIO
Multiplexer select bit 1 = 1 and bit 0 = 0: EIO SPI pin is connected to FPGA (EIO is SPI slave)
Multiplexer select bit 0 = 1: EIO SPI pin is connected to FPGA (EIO is SPI master)

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Unused | R0 | 0x00 | - |
| 1 | 1 | EioClkMux[0] | RW | 0x00 | Multiplexer select bit 0 for pin EIO0_CLK |

| Bit 1 | Bit 0 | Description |
|---|---|---|
| 0 | 0 | GPIO |
| 0 | 1 | Connection EIO0_CLK → IOB_D16_SC0_DIP1# |
| 1 | 0 | Connection EIO0_CLK ← IOB_D16_SC0_DIP1# |
| 1 | 1 | Connection EIO0_CLK → IOB_D16_SC0_DIP1# |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 1 | EioMosiMux[0] | RW | 0x00 | Multiplexer select bit 0 for pin EIO2_MOSI |

| Bit 1 | Bit 0 | Description |
|---|---|---|
| 0 | 0 | GPIO |
| 0 | 1 | Connection EIO2_MOSI → IOB_D18_SC2_DIP3# |

*Continued on next page...*

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | 1 | 0 | Connection EIO2_MOSI ← IOB_D18_SC2_DIP3# |
| | | | | | 1 | 1 | Connection EIO2_MOSI → IOB_D18_SC2_DIP3# |
| 3 | 1 | EioMisoMux[0] | RW | 0x00 | Multiplexer select bit 0 for pin EIO1_MISO | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 1 | Connection IOB_D19_SC3_DIP4# → EIO1_MISO |
| | | | | | 1 | 0 | Connection IOB_D19_SC3_DIP4# ← EIO1_MISO |
| | | | | | 1 | 1 | Connection IOB_D19_SC3_DIP4# → EIO1_MISO |
| 4 | 1 | EioCsNMux[0] | RW | 0x00 | Multiplexer select bit 0 for pin EIO3_CS# | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 1 | Connection EIO3_CS# → IOB_D17_SC1_DIP2# |
| | | | | | 1 | 0 | Connection EIO3_CS# ← IOB_D17_SC1_DIP2# |
| | | | | | 1 | 1 | Connection EIO3_CS# → IOB_D17_SC1_DIP2# |
| 7:5 | 3 | Unused | R0 | 0x00 | - | | |

*Table 27: EioGpioMux0 Register (0x14)*

### 2.3.21 EIO GPIO Multiplexer 1 Register (0x15)

This register is not used on this product. Do not write any value other than zero to this register.

This register is used together with EioGpioMux0 register (0x14) to control the EIO SPI multiplexer. It allows to route the EIO SPI signals to the FPGA pins.

Multiplexer select bit 1 = 0 and bit 0 = 0: EIO SPI pin is used as GPIO
Multiplexer select bit 1 = 1 and bit 0 = 0: EIO SPI pin is connected to FPGA (EIO is SPI slave)
Multiplexer select bit 0 = 1: EIO SPI pin is connected to FPGA (EIO is SPI master)

| Bit | Size | Field Name | Type | Reset | Description | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | Unused | R0 | 0x00 | - | | |
| 1 | 1 | EioClkMux[1] | RW | 0x00 | Multiplexer select bit 1 for pin EIO0_CLK | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 1 | Connection EIO0_CLK → IOB_D16_SC0_DIP1# |
| | | | | | 1 | 0 | Connection EIO0_CLK ← IOB_D16_SC0_DIP1# |
| | | | | | 1 | 1 | Connection EIO0_CLK → IOB_D16_SC0_DIP1# |
| 2 | 1 | EioMosiMux[1] | RW | 0x00 | Multiplexer select bit 1 for pin EIO2_MOSI | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 1 | Connection EIO2_MOSI → IOB_D18_SC2_DIP3# |
| | | | | | 1 | 0 | Connection EIO2_MOSI ← IOB_D18_SC2_DIP3# |

