

# PSoC® Creator™ Project Datasheet for BeoM\_main

Creation Time: 07/25/2020 16:22:33

User: ALA-PC\Ala

Project: BeoM\_main

**Tool: PSoC Creator 4.3** 

Cypress Semiconductor 198 Champion Court San Jose, CA 95134-1709 Phone (USA): 800.858.1810 Phone (Intl): 408.943.2600 http://www.cypress.com



#### Copyright

Copyright © 2020 Cypress Semiconductor Corporation. All rights reserved. Any design information or characteristics specifically provided by our customer or other third party inputs contained in this document are not intended to be claimed under Cypress's copyright.

#### **Trademarks**

PSoC and CapSense are registered trademarks of Cypress Semiconductor Corporation. PSoC Creator is a trademark of Cypress Semiconductor Corporation. All other trademarks or registered trademarks referenced herein are the property of their respective owners.

#### **Philips I2C Patent Rights**

Purchase of I2C components from Cypress or one of its sublicensed Associated Companies conveys a license under the Philips I2C Patent Rights to use these components in an I2C system, provided that the system conforms to the I2C Standard Specification as defined by Philips. As from October 1st, 2006 Philips Semiconductors has a new trade name, NXP Semiconductors.

#### Disclaimer

CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. While reasonable precautions have been taken, Cypress assumes no responsibility for any errors that may appear in this document. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of a Cypress product in a life support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

#### **Flash Code Protection**

Cypress products meet the specifications contained in their particular Cypress PSoC Datasheets. Cypress believes that its family of PSoC products is one of the most secure families of its kind on the market today, regardless of how they are used. There may be methods, unknown to Cypress, that can breach the code protection features. Any of these methods, to our knowledge, would be dishonest and possibly illegal. Neither Cypress nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as 'unbreakable.'

Cypress is willing to work with the customer who is concerned about the integrity of their code. Code protection is constantly evolving. We at Cypress are committed to continuously improving the code protection features of our products.



# **Table of Contents**

| 1 Overview                               | 1    |
|--|------|
| 2 Pins                                   | 3    |
| 2.1 Hardware Pins                        |      |
| 2.2 Hardware Ports                       | 6    |
| 2.3 Software Pins.                       | 8    |
| 3 System Settings                        | . 10 |
| 3.1 System Configuration                 |      |
| 3.2 System Debug Settings                | 10   |
| 3.3 System Operating Conditions          |      |
| 4 Clocks.                                |      |
| 4.1 System Clocks                        | . 12 |
| 4.2 Local and Design Wide Clocks         | . 12 |
| 5 Interrupts                             |      |
| 5.1 Interrupts                           | . 14 |
| 6 Flash Memory                           | . 15 |
| 7 Bootloader and Bootloadable            |      |
| 7.1 Bootloadable Application             | 16   |
| 7.2 Bootloader Application               |      |
| 8 Design Contents                        |      |
| 8.1 Schematic Sheet: Page 1              | . 18 |
| 9 Components                             | . 19 |
| 9.1 Component type: Bootloadable [v1.60] | . 19 |
| 9.1.1 Instance Bootloadable_1            | . 19 |
| 9.2 Component type: cydff [v1.30]        |      |
| 9.2.1 Instance cydff_1                   | . 20 |
| 9.2.2 Instance cydff 2                   | . 20 |
| 9.3 Component type: Debouncer [v1.0]     | . 20 |
| 9.3.1 Instance Debouncer_1               | . 20 |
| 9.3.2 Instance Debouncer_2               | . 21 |
| 9.4 Component type: Em_EEPROM [v2.20]    | . 21 |
| 9.4.1 Instance Em_EEPROM                 | . 21 |
| 9.5 Component type: SCB_P4 [v4.0]        | . 22 |
| 9.5.1 Instance SPI                       | . 22 |
| 9.5.2 Instance UART                      | . 36 |
| 9.6 Component type: TCPWM_P4 [v2.10]     | . 51 |
| 9.6.1 Instance Timer_1                   | .51  |
| 9.6.2 Instance Timer_2                   |      |
| 9.7 Component type: Timer [v2.80]        |      |
| 9.7.1 Instance DATALINK_Timer            |      |
| 10 Other Resources.                      | .61  |
|  |      |



#### 1 Overview

The Cypress PSoC 4 is a family of 32-bit devices with the following characteristics:

- Digital system that includes configurable Universal Digital Blocks (UDBs) and specific function peripherals such as PWM, UART, SPI and I2C
- Analog subsystem that includes 12-bit SAR ADC, comparators, op amps, CapSense, LCD drive and more
- Several types of memory elements, including SRAM and flash
- Programming and debug system through Serial Wire Debug (SWD)
- High-performance 32-bit ARM Cortex-M0 core with a nested vectored interrupt controller (NVIC)
- Flexible routing to all pins

Figure 1 shows the major components of a typical <u>PSoC 4200</u> series member PSoC 4 device. For details on all the systems listed above, please refer to the <u>PSoC 4 Technical Reference Manual</u>.

CPU & Memory PSoC 4200 PSoC4 Architecture SWD/TC Cortex **FLASH SRAM** SROM 32-bit M0 4 kB 32 kB 4 kB j [ 48 MHz AHB-Lite FAST MUL NVIC, IRQMX Read Accelerator SRAM Controller ROM Controller System Resources Power
Sleep Control
WIC System Interconnect (Single Layer AHB) POR LVD
REF BOD Peripherals PWRSYS NVLatches Peripheral Interconnect (MMIO) Clock Programmable Programmable 2x SCB-I2C/SPI/UART Clock Control WDT 2x LP Comparator Analog Digital IMO ILO TCPWM Capsense PCLK 0 UDB UDB x1 GPIO Reset Control XRES . . SAR (12-bit) 36x UDB UDB DFT Analog Port Interface & Digital System Interconnect (DSI) Active/Sleep Deep Sleep Hibernate igh Speed I/O Matrix x1 ### 2x OpAmp Programmable I/O

Figure 1. PSoC 4200 Device Series Block Diagram



Table 1 lists the key characteristics of this device.

Table 1. Device Characteristics

| Name                 | Value                      |
|----------------------|----------------------------|
| Part Number          | CY8C4245AXI-483            |
| Package Name         | 44-TQFP                    |
| Family               | PSoC 4                     |
| Series               | PSoC 4200                  |
| Max CPU speed (MHz)  | 48                         |
| Flash size (kB)      | 32                         |
| SRAM size (kB)       | 4                          |
| Vdd range (V)        | 1.71 to 5.5                |
| Automotive qualified | No (Industrial Grade Only) |
| Temp range (Celsius) | -40 to 85                  |

NOTE: The CPU speed noted above is the maximum available speed. The CPU is clocked by HFCLK, listed in the <u>System Clocks</u> section below.

Table 2 lists the device resources that this design uses:

Table 2. Device Resources

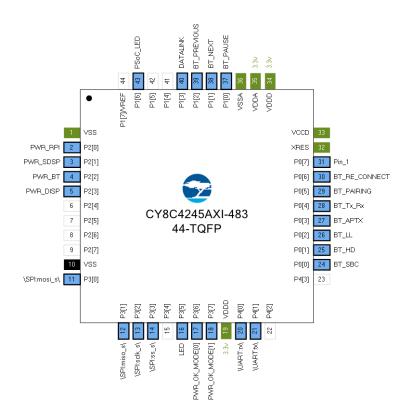
| Resource Type              | Used | Free | Max | % Used   |
|----------------------------|------|------|-----|----------|
| Digital Clocks             | 2    | 2    | 4   | 50.00 %  |
| Interrupts                 | 4    | 28   | 32  | 12.50 %  |
| Ю                          | 26   | 10   | 36  | 72.22 %  |
| Segment LCD                | 0    | 1    | 1   | 0.00 %   |
| CapSense                   | 0    | 1    | 1   | 0.00 %   |
| Die Temp                   | 0    | 1    | 1   | 0.00 %   |
| Serial Communication (SCB) | 2    | 0    | 2   | 100.00 % |
| Timer/Counter/PWM          | 2    | 2    | 4   | 50.00 %  |
| UDB                        |      |      |     |          |
| Macrocells                 | 21   | 11   | 32  | 65.63 %  |
| Unique P-terms             | 23   | 41   | 64  | 35.94 %  |
| Total P-terms              | 25   |      |     |          |
| Datapath Cells             | 1    | 3    | 4   | 25.00 %  |
| Status Cells               | 1    | 3    | 4   | 25.00 %  |
| Statusl Registers          | 1    |      |     |          |
| Control Cells              | 0    | 4    | 4   | 0.00 %   |
| Comparator/Opamp           | 0    | 2    | 2   | 0.00 %   |
| LP Comparator              | 0    | 2    | 2   | 0.00 %   |
| SAR ADC                    | 0    | 1    | 1   | 0.00 %   |
| DAC                        |      |      |     |          |
| 7-bit IDAC                 | 0    | 1    | 1   | 0.00 %   |
| 8-bit IDAC                 | 0    | 1    | 1   | 0.00 %   |



#### 2 Pins

Figure 2 shows the pin layout of this device.

Figure 2. Device Pin Layout





#### 2.1 Hardware Pins

Table 3 contains information about the pins on this device in device pin order. (No connection ["n/c"] pins have been omitted.)

Table 3. Device Pins

| Pin | Port  | Name           | Туре               | <b>Drive Mode</b> |
|-----|-------|----------------|--------------------|-------------------|
| 1   | VSS   | VSS            | Power              |                   |
| 2   | P2[0] | PWR_RPI        | Software<br>In/Out | Strong drive      |
| 3   | P2[1] | PWR_SDSP       | Software In/Out    | Strong drive      |
| 4   | P2[2] | PWR_BT         | Software           | Strong drive      |
| 5   | P2[3] | PWR_DISP       | In/Out<br>Software | Strong drive      |
|     |       |                | In/Out             |                   |
| 6   | P2[4] | GPIO [unused]  |                    |                   |
| 7   | P2[5] | GPIO [unused]  |                    |                   |
| 8   | P2[6] | GPIO [unused]  |                    |                   |
| 9   | P2[7] | GPIO [unused]  |                    |                   |
| 11  | P3[0] | \SPI:mosi_s\   | Dgtl In            | HiZ digital       |
| 12  | P3[1] | \SPI:miso_s\   | Dgtl Out           | Strong drive      |
| 13  | P3[2] | \SPI:sclk_s\   | Dgtl In            | HiZ digital       |
| 14  | P3[3] | \SPI:ss_s\     | Dgtl In            | HiZ digital       |
| 15  | P3[4] | GPIO [unused]  |                    | _                 |
| 16  | P3[5] | LED            | Software<br>In/Out | Strong drive      |
| 17  | P3[6] | PWR_OK_MODE[0] | Dgtl In            | Res pull up       |
| 18  | P3[7] | PWR_OK_MODE[1] | Dgtl In            | Res pull up       |
| 19  | VDDD  | VDDD           | Power              |                   |
| 20  | P4[0] | \UART:rx\      | Dgtl In            | HiZ digital       |
| 21  | P4[1] | \UART:tx\      | Dgtl Out           | Strong drive      |
| 22  | P4[2] | GPIO [unused]  |                    | J                 |
| 23  | P4[3] | GPIO [unused]  |                    |                   |
| 24  | P0[0] | BT_SBC         | Software<br>In/Out | HiZ digital       |
| 25  | P0[1] | BT_HD          | Software<br>In/Out | HiZ digital       |
| 26  | P0[2] | BT_LL          | Software<br>In/Out | HiZ digital       |
| 27  | P0[3] | BT_APTX        | Software<br>In/Out | HiZ digital       |
| 28  | P0[4] | BT_Tx_Rx       | Software<br>In/Out | Strong drive      |
| 29  | P0[5] | BT_PAIRING     | Software<br>In/Out | HiZ digital       |
| 30  | P0[6] | BT_RE_CONNECT  | Software<br>In/Out | Strong drive      |
| 31  | P0[7] | Pin_1          | Software<br>In/Out | Res pull up       |
| 32  | XRES  | XRES           | Dedicated          |                   |
| 33  | VCCD  | VCCD           | Power              |                   |
| 34  | VDDD  | VDDD           | Power              |                   |
| 35  | VDDA  | VDDA           | Power              |                   |



| Pin | Port       | Name          | Туре               | Drive Mode   |
|-----|------------|---------------|--------------------|--------------|
| 36  | VSSA       | VSSA          | Power              |              |
| 37  | P1[0]      | BT_PAUSE      | Software<br>In/Out | Strong drive |
| 38  | P1[1]      | BT_NEXT       | Software<br>In/Out | Strong drive |
| 39  | P1[2]      | BT_PREVIOUS   | Software<br>In/Out | Strong drive |
| 40  | P1[3]      | DATALINK      | Dgtl In            | HiZ digital  |
| 41  | P1[4]      | GPIO [unused] |                    |              |
| 42  | P1[5]      | GPIO [unused] |                    |              |
| 43  | P1[6]      | PSoC_LED      | Software<br>In/Out | Strong drive |
| 44  | P1[7]/VREF | GPIO [unused] |                    |              |

Abbreviations used in Table 3 have the following meanings:

- Dgtl In = Digital Input
- HiZ digital = High impedance digital
- Dgtl Out = Digital Output
- Res pull up = Resistive pull up



#### 2.2 Hardware Ports

Table 4 contains information about the pins on this device in device port order. (No connection ["n/c"], power and dedicated pins have been omitted.)