*Continued on next page...*

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | 1 | 1 | Connection EIO2_MOSI → IOB_D18_SC2_DIP3# |
| 3 | 1 | EioMisoMux[1] | RW | 0x00 | Multiplexer select bit 1 for pin EIO1_MISO | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 1 | Connection IOB_D19_SC3_DIP4# → EIO1_MISO |
| | | | | | 1 | 0 | Connection IOB_D19_SC3_DIP4# ← EIO1_MISO |
| | | | | | 1 | 1 | Connection IOB_D19_SC3_DIP4# → EIO1_MISO |
| 4 | 1 | EioCsNMux[1] | RW | 0x00 | Multiplexer select bit 1 for pin EIO3_CS# | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 1 | Connection EIO3_CS# → IOB_D17_SC1_DIP2# |
| | | | | | 1 | 0 | Connection EIO3_CS# ← IOB_D17_SC1_DIP2# |
| | | | | | 1 | 1 | Connection EIO3_CS# → IOB_D17_SC1_DIP2# |
| 7:5 | 3 | Unused | R0 | 0x00 | - | | |

*Table 28: EioGpioMux1 Register (0x15)*

### 2.3.22 EIO GPIO Enable Register (0x16)

Setting these bits to 1 allows to override the pin values if the EIO GPIO multiplexer (register 0x14/0x15) is configured to select GPIO functionality.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | EioEnN | RW | 0x00 | Pin: EIO_EN#<br>0 = High impedance<br>1 = Output value is set by EioGpioWrite register (0x18) bit 0 |
| 1 | 1 | EioClk | RW | 0x00 | Pin: EIO0_CLK<br>0 = High impedance<br>1 = Output value is set by EioGpioWrite register (0x18) bit 1 |
| 2 | 1 | EioMosi | RW | 0x00 | Pin: EIO2_MOSI<br>0 = High impedance<br>1 = Output value is set by EioGpioWrite register (0x18) bit 2 |
| 3 | 1 | EioMiso | RW | 0x00 | Pin: EIO1_MISO<br>0 = High impedance<br>1 = Output value is set by EioGpioWrite register (0x18) bit 3 |
| 4 | 1 | EioCsN | RW | 0x00 | Pin: EIO3_CS#<br>0 = High impedance<br>1 = Output value is set by EioGpioWrite register (0x18) bit 4 |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 29: EioGpioEn Register (0x16)*

### 2.3.23 EIO GPIO Read Register (0x17)

This register contains the read values of the GPIO pins.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|

*Continued on next page...*

| | | | | | |
|---|---|---|---|---|---|
| 0 | 1 | EioEnN | R | 0x00 | Read value of pin EIO_EN# |
| 1 | 1 | EioClk | R | 0x00 | Read value of pin EIO0_CLK |
| 2 | 1 | EioMosi | R | 0x00 | Read value of pin EIO2_MOSI |
| 3 | 1 | EioMiso | R | 0x00 | Read value of pin EIO1_MISO |
| 4 | 1 | EioCsN | R | 0x00 | Read value of pin EIO3_CS# |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 30: EioGpioRead Register (0x17)*

### 2.3.24 EIO GPIO Write Register (0x18)

If the corresponding bit in EioGpioEn register (0x16) is set and the EIO GPIO Multiplexer (register 0x14/0x15) is configured to select GPIO functionality, this register allows to drive the GPIO pin low or high depending on the selected value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | EioEnN | RW | 0x00 | Output value of pin EIO_EN# when bit 0 in EioGpioEn register (0x16) is set |
| 1 | 1 | EioClk | RW | 0x00 | Output value of pin EIO0_CLK when bit 1 in EioGpioEn register (0x16) is set |
| 2 | 1 | EioMosi | RW | 0x00 | Output value of pin EIO2_MOSI when bit 2 in EioGpioEn register (0x16) is set |
| 3 | 1 | EioMiso | RW | 0x00 | Output value of pin EIO1_MISO when bit 3 in EioGpioEn register (0x16) is set |
| 4 | 1 | EioCsN | RW | 0x00 | Output value of pin EIO3_CS# when bit 4 in EioGpioEn register (0x16) is set |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 31: EioGpioWrite Register (0x18)*

### 2.3.25 USB GPIO Enable Register (0x19)

Setting these bits to 1 allows to override the pin values.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | UsbSel | RW | 0x00 | Pin: USB_SEL<br>0 = High impedance<br>1 = Output value is set by UsbGpioWrite register (0x1B) bit 0 |
| 1 | 1 | UsbId | RW | 0x00 | Pin: USB_ID<br>0 = High impedance<br>1 = Output value is set by UsbGpioWrite register (0x1B) bit 1 |
| 2 | 1 | UsbMode | RW | 0x01 | Pin: USB_MODE<br>0 = High impedance<br>1 = Output value is set by UsbGpioWrite register (0x1B) bit 2 |
| 7:3 | 5 | Unused | R0 | 0x00 | - |