Table 4. Device Ports

| Port       | Pin | Name          | Туре               | Drive Mode    |
|------------|-----|---------------|--------------------|---------------|
| P0[0]      | 24  | BT SBC        | Software           | HiZ digital   |
| . 0[0]     |     | 51_050        | In/Out             | i ii aigitai  |
| P0[1]      | 25  | BT_HD         | Software           | HiZ digital   |
|            |     | _             | In/Out             |               |
| P0[2]      | 26  | BT_LL         | Software           | HiZ digital   |
|            |     |               | In/Out             |               |
| P0[3]      | 27  | BT_APTX       | Software           | HiZ digital   |
|            |     |               | In/Out             |               |
| P0[4]      | 28  | BT_Tx_Rx      | Software           | Strong drive  |
| Dore:      | 00  | DT DAIDING    | In/Out             | 11.2 1 1      |
| P0[5]      | 29  | BT_PAIRING    | Software           | HiZ digital   |
| Dotei      | 20  | DT DE CONNECT | In/Out<br>Software | Ctuana duiva  |
| P0[6]      | 30  | BT_RE_CONNECT | In/Out             | Strong drive  |
| P0[7]      | 31  | Pin_1         | Software           | Res pull up   |
|            | 31  | FIII_I        | In/Out             | Res puil up   |
| P1[0]      | 37  | BT PAUSE      | Software           | Strong drive  |
| 1 1[0]     | "   | BI_I AGGE     | In/Out             | Ottorig drive |
| P1[1]      | 38  | BT NEXT       | Software           | Strong drive  |
| [.]        |     | D\\           | In/Out             |               |
| P1[2]      | 39  | BT PREVIOUS   | Software           | Strong drive  |
|            |     | _             | In/Out             | 5             |
| P1[3]      | 40  | DATALINK      | Dgtl In            | HiZ digital   |
| P1[4]      | 41  | GPIO [unused] |                    |               |
| P1[5]      | 42  | GPIO [unused] |                    |               |
| P1[6]      | 43  | PSoC_LED      | Software           | Strong drive  |
|            |     |               | In/Out             |               |
| P1[7]/VREF | 44  | GPIO [unused] |                    |               |
| P2[0]      | 2   | PWR_RPI       | Software           | Strong drive  |
|            |     |               | In/Out             |               |
| P2[1]      | 3   | PWR_SDSP      | Software           | Strong drive  |
| Dorot      |     | DWD DT        | In/Out             | 0, 1,         |
| P2[2]      | 4   | PWR_BT        | Software<br>In/Out | Strong drive  |
| DOIOI      | 5   | DWD DISD      | Software           | Ctrong drive  |
| P2[3]      | 5   | PWR_DISP      | In/Out             | Strong drive  |
| P2[4]      | 6   | GPIO [unused] | iii/Out            |               |
| P2[5]      | 7   | GPIO [unused] |                    |               |
| P2[6]      | 8   | GPIO [unused] |                    |               |
| P2[7]      | 9   | GPIO [unused] |                    |               |
| P3[0]      | 11  | \SPI:mosi s\  | Dgtl In            | HiZ digital   |
| P3[1]      | 12  | \SPI:miso s\  | Dgtl Out           | Strong drive  |
| P3[2]      | 13  | \SPI:sclk s\  | Dgti Out           | HiZ digital   |
| P3[3]      | 14  | \SPI:ss s\    | Dgtl In            | HiZ digital   |
| P3[4]      | 15  | GPIO [unused] | Dyn III            | i iiz digitai |
| P3[5]      | 16  | LED           | Software           | Strong drive  |
| ا المام    | '0  | LLD           | In/Out             | Strong unve   |



| Port  | Pin | Name           | Type     | Drive Mode   |
|-------|-----|----------------|----------|--------------|
| P3[6] | 17  | PWR_OK_MODE[0] | Dgtl In  | Res pull up  |
| P3[7] | 18  | PWR_OK_MODE[1] | Dgtl In  | Res pull up  |
| P4[0] | 20  | \UART:rx\      | Dgtl In  | HiZ digital  |
| P4[1] | 21  | \UART:tx\      | Dgtl Out | Strong drive |
| P4[2] | 22  | GPIO [unused]  |          |              |
| P4[3] | 23  | GPIO [unused]  |          |              |

Abbreviations used in Table 4 have the following meanings:

- HiZ digital = High impedance digital
- Res pull up = Resistive pull up
- Dgtl In = Digital Input
- Dgtl Out = Digital Output



#### 2.3 Software Pins

Table 5 contains information about the software pins on this device in alphabetical order. (Only software-accessible pins are shown.)

Table 5. Software Pins

| Name          | Port       | Туре               |
|---------------|------------|--------------------|
| \SPI:miso_s\  | P3[1]      | Dgtl Out           |
| \SPI:mosi_s\  | P3[0]      | Dgtl In            |
| \SPI:sclk_s\  | P3[2]      | Dgtl In            |
| \SPI:ss_s\    | P3[3]      | Dgtl In            |
| \UART:rx\     | P4[0]      | Dgtl In            |
| \UART:tx\     | P4[1]      | Dgtl Out           |
| BT APTX       | P0[3]      | Software           |
| _             |            | In/Out             |
| BT_HD         | P0[1]      | Software           |
|               |            | In/Out             |
| BT_LL         | P0[2]      | Software           |
|               |            | In/Out             |
| BT_NEXT       | P1[1]      | Software           |
|               |            | In/Out             |
| BT_PAIRING    | P0[5]      | Software           |
|               |            | In/Out             |
| BT_PAUSE      | P1[0]      | Software           |
| DT DDEVIOUS   | D4101      | In/Out             |
| BT_PREVIOUS   | P1[2]      | Software<br>In/Out |
| BT RE CONNECT | Dotel      | Software           |
| BI_RE_CONNECT | P0[6]      | In/Out             |
| BT SBC        | P0[0]      | Software           |
| B1_0B0        | 1 0[0]     | In/Out             |
| BT Tx Rx      | P0[4]      | Software           |
| B1_1X_10X     | , 0[1]     | In/Out             |
| DATALINK      | P1[3]      | Dgtl In            |
| GPIO [unused] | P4[3]      |                    |
| GPIO [unused] | P1[4]      |                    |
| GPIO [unused] | P1[5]      |                    |
| GPIO [unused] | P2[7]      |                    |
| GPIO [unused] | P3[4]      |                    |
| GPIO [unused] | P2[5]      |                    |
| GPIO [unused] | P2[6]      |                    |
| GPIO [unused] | P1[7]/VREF |                    |
| GPIO [unused] | P2[4]      |                    |
| GPIO [unused] | P4[2]      |                    |
| LED           | P3[5]      | Software           |
|               | , 0[0]     | In/Out             |
| Pin 1         | P0[7]      | Software           |
| _             |            | In/Out             |
| PSoC_LED      | P1[6]      | Software           |
| _             |            | In/Out             |
| PWR_BT        | P2[2]      | Software           |
|               |            | In/Out             |
| PWR_DISP      | P2[3]      | Software           |
|               |            | In/Out             |



| Name           | Port  | Type               |
|----------------|-------|--------------------|
| PWR_OK_MODE[0] | P3[6] | Dgtl In            |
| PWR_OK_MODE[1] | P3[7] | Dgtl In            |
| PWR_RPI        | P2[0] | Software<br>In/Out |
| PWR_SDSP       | P2[1] | Software<br>In/Out |

Abbreviations used in Table 5 have the following meanings:

- Dgtl Out = Digital Output
- Dgtl In = Digital Input

For more information on reading, writing and configuring pins, please refer to:

- Pins chapter in the <u>System Reference Guide</u>
   CyPins API routines
- Programming Application Interface section in the cy\_pins component datasheet



# **3 System Settings**

# 3.1 System Configuration

Table 6. System Configuration Settings

| Name  | Value          |
|---|----------------|
| Device Configuration Mode                   | Compressed     |
| Unused Bonded IO                            | Allow but warn |
| Heap Size (bytes)                           | 0x0100         |
| Stack Size (bytes)                          | 0x0400         |
| Include CMSIS Core Peripheral Library Files | True           |

# 3.2 System Debug Settings

Table 7. System Debug Settings

| Name            | Value |
|-----------------|-------|
| Debug Select    | GPIO  |
| Chip Protection | Open  |

# 3.3 System Operating Conditions

Table 8. System Operating Conditions

| Name          | Value |
|---------------|-------|
| VDDA (V)      | 3.3   |
| VDDD (V)      | 3.3   |
| Variable VDDA | True  |

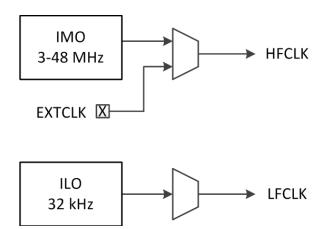


#### 4 Clocks

The clock system includes these clock resources:

- Two internal clock sources:
  - o 3 to 48 MHz Internal Main Oscillator (IMO) ±2% at 3 MHz
  - o 32 kHz Internal Low Speed Oscillator (ILO) output
- HFCLK can be generated using an external signal from EXTCLK pin
- Twelve clock dividers, each with 16-bit divide capability:
  - o Eight can be used for fixed-function blocks
  - o Four can be used for the UDBs

Figure 3. System Clock Configuration





#### 4.1 System Clocks

Table 9 lists the system clocks used in this design.

Table 9. System Clocks

| Name          | Domain | Source     | Desired<br>Freq | Nominal<br>Freq | Accuracy (%) | Start<br>at | Enabled |
|---------------|--------|------------|-----------------|-----------------|--------------|-------------|---------|
|               |        |            | 1104            | 1104            | (70)         | Reset       |         |
| DPLL_Sel      | NONE   | IMO        | 24 MHz          | 24 MHz          | ±2           | True        | True    |
| SYSCLK        | NONE   | HFCLK      | ? MHz           | 24 MHz          | ±2           | True        | True    |
| Direct_Sel    | NONE   | IMO        | 24 MHz          | 24 MHz          | ±2           | True        | True    |
| PLL1_Sel      | NONE   | IMO        | 24 MHz          | 24 MHz          | ±2           | True        | True    |
| PLL0_Sel      | NONE   | IMO        | 24 MHz          | 24 MHz          | ±2           | True        | True    |
| HFCLK         | NONE   | Direct_Sel | 24 MHz          | 24 MHz          | ±2           | True        | True    |
| IMO           | NONE   |            | 24 MHz          | 24 MHz          | ±2           | True        | True    |
| LFCLK         | NONE   | ILO        | ? MHz           | 32 kHz          | ±60          | True        | True    |
| ILO           | NONE   |            | 32 kHz          | 32 kHz          | ±60          | True        | True    |
| Timer2 (WDT2) | NONE   | LFClk      | ? MHz           | ? MHz           | ±0           | False       | False   |
| EXTCLK        | NONE   |            | 24 MHz          | ? MHz           | ±0           | False       | False   |
| DigSig3       | NONE   |            | ? MHz           | ? MHz           | ±0           | False       | False   |
| DigSig2       | NONE   |            | ? MHz           | ? MHz           | ±0           | False       | False   |
| DigSig4       | NONE   |            | ? MHz           | ? MHz           | ±0           | False       | False   |
| DigSig1       | NONE   |            | ? MHz           | ? MHz           | ±0           | False       | False   |
| Timer1 (WDT1) | NONE   | LFClk      | ? MHz           | ? MHz           | ±0           | False       | False   |
| Timer0 (WDT0) | NONE   | LFClk      | ? MHz           | ? MHz           | ±0           | False       | False   |

### 4.2 Local and Design Wide Clocks

Local clocks drive individual analog and digital blocks. Design wide clocks are a user-defined optimization, where two or more analog or digital blocks that share a common clock profile (frequency, etc) can be driven from the same clock divider output source.

Figure 4. Local and Design Wide Clock Configuration

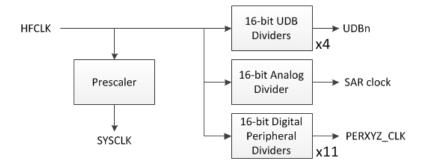


Table 10 lists the local clocks used in this design.