*Table 32: UsbGpioEn Register (0x19)*

### 2.3.26 USB GPIO Read Register (0x1A)

This register contains the read values of the GPIO pins.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | UsbSel | R | 0x00 | Read value of pin USB_SEL |
| 1 | 1 | UsbId | R | 0x00 | Read value of pin USB_ID |
| 2 | 1 | UsbMode | R | 0x00 | Read value of pin USB_MODE |
| 3 | 1 | UsbVbusDetect | R | 0x00 | Read value of pin VBUS_DETECT |
| 4 | 1 | UsbModCpEN | R | 0x00 | Read value of pin USBMOD_CPEN |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 33: UsbGpioRead Register (0x1A)*

### 2.3.27 USB GPIO Write Register (0x1B)

If the corresponding bit in UsbGpioEn register (0x19) is set, this register allows to drive the GPIO pin low or high depending on the selected value.
The reset value of UsbMode bit is the startup value of pin IOE_SEL_SCMODE1#.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | UsbSel | RW | 0x00 | Output value of pin USB_SEL when bit 0 in UsbGpioEn register (0x19) is set |
| 1 | 1 | UsbId | RW | 0x00 | Output value of pin USB_ID when bit 1 in UsbGpioEn register (0x19) is set |
| 2 | 1 | UsbMode | RW | 0x00 | Output value of pin USB_MODE when bit 2 in UsbGpioEn register (0x19) is set |
| 7:3 | 5 | Unused | R0 | 0x00 | - |

*Table 34: UsbGpioWrite Register (0x1B)*

### 2.3.28 GPO Register (0x1C)

This register allows to drive the GPO pin low or high depending on the selected value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | GpoUsbC | RW | 0x00 | Output value of pin I2C_EN_FMC |
| 1 | 1 | GpoClkSel | RW | 0x00 | Output value of pin CLK_SEL |
| 2 | 1 | Gpo0 | RW | 0x00 | Output value of pin I2C_EN_IOB |
| 3 | 1 | Gpo1 | RW | 0x00 | Output value of pin I2C_EN_IOA |
| 7:4 | 4 | Unused | R0 | 0x00 | - |

*Table 35: Gpo Register (0x1C)*

### 2.3.29 GPIO 0 Enable Register (0x1D)

Setting these bits to 1 allows to override the pin values.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio0Sc0 | RW | 0x00 | Pin: IOB_D16_SC0_DIP1#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 0<br>Note that the GPIO functionality of pin IOB_D16_SC0_DIP1# is disabled if bit 1 of EioGpioMuxZero register (0x14) or EioGpioMuxOne register (0x15) is set. |
| 1 | 1 | Gpio0Sc1 | RW | 0x00 | Pin: IOB_D17_SC1_DIP2#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 1<br>Note that the GPIO functionality of pin IOB_D17_SC1_DIP2# is disabled if bit 4 of EioGpioMuxZero register (0x14) or EioGpioMuxOne register (0x15) is set. |
| 2 | 1 | Gpio0Sc2 | RW | 0x00 | Pin: IOB_D18_SC2_DIP3#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 2<br>Note that the GPIO functionality of pin IOB_D18_SC2_DIP3# is disabled if bit 2 of EioGpioMuxZero register (0x14) or EioGpioMuxOne register (0x15) is set. |
| 3 | 1 | Gpio0Sc3 | RW | 0x00 | Pin: IOB_D19_SC3_DIP4#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 3<br>Note that the GPIO functionality of pin IOB_D19_SC3_DIP4# is disabled if bit 3 of EioGpioMuxZero register (0x14) or EioGpioMuxOne register (0x15) is set. |
| 4 | 1 | Gpio0Sc4 | RW | 0x00 | Pin: IOB_D20_SC4_BTN0#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 4 |
| 5 | 1 | Gpio0Sc5 | RW | 0x00 | Pin: IOB_D21_SC5_BTN1#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 5 |
| 6 | 1 | Gpio0Sc6 | RW | 0x00 | Pin: IOB_D22_SC6_BTN2#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 6 |
| 7 | 1 | Gpio0Sc7 | RW | 0x00 | Pin: IOB_D23_SC7_BTN3#<br>0 = High impedance<br>1 = Output value is set by Gpio0Write register (0x1F) bit 7 |