Table 10. Local Clocks

| Name       | Domain                 | Source | Desired<br>Freq | Nominal<br>Freq | Accuracy<br>(%) | Start<br>at<br>Reset | Enabled |
|------------|------------------------|--------|-----------------|-----------------|-----------------|----------------------|---------|
| SPI_SCBCLK | FIXED<br>FUNCT-<br>ION | HFCLK  | 16 MHz          | 24 MHz          | ±2              | True                 | True    |



| Name        | Domain                             | Source | Desired<br>Freq | Nominal<br>Freq | Accuracy<br>(%) | at                   | Enabled |
|-------------|------------------------------------|--------|-----------------|-----------------|-----------------|----------------------|---------|
| UART_SCBCLK | FIXED<br>FUNCT-<br>ION             | HFCLK  | 1.382<br>MHz    | 1.412<br>MHz    | ±2              | <b>Reset</b><br>True | True    |
| SwClock_1   | DIGITAL                            | HFCLK  | 5.12 kHz        | 5.119<br>kHz    | ±2              | True                 | True    |
| SwClock     | DIGITAL,<br>FIXED<br>FUNCT-<br>ION | HFCLK  | 200 Hz          | 200 Hz          | ±2              | True                 | True    |

For more information on clocking resources, please refer to:

- Clocking System chapter in the PSoC 4 Technical Reference Manual
- - o CySysClkWrite API routines



# **5 Interrupts**

### 5.1 Interrupts

This design contains the following interrupt components: (0 is the highest priority)

Table 11. Interrupts

| Name              | Intr<br>Num | Vector | Priority |
|-------------------|-------------|--------|----------|
| DATALINK_INTRRUPT | 0           | 0      | 3        |
| PWR_OK_MODE_SHORT | 2           | 2      | 3        |
| MODE_LONG         | 16          | 16     | 3        |
| PWR_OK_LONG       | 17          | 17     | 3        |

For more information on interrupts, please refer to:

- Interrupt Controller chapter in the PSoC 4 Technical Reference Manual
- Interrupts chapter in the <u>System Reference Guide</u>
   Cylnt API routines and related registers
- Datasheet for cy isr component



# **6 Flash Memory**

PSoC 4 devices offer a host of Flash protection options and device security features that you can leverage to meet the security and protection requirements of an application. These requirements range from protecting configuration settings or Flash data to locking the entire device from external access.

Table 12 lists the Flash protection settings for your design.

Table 12. Flash Protection Settings

| Start<br>Address | End<br>Address | Protection Level    |
|------------------|----------------|---------------------|
| 0x0              | 0xFFF          | W - Full Protection |
| 0x1000           | 0x7FFF         | U - Unprotected     |

Flash memory is organized as rows with each row of flash having 128 bytes. Each flash row can be assigned one of four protection levels:

- U Unprotected
- W Full Protection

For more information on Flash memory and protection, please refer to:

- Flash Protection chapter in the PSoC 4 Technical Reference Manual
- Flash and EEPROM chapter in the System Reference Guide
  - CySysFlash API routines



#### 7 Bootloader and Bootloadable

Figure 5 details the Flash memory map for the bootloader and/or bootloadable application(s) included in this design.

Figure 5. Bootloader Memory Map

Bootloadable application

Start
Address 1

Bootloader application

### 7.1 Bootloadable Application

Table 13. Bootloadable Settings

| Name                               | Value  |
|------------------------------------|--------|
| Application Version                | 0x0000 |
| Application ID                     | 0x0000 |
| Application Custom ID              | 0x0    |
| Application Image 1 Start Address  | 0x1100 |
| Application Image 1 End Address    | 0x7FFF |
| Manual Application Image Placement | False  |

Address 0

### 7.2 Bootloader Application

Table 14. Bootloader Settings

| 5                                    |               |
|--------------------------------------|---------------|
| Name                                 | Value         |
| Checksum Type                        | BasicChecksum |
| Supports Multiple Application Images | False         |
| Application Version                  | 0x0000        |
| Bootloader Start Address             | 0x0           |
| Bootloader End Address               | 0x10A8        |

For more information on the bootloader and startup please refer to:

• Startup and Linking chapter in the <u>System Reference Guide</u>



• Datasheet for Bootloader and Bootloadable component

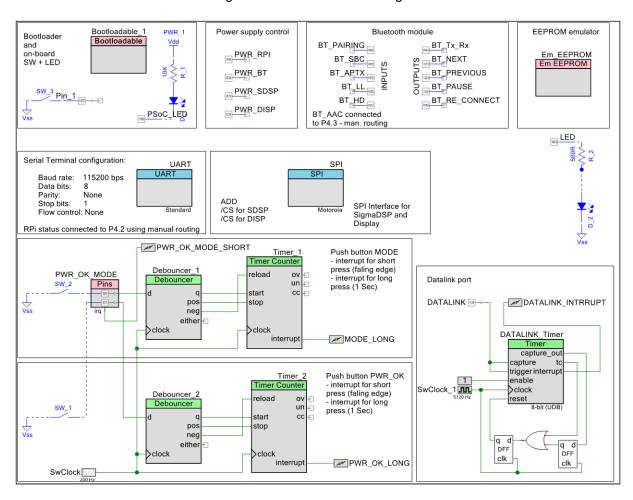


## **8 Design Contents**

This design's schematic content consists of the following schematic sheet:

#### 8.1 Schematic Sheet: Page 1

Figure 6. Schematic Sheet: Page 1



This schematic sheet contains the following component instances:

- Instance <u>Bootloadable\_1</u> (type: Bootloadable\_v1\_60)
- Instance cydff\_1 (type: cydff\_v1\_30)
- Instance cydff\_2 (type: cydff\_v1\_30)
- Instance <u>DATALINK Timer</u> (type: Timer v2 80)
- Instance <u>Debouncer\_1</u>(type: Debouncer\_v1\_0)
- Instance <u>Debouncer\_2</u> (type: Debouncer\_v1\_0)
- Instance <u>Em\_EEPROM</u> (type: Em\_EEPROM\_v2\_20)
- Instance <u>SPI</u>(type: SCB\_P4\_v4\_0)
- Instance <u>Timer\_1</u>(type: TCPWM\_P4\_v2\_10)
- Instance <u>Timer\_2</u> (type: TCPWM\_P4\_v2\_10)
- Instance <u>UART</u>(type: SCB\_P4\_v4\_0)



# 9 Components

9.1 Component type: Bootloadable [v1.60]

9.1.1 Instance Bootloadable\_1

Description: Provides bootloadable application functionality.

Instance type: Bootloadable [v1.60]

Datasheet: online component datasheet for Bootloadable

Table 15. Component Parameters for Bootloadable\_1

| Parameter Name      | -<br>Value  | Description  |
|---------------------|---|--|
| appCustomID         | 0   | Provides a 4 byte custom ID number to represent anything in the Bootloadable application.  |
| appID               | 0   | Provides a 2 byte number to represent the ID of the bootloadable application.  |
| appVersion          | 0   | Provides a 2 byte number to represent the version of the bootloadable application.   |
| autoPlacement       | true  | Provides a method for PSoC Creator to place a Bootloadable application image at a specified location. If true, the image will be placed automatically. If false, the image will be placed at an address specified by the Placement Address option. |
| checksumExcludeSize | 128   | Provides a size in bytes of checksum exclude section   |
| elfFilePath         | C:\Program Files (x86)\Cypress\CY8CKIT-049- 42xx\1.0\Firmware\SCB Bootloader\UART Bootloader.cydsn\Co- rtexM0\ARM_GCC 484\Debug\UART Bootloader.elf | Provides a reference to the Bootloader application (.elf) that is associated with this Bootloadable application.   |
| hexFilePath         | C:\Program Files (x86)\Cypress\CY8CKIT-049- 42xx\1.0\Firmware\SCB Bootloader\UART Bootloader.cydsn\Co- rtexM0\ARM_GCC 484\Debug\UART Bootloader.hex | Provides a reference to the Bootloader application (.hex) that is associated with this Bootloadable application.   |
| placementAddress    | 0   | Allows specifying an address where the bootloadable application will be placed in the memory. Available only if the Automatic Application Image Placement option is true.  |
| User Comments       |   | Instance-specific comments.  |