*Table 36: Gpio0En Register (0x1D)*

### 2.3.30 GPIO 0 Read Register (0x1E)

This register contains the read values of the GPIO pins.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio0Sc0 | R | 0x00 | Read value of pin IOB_D16_SC0_DIP1# |
| 1 | 1 | Gpio0Sc1 | R | 0x00 | Read value of pin IOB_D17_SC1_DIP2# |
| 2 | 1 | Gpio0Sc2 | R | 0x00 | Read value of pin IOB_D18_SC2_DIP3# |
| 3 | 1 | Gpio0Sc3 | R | 0x00 | Read value of pin IOB_D19_SC3_DIP4# |
| 4 | 1 | Gpio0Sc4 | R | 0x00 | Read value of pin IOB_D20_SC4_BTN0# |
| 5 | 1 | Gpio0Sc5 | R | 0x00 | Read value of pin IOB_D21_SC5_BTN1# |
| 6 | 1 | Gpio0Sc6 | R | 0x00 | Read value of pin IOB_D22_SC6_BTN2# |
| 7 | 1 | Gpio0Sc7 | R | 0x00 | Read value of pin IOB_D23_SC7_BTN3# |

*Table 37: Gpio0Read Register (0x1E)*

### 2.3.31 GPIO 0 Write Register (0x1F)

If the corresponding bit in Gpio0En register (0x1D) is set, this register allows to drive the GPIO pin low or high depending on the selected value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio0Sc0 | RW | 0x00 | Output value of pin IOB_D16_SC0_DIP1# when bit 0 in Gpio0En register (0x1D) is set and bit 1 of following registers is cleared: EioGpioMuxZero (0x14) and EioGpioMuxOne (0x15) |
| 1 | 1 | Gpio0Sc1 | RW | 0x00 | Output value of pin IOB_D17_SC1_DIP2# when bit 1 in Gpio0En register (0x1D) is set and bit 4 of following registers is cleared: EioGpioMuxZero (0x14) and EioGpioMuxOne (0x15) |
| 2 | 1 | Gpio0Sc2 | RW | 0x00 | Output value of pin IOB_D18_SC2_DIP3# when bit 2 in Gpio0En register (0x1D) is set and bit 2 of following registers is cleared: EioGpioMuxZero (0x14) and EioGpioMuxOne (0x15) |
| 3 | 1 | Gpio0Sc3 | RW | 0x00 | Output value of pin IOB_D19_SC3_DIP4# when bit 3 in Gpio0En register (0x1D) is set and bit 3 of following registers is cleared: EioGpioMuxZero (0x14) and EioGpioMuxOne (0x15) |
| 4 | 1 | Gpio0Sc4 | RW | 0x00 | Output value of pin IOB_D20_SC4_BTN0# when bit 4 in Gpio0En register (0x1D) is set |
| 5 | 1 | Gpio0Sc5 | RW | 0x00 | Output value of pin IOB_D21_SC5_BTN1# when bit 5 in Gpio0En register (0x1D) is set |
| 6 | 1 | Gpio0Sc6 | RW | 0x00 | Output value of pin IOB_D22_SC6_BTN2# when bit 6 in Gpio0En register (0x1D) is set |
| 7 | 1 | Gpio0Sc7 | RW | 0x00 | Output value of pin IOB_D23_SC7_BTN3# when bit 7 in Gpio0En register (0x1D) is set |

*Table 38: Gpio0Write Register (0x1F)*

### 2.3.32 GPIO 1 Enable Register (0x20)

Setting these bits to 1 allows to override the pin values.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio1Gpio0 | RW | 0x00 | Pin: MINI_WDISABLE#<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 0 |
| 1 | 1 | Gpio1Gpio1 | RW | 0x00 | Pin: MINI_PERST#<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 1 |
| 2 | 1 | Gpio1Gpio2 | RW | 0x00 | Pin: MINI_WAKE#<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 2 |
| 3 | 1 | Gpio1Gpio3 | RW | 0x00 | Pin: IOE_D4<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 3 |
| 4 | 1 | Gpio1Gpio4 | RW | 0x00 | Pin: IOE_D7<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 4 |
| 5 | 1 | Gpio1Gpio5 | RW | 0x00 | Pin: IOE_D5<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 5 |
| 6 | 1 | Gpio1Gpio6 | RW | 0x00 | Pin: RSVD_CLKEXT_SC<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 6 |
| 7 | 1 | Gpio1Gpio7 | RW | 0x00 | Pin: IOE_D6<br>0 = High impedance<br>1 = Output value is set by Gpio1Write register (0x22) bit 7 |