# 9.2 Component type: cydff [v1.30]



#### 9.2.1 Instance cydff\_1

Description: D Flip Flop with configurable reset and preset

Instance type: cydff [v1.30]

Datasheet: online component datasheet for cydff

Table 16. Component Parameters for cydff\_1

| Parameter Name   | Value | Description   |
|------------------|-------|---|
| ArrayWidth       | 1     | Width of d and q terminals. Must be between 1 and 32.                           |
| MultiPresetReset | true  | Defines options to set the preset and reset to be either a bus or a single bit. |
| PresetOrReset    | None  | Preset/ Reset parameter   |
| SmallMode        | true  |   |
| User Comments    |       | Instance-specific comments.   |

#### 9.2.2 Instance cydff\_2

Description: D Flip Flop with configurable reset and preset

Instance type: cydff [v1.30]

Datasheet: online component datasheet for cydff

Table 17. Component Parameters for cydff\_2

| Parameter Name   | Value | Description   |
|------------------|-------|---|
| ArrayWidth       | 1     | Width of d and q terminals. Must  |
|                  |       | be between 1 and 32.  |
| MultiPresetReset | true  | Defines options to set the preset and reset to be either a bus or a single bit. |
| PresetOrReset    | None  | Preset/ Reset parameter   |
| SmallMode        | true  |   |
| User Comments    |       | Instance-specific comments.   |

#### 9.3 Component type: Debouncer [v1.0]

#### 9.3.1 Instance Debouncer\_1

Description: Debounces the input digital signal from most types of switches

Instance type: Debouncer [v1.0]

Datasheet: online component datasheet for Debouncer

Table 18. Component Parameters for Debouncer\_1

| Parameter Name   | Value | Description                    |
|------------------|-------|--------------------------------|
| EitherEdgeDetect | true  | Specifies whether the positive |
|                  |       | or negative edge detection is  |
|                  |       | enabled for the component.     |
| NegEdgeDetect    | true  | Specifies whether the negative |
|                  |       | edge detection is enabled for  |
|                  |       | the component.                 |
| PosEdgeDetect    | true  | Specifies whether the positive |
|                  |       | edge detection is enabled for  |
|                  |       | the component.                 |



| Parameter Name | Value | Description                 |
|----------------|-------|-----------------------------|
| SignalWidth    | 1     | Determines the bus width of |
|                |       | input and output terminals. |
| User Comments  |       | Instance-specific comments. |

### 9.3.2 Instance Debouncer\_2

Description: Debounces the input digital signal from most types of switches

Instance type: Debouncer [v1.0]

Datasheet: online component datasheet for Debouncer

Table 19. Component Parameters for Debouncer\_2

| Parameter Name   | Value | Description   |
|------------------|-------|---|
| EitherEdgeDetect | true  | Specifies whether the positive or negative edge detection is enabled for the component. |
| NegEdgeDetect    | true  | Specifies whether the negative edge detection is enabled for the component.             |
| PosEdgeDetect    | true  | Specifies whether the positive edge detection is enabled for the component.             |
| SignalWidth      | 1     | Determines the bus width of input and output terminals.                                 |
| User Comments    |       | Instance-specific comments.   |

# 9.4 Component type: Em\_EEPROM [v2.20]

#### 9.4.1 Instance Em\_EEPROM

Description: Emulates an EEPROM device in flash memory.

Instance type: Em\_EEPROM [v2.20]

Datasheet: online component datasheet for Em\_EEPROM

Table 20. Component Parameters for Em\_EEPROM

| Parameter Name | Value | Description   |
|----------------|-------|---|
| EEPROM Size    | 64    | Sets size of EEPROM. The size is rounded up to a full EEPROM page size.   |
| Redundant Copy | No    | If selected, then an 8-bit checksum is calculated on each row of data (that checksum is stored in the row), and a redundant copy of the row is stored in another location.  When data is read the checksum is checked first. If the checksum is bad the redundant copy is restored. |
| User Comments  |       | Instance-specific comments.   |



| Parameter Name    | Value | Description                    |
|-------------------|-------|--------------------------------|
| Wear Level Factor | None  | Selects how much wear leveling |
|                   |       | is required. The higher the    |
|                   |       | factor the more flash is used, |
|                   |       | but the higher number of       |
|                   |       | erase/write cycles can be done |
|                   |       | on the EEPROM. Multiply this   |
|                   |       | number by the datasheet write  |
|                   |       | endurance spec to determine    |
|                   |       | max write cycles.              |

9.5 Component type: SCB\_P4 [v4.0]

#### 9.5.1 Instance SPI

**Description: Serial Communication Block (SCB)** 

Instance type: SCB\_P4 [v4.0]
Datasheet: online component datasheet for SCB\_P4

Table 21. Component Parameters for SPI

| Parameter Name           | Value | Description   |
|--------------------------|-------|---|
| Ezl2cByteModeEnable      | false | When the SCB mode is EZI2C, this parameter specifies the number of bits per FIFO data element.  The byte mode – false: a 16-bit FIFO data element. The FIFO depth is 8 entries.  The byte mode – true: an 8-bit FIFO data element. The FIFO depth is 16 entries.  Applicable only for devices other than PSoC 4000/PSoC |
|                          |       | 4100/PSoC 4200.   |
| Ezl2cClockFromTerm       | false | When the SCB mode is EZI2C,<br>this parameter provides a clock<br>terminal to connect a clock<br>outside the component.   |
| Ezl2cClockStretching     | true  | When the SCB mode is EZI2C,<br>this parameter specifies<br>whether the SCL is stretched<br>while in EZI2C operation.  |
| Ezl2cDataRate            | 100   | When the SCB mode is EZI2C, this parameter defines EZI2C Data rate in kbps. The standard data rates are: 100, 400 and 1000 kbps.  |
| Ezl2cNumberOfAddresses   | 1     | When the SCB mode is EZI2C,<br>this parameter defines the<br>number of I2C slave addresses<br>that device respond to.   |
| Ezl2cPrimarySlaveAddress | 8     | When the SCB mode is EZI2C,<br>this parameter specifies EZI2C<br>primary 7-bits slave address<br>(MSB ignored).   |



| Parameter Name                                   | Value | Description   |
|--|-------|---|
| Ezl2cSecondarySlaveAddress                       | 9     | When the SCB mode is EZI2C,                               |
| -  |       | this parameter specifies EZI2C                            |
|  |       | secondary 7-bits slave address                            |
|  |       | (MSB ignored).  |
|  |       | Only applicable when EZI2C                                |
|  |       | clock stretching option is set.                           |
| Ezl2cSubAddressSize                              | 8     | When the SCB mode is EZI2C,                               |
|  |       | this parameter specifies the                              |
|  |       | maximum size of the slave                                 |
|  |       | buffer that is exposed to the                             |
|  |       | master: 8bits – maximum buffer                            |
|  |       | size is 256 bytes, 16 bits –                              |
|  |       | maximum buffer size is 65535                              |
| 5 10 W 1 5 11                                    | 6.1   | bytes.  |
| Ezl2cWakeEnable                                  | false | When the SCB mode is EZI2C,                               |
|  |       | this parameter enables wakeup                             |
|  |       | from Deep Sleep on I2C                                    |
| 100 D 1/ //                                      | -     | address match event.                                      |
| I2C Bus Voltage                                  | 3.3   | When the SCB mode is I2C, this                            |
|  |       | parameter specifies the voltage                           |
|  |       | applied to the pull-up resistors                          |
|  |       | on the I2C bus.   |
|  |       | Only applicable for devices                               |
|  |       | other than PSoC 4000/PSoC                                 |
|  |       | 4100/PSoC 4200.   |
| I2C Bus Voltage                                  | 3.3   | When the SCB mode is EZI2C,                               |
| 1 - 5 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 -          |       | this parameter specifies the                              |
|  |       | voltage applied to the pull-up                            |
|  |       | resistors on the I2C bus.                                 |
|  |       |   |
|  |       | Only applicable for devices                               |
|  |       | other than PSoC 4000/PSoC                                 |
|  |       | 4100/PSoC 4200.   |
| I2cAcceptAddress                                 | false | When the SCB mode is I2C, this                            |
|  |       | parameter specifies whether to                            |
|  |       | accept the match slave address                            |
|  |       | in RX FIFO or not. All slave                              |
|  |       | matched addresses are ACKed.                              |
|  |       | The user has to register the                              |
|  |       | callback function to handle                               |
|  |       | accepted addresses. This                                  |
|  |       | feature has to be used when more than one address support |
|  |       | is required.  |
| I2cAcceptGeneralCall                             | false | When the SCB mode is I2C, this                            |
| IZUAUUEPIUEI III III III III III III III III III | laise | parameter specifies whether to                            |
|  |       | accept the general call address.                          |
|  |       | The general call address is                               |
|  |       | ACKed when accepted and                                   |
|  |       | NAKed otherwise. The user has                             |
|  |       | to register the callback function                         |
|  |       | to handle the general call                                |
|  |       | address.  |



| Devementes Neme            | Value | EMBEDDED IN TO  |
|----------------------------|-------|---|
| Parameter Name             | Value | Description   |
| I2cByteModeEnable          | false | When the SCB mode is I2C, this parameter specifies the number of bits per FIFO data element. The byte mode – false: a 16-bit FIFO data element. The FIFO depth is 8 entries.  |
|                            |       | The byte mode – true: an 8-bit FIFO data element. The FIFO depth is 16 entries.  Applicable only for devices  |
|                            |       | other than PSoC 4000/PSoC 4100/PSoC 4200.   |
| I2cClockFromTerm           | false | When the SCB mode is I2C, this parameter provides a clock terminal to connect a clock outside the component.  |
| I2cDataRate                | 100   | When the SCB mode is I2C, this parameter specifies the data rate in kbps. The standard data rates are: 100, 400 and 1000 kbps.  |
| I2cExternIntrHandler       | false | When the SCB mode is I2C, this parameter specifies whether the I2C interrupt handler is configured in SCB_I2CInit(). This parameter is intended to be used by the PM/SM bus component. The modification parameter default value causes I2C mode failures. |
| I2cManualOversampleControl | true  | When the SCB mode is I2C, this parameter specifies the method of calculating the oversampling as manual or automatic.   |
| I2cMode                    | Slave | When the SCB mode is I2C, this parameter defines the I2C operation mode as: Slave, Master, Multi-Master or Multi-Master-Slave.  |
| I2cOvsFactor               | 16    | When the SCB mode is I2C, this parameter defines the oversampling factor of SCBCLK.   |
| I2cOvsFactorHigh           | 8     | When the SCB mode is I2C, this parameter defines the high oversampling factor of SCBCLK. Only applicable for I2C Master modes.  |
| I2cOvsFactorLow            | 8     | When the SCB mode is I2C, this parameter defines the low oversampling factor of SCBCLK. Only applicable for I2C Master modes.   |



| Parameter Name       | Value | Description  |
|----------------------|-------|--|
| I2cSlaveAddress      | 8     | When the SCB mode is I2C, this parameter specifies the I2C 7-bits slave address (MSB ignored).   |
| I2cSlaveAddressMask  | 254   | When the SCB mode is I2C, this parameter specifies the I2C Slave address mask.  Bit value 0 – excludes bit from address comparison.  Bit value 1 – the bit needs to match with the corresponding bit of the I2C slave address. |
| I2cWakeEnable        | false | When the SCB mode is I2C, this parameter enables wakeup from Deep Sleep on an I2C address match event.   |
| ScbMisoSdaTxEnable   | true  | This parameter defines the availability of the spi_miso_i2csda_uart_tx pin.  |
| ScbMode              | SPI   | This parameter defines the mode of operation for the SCB component.  |
| ScbMosiSclRxEnable   | true  | This parameter defines the availability of the spi_mosi_i2cscl_uart_rx pin.  |
| ScbRxWakeIrqEnable   | false | This parameter defines the availability of the spi_mosi_i2cscl_uart_rx_wake pin.   |
| ScbSclkEnable        | false | This parameter defines the availability of the sclk pin.   |
| ScbSs0Enable         | false | This parameter defines the availability of the ss0 pin.  |
| ScbSs1Enable         | false | This parameter defines the availability of the ss1 pin.  |
| ScbSs2Enable         | false | This parameter defines the availability of the ss2 pin.  |
| ScbSs3Enable         | false | This parameter defines the availability of the ss3 pin.  |
| Show EZI2C Terminals | false | When the SCB mode is EZI2C, this parameter removes internal pins and expose signals to terminals. The exposed terminals must be connected to the pins.   |
| Show I2C Terminals   | false | When the SCB mode is I2C, this parameter removes internal pins and expose signals to terminals. The exposed terminals must be connected to the pins.   |
| Show SPI Terminals   | false | When the SCB mode is SPI, this parameter removes internal pins and expose signals to terminals. The exposed terminals must be connected to the pins or SmartIO component.  |