*Table 39: Gpio1En Register (0x20)*

### 2.3.33 GPIO 1 Read Register (0x21)

This register contains the read values of the GPIO pins.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio1Gpio0 | R | 0x00 | Read value of pin MINI_WDISABLE# |
| 1 | 1 | Gpio1Gpio1 | R | 0x00 | Read value of pin MINI_PERST# |
| 2 | 1 | Gpio1Gpio2 | R | 0x00 | Read value of pin MINI_WAKE# |
| 3 | 1 | Gpio1Gpio3 | R | 0x00 | Read value of pin IOE_D4 |
| 4 | 1 | Gpio1Gpio4 | R | 0x00 | Read value of pin IOE_D7 |
| 5 | 1 | Gpio1Gpio5 | R | 0x00 | Read value of pin IOE_D5 |
| 6 | 1 | Gpio1Gpio6 | R | 0x00 | Read value of pin RSVD_CLKEXT_SC |
| 7 | 1 | Gpio1Gpio7 | R | 0x00 | Read value of pin IOE_D6 |

*Table 40: Gpio1Read Register (0x21)*

### 2.3.34 GPIO 1 Write Register (0x22)

If the corresponding bit in Gpio1En register (0x20) is set, this register allows to drive the GPIO pin low or high depending on the selected value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio1Gpio0 | RW | 0x00 | Output value of pin MINI_WDISABLE# when bit 0 in Gpio1En register (0x20) is set |
| 1 | 1 | Gpio1Gpio1 | RW | 0x00 | Output value of pin MINI_PERST# when bit 1 in Gpio1En register (0x20) is set |
| 2 | 1 | Gpio1Gpio2 | RW | 0x00 | Output value of pin MINI_WAKE# when bit 2 in Gpio1En register (0x20) is set |
| 3 | 1 | Gpio1Gpio3 | RW | 0x00 | Output value of pin IOE_D4 when bit 3 in Gpio1En register (0x20) is set |
| 4 | 1 | Gpio1Gpio4 | RW | 0x00 | Output value of pin IOE_D7 when bit 4 in Gpio1En register (0x20) is set |
| 5 | 1 | Gpio1Gpio5 | RW | 0x00 | Output value of pin IOE_D5 when bit 5 in Gpio1En register (0x20) is set |
| 6 | 1 | Gpio1Gpio6 | RW | 0x00 | Output value of pin RSVD_CLKEXT_SC when bit 6 in Gpio1En register (0x20) is set |
| 7 | 1 | Gpio1Gpio7 | RW | 0x00 | Output value of pin IOE_D6 when bit 7 in Gpio1En register (0x20) is set |

*Table 41: Gpio1Write Register (0x22)*

### 2.3.35 GPIO 2 Multiplexer 0 Register (0x23)

Some GPIO 2 pins can be used for other purposes than GPIO. This register in combination with Gpio2Mux1 register (0x24) is used to control the pin multiplexer.

| Bit | Size | Field Name | Type | Reset | Description | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | Gpio2Mux0[0] | RW | 0x01 | Multiplexer select bit 0 for pin SIO0_SCINT# | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 1 | Connection SIO0_SCINT# ← PCIE_PERST# |
| | | | | | 1 | 1 | Connection SIO0_SCINT# ← PCIE_PERST# |

*Continued on next page...*

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | 1 | 0 | Interrupt output |
| 1 | 1 | Gpio2Mux1 | RW | 0x00 | Multiplexer select for pin SIO1_CPULED#<br>0 = GPIO<br>1 = Interrupt output | | |
| 2 | 1 | Gpio2Mux2 | RW | 0x01 | Multiplexer select for pin SIO2_SDCD#<br>0 = GPIO<br>1 = Connection SIO2_SDCD# ← SDCARD_CD# | | |
| 7:3 | 5 | Unused | R0 | 0x00 | - | | |

*Table 42: Gpio2Mux0 Register (0x23)*

## 2.3.36 GPIO 2 Multiplexer 1 Register (0x24)

Some GPIO 2 pins can be used for other purposes than GPIO. This register in combination with Gpio2Mux0 register (0x23) is used to control the pin multiplexer.

| Bit | Size | Field Name | Type | Reset | Description | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | Gpio2Mux0[1] | RW | 0x00 | Multiplexer select bit 1 for pin SIO0_SCINT# | | |
| | | | | | **Bit 1** | **Bit 0** | **Description** |
| | | | | | 0 | 0 | GPIO |
| | | | | | 0 | 0 | Connection SIO0_SCINT# ← PCIE_PERST# |
| | | | | | 1 | 0 | Connection SIO0_SCINT# ← PCIE_PERST# |
| | | | | | 1 | 0 | Interrupt output |
| 7:1 | 7 | Unused | R0 | 0x00 | - | | |

*Table 43: Gpio2Mux1 Register (0x24)*

## 2.3.37 GPIO 2 Enable Register (0x25)

If the GPIO 2 multiplexer (register 0x23/0x24) is configured to select GPIO functionality, setting these bits to 1 allows to override the pin values.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio2Sio0 | RW | 0x00 | Pin: SIO0_SCINT#<br>0 = High impedance<br>1 = Output value is set by Gpio2Write register (0x27) bit 0 |
| 1 | 1 | Gpio2Sio1 | RW | 0x00 | Pin: SIO1_CPULED#<br>0 = High impedance<br>1 = Output value is set by Gpio2Write register (0x27) bit 1 |
| 2 | 1 | Gpio2Sio2 | RW | 0x00 | Pin: SIO2_SDCD#<br>0 = High impedance<br>1 = Output value is set by Gpio2Write register (0x27) bit 2 |
| 3 | 1 | Gpio2Sio3 | RW | 0x00 | Pin: SIO3_SDIOSEL#<br>0 = High impedance<br>1 = Output value is set by Gpio2Write register (0x27) bit 3 |
| 4 | 1 | Gpio2Gpio8 | RW | 0x00 | Pin: SDIO_SEL<br>0 = High impedance<br>1 = Output value is set by Gpio2Write register (0x27) bit 4 |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 44: Gpio2En Register (0x25)*

### 2.3.38 GPIO 2 Read Register (0x26)

This register contains the read values of the GPIO pins.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio2Sio0 | R | 0x00 | Read value of pin SIO0_SCINT# |
| 1 | 1 | Gpio2Sio1 | R | 0x00 | Read value of pin SIO1_CPULED# |
| 2 | 1 | Gpio2Sio2 | R | 0x00 | Read value of pin SIO2_SDCD# |
| 3 | 1 | Gpio2Sio3 | R | 0x00 | Read value of pin SIO3_SDIOSEL# |
| 4 | 1 | Gpio2Gpio8 | R | 0x00 | Read value of pin SDIO_SEL |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 45: Gpio2Read Register (0x26)*

### 2.3.39 GPIO 2 Write Register (0x27)

If the corresponding bit in Gpio2En register (0x25) is set and the GPIO2 multiplexer (0x23/0x24) is configured to enable the GPIO functionality of the pin, this register allows to drive the GPIO pin low or high depending on the selected value.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 0 | 1 | Gpio2Sio0 | RW | 0x00 | Output value of pin SIO0_SCINT# when bit 0 in Gpio2En register (0x25) is set |
| 1 | 1 | Gpio2Sio1 | RW | 0x00 | Output value of pin SIO1_CPULED# when bit 1 in Gpio2En register (0x25) is set |
| 2 | 1 | Gpio2Sio2 | RW | 0x00 | Output value of pin SIO2_SDCD# when bit 2 in Gpio2En register (0x25) is set |
| 3 | 1 | Gpio2Sio3 | RW | 0x00 | Output value of pin SIO3_SDIOSEL# when bit 3 in Gpio2En register (0x25) is set |
| 4 | 1 | Gpio2Gpio8 | RW | 0x00 | Pin: SDIO_SEL<br>Output value of pin SDIO_SEL when bit 4 in Gpio2En register (0x25) is set |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 46: Gpio2Write Register (0x27)*

### 2.3.40 Hardware ID Register (0x28)

The hardware ID is read at startup. It is used to identify the board and to enable/disable board specific features.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 3:0 | 4 | HwId | R | 0x0B | Hardware ID read at startup |
| 4 | 1 | Reserved | R1 | 0x01 | - |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 47: HwId Register (0x28)*

### 2.3.41 Version Year MSB Register (0x2B)

The build version is stored in four registers. This register contains the most significant bits of the build year.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 2:0 | 3 | VersionYearMsb | R | 0x00 | Version Year MSB (Bit 10:8) |
| 7:3 | 5 | Unused | R0 | 0x00 | - |

*Table 48: VersionYearMsb Register (0x2B)*

### 2.3.42 Version Year LSB Register (0x2C)

The build version is stored in four registers. This register contains the least significant bits of the build year.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 7:0 | 8 | VersionYearLsb | R | 0x00 | Version Year LSB (Bit 7:0) |

*Table 49: VersionYearLsb Register (0x2C)*

### 2.3.43 Version Month Register (0x2D)

The build version is stored in four registers. This register contains the build month.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 3:0 | 4 | VersionMonth | R | 0x00 | Version Month |
| 7:4 | 4 | Unused | R0 | 0x00 | - |

*Table 50: VersionMonth Register (0x2D)*

### 2.3.44 Version Day Register (0x2E)

The build version is stored in four registers. This register contains the build day.

| Bit | Size | Field Name | Type | Reset | Description |
|---|---|---|---|---|---|
| 4:0 | 5 | VersionDay | R | 0x00 | Version Day |
| 7:5 | 3 | Unused | R0 | 0x00 | - |

*Table 51: VersionDay Register (0x2E)*

# 3 Upgrade

## 3.1 Upgrade using Enclustra Module Configuration Tool

The Enclustra Module Configuration Tool (MCT) provides a user friendly way to upgrade the System Controller firmware. MCT as well as the most recent System Controller binary file can be downloaded from the Enclustra web page. This feature is supported only in MCT version 2.6 or later.

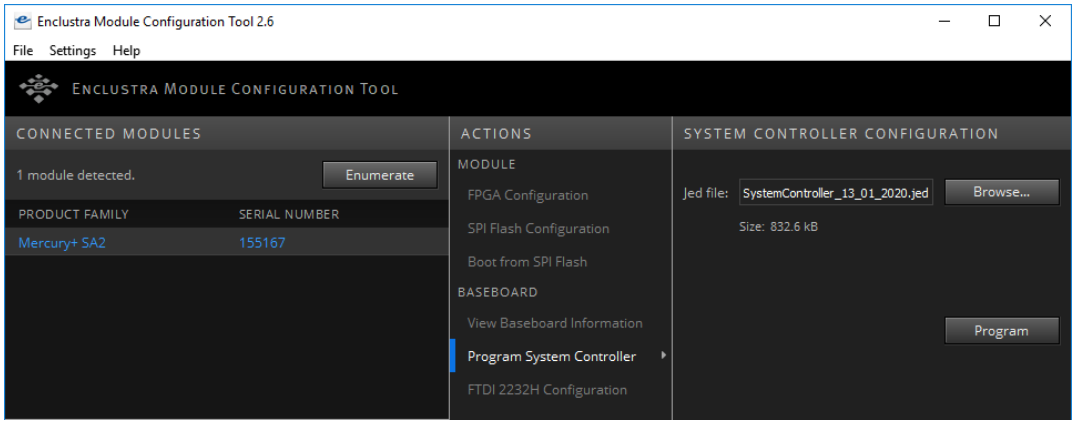


*Figure 4: Upgrade using MCT*

To perform an upgrade, following steps need to be executed:

- Equip the Mercury+ PE1 base board with an Enclustra module
- Power up the Mercury+ PE1 base board
- Connect the micro USB port labeled with USBUB of the Mercury+ PE1 base board to your PC
- Start the Enclustra Module Configuration Tool
- Click on "Enumerate", if no module was detected
- Select "Program System Controller" as action and enter the JEDEC programming file, as shown in figure 4
- Click "Program" and wait until the programming procedure has completed. This process takes about two minutes.
- The System Controller immediately reconfigures itself with the upgraded configuration. No restart or power cycle is needed.

## List of Figures



## List of Tables