| Damana - t N        | \/-I      | EMBEDDED IN TO                    |
|---------------------|-----------|-----------------------------------|
| Parameter Name      | Value     | Description                       |
| Show UART Terminals | false     | When the SCB mode is UART,        |
|                     |           | this parameter removes internal   |
|                     |           | pins and expose signals to        |
|                     |           | terminals. The exposed            |
|                     |           | terminals must be connected to    |
|                     |           | the pins or SmartIO component.    |
| Slew Rate           | Fast      | When the SCB mode is EZI2C,       |
|                     |           | this parameter specifies the      |
|                     |           | slew rate settings of the I2C     |
|                     |           | pins.                             |
|                     |           | For devices supporting GPIO       |
|                     |           | Over-Voltage Tolerance            |
|                     |           | (GPIO_OVT) pins, I2C FM+          |
|                     |           | options should be used when       |
|                     |           | I2C data rate is greater than     |
|                     |           | 400 kbps. This option also        |
|                     |           | requires the I2C bus voltage to   |
|                     |           | be defined.Refer to the Device    |
|                     |           | Datasheet to determine which      |
|                     |           | pins are GPIO_OVT capable.        |
| Slew Rate           | Fast      | When the SCB mode is I2C, this    |
|                     |           | parameter specifies the slew      |
|                     |           | rate settings of the I2C pins.    |
|                     |           | For devices supporting GPIO       |
|                     |           | Over-Voltage Tolerance            |
|                     |           | (GPIO_OVT) pins, I2C FM+          |
|                     |           | options should be used when       |
|                     |           | I2C data rate is greater than     |
|                     |           | 400 kbps. This option also        |
|                     |           | requires the I2C bus voltage to   |
|                     |           | be defined. Refer to the Device   |
|                     |           | Datasheet to determine which      |
|                     |           | pins are GPIO_OVT capable.        |
| SpiBitRate          | 1000      | When the SCB mode is SPI,         |
| ·                   |           | this parameter specifies the Bit  |
|                     |           | rate in kbps (up to 8000 kbps);   |
|                     |           | the actual rate may differ based  |
|                     |           | on available clock frequency      |
|                     |           | and component settings. This      |
|                     |           | parameter has no effect if the    |
|                     |           | Clock from terminal parameter     |
|                     |           | is enabled.                       |
| SpiBitsOrder        | MSB First | When the SCB mode is SPI,         |
| '                   |           | this parameter defines the bit    |
|                     |           | order as: MSB first or LSB first. |
| SpiByteModeEnable   | false     | When the SCB mode is SPI,         |
| opisytemodeznasie   | laiss     | this parameter specifies the      |
|                     |           | number of bits per FIFO data      |
|                     |           | element.                          |
|                     |           | The byte mode – false: a 16-bit   |
|                     |           | FIFO data element. The FIFO       |
|                     |           | depth is 8 entries.               |
|                     |           | The byte mode – true: an 8-bit    |
|                     |           | FIFO data element. The FIFO       |
|                     |           | depth is 16 entries.              |
|                     |           | depui is to enuies.               |
|                     |           | Applicable only for devices       |
|                     |           | other than PSoC 4000/PSoC         |
|                     |           | 4100/PSoC 4200.                   |
|                     |           | 4100/F30C 4200.                   |



| Parameter Name       | Value | Description                                 |
|----------------------|-------|---|
| SpiClockFromTerm     | false | When the SCB mode is SPI,                   |
|                      |       | this parameter provides a clock             |
|                      |       | terminal to connect a clock                 |
|                      |       | outside the component.                      |
| SpiFreeRunningSclk   | false | When the SCB mode is SPI,                   |
|                      |       | this parameter specifies the                |
|                      |       | SCLK generation by the master               |
|                      |       | as: gated or free running                   |
|                      |       | (continuous).                               |
|                      |       | Applicable only for devices                 |
|                      |       | other than PSoC 4000/PSoC                   |
|                      |       | 4100/PSoC 4200.                             |
| SpiInterruptMode     | None  | When the SCB mode is SPI,                   |
|                      |       | this parameter specifies the                |
|                      |       | interrupt mode. None: Removes               |
|                      |       | all interrupt support. Internal:            |
|                      |       | Leaves the interrupt SCBIRQ                 |
|                      |       | inside the component - the                  |
|                      |       | interrupt terminal becomes                  |
|                      |       | invisible. External: Provides an            |
|                      |       | interrupt terminal to connect an            |
|                      |       | interrupt outside the component.            |
| SpiIntrMasterSpiDone | false | When the SCB mode is SPI,                   |
|                      |       | this parameter enables the                  |
|                      |       | SCB.INTR_M. SPI_DONE                        |
|                      |       | interrupt source. SCB.INTR M. SPI DONE: all |
|                      |       | data are sent into TX FIFO and              |
|                      |       | the TX FIFO and the shifter                 |
|                      |       | register are emptied.                       |
|                      |       | Only applicable for SPI Master              |
|                      |       | mode.                                       |
| SpiIntrRxFull        | false | When the SCB mode is SPI,                   |
|                      |       | this parameter enables the                  |
|                      |       | SCB.INTR_RX.FULL interrupt                  |
|                      |       | source.                                     |
|                      |       | SCB.INTR_RX.FULL trigger                    |
| 0.11.12.11.15        |       | condition: RX FIFO is full.                 |
| SpiIntrRxNotEmpty    | false | When the SCB mode is SPI,                   |
|                      |       | this parameter enables the                  |
|                      |       | SCB.INTR_RX.NOT_EMPTY interrupt source.     |
|                      |       | SCB.INTR RX.NOT EMPTY                       |
|                      |       | trigger condition: RX FIFO is not           |
|                      |       | empty. There is at least one                |
|                      |       | entry to get data from.                     |
| SpiIntrRxOverflow    | false | When the SCB mode is SPI,                   |
|                      |       | this parameter enables the                  |
|                      |       | SCB.INTR_RX.OVERFLOW                        |
|                      |       | interrupt source.                           |
|                      |       | SCB.INTR_RX.OVERFLOW                        |
|                      |       | trigger condition: attempt to               |
|                      |       | write to a full RX FIFO.                    |



| Parameter Name       | Value | Description Description           |
|----------------------|-------|-----------------------------------|
| SpilntrRxTrigger     | false | When the SCB mode is SPI,         |
| Spiritifix riiggei   | laise | this parameter enables the        |
|                      |       | SCB.INTR_RX.TRIGGER               |
|                      |       | interrupt source.                 |
|                      |       | SCB.INTR_RX.TRIGGER               |
|                      |       | trigger condition: remains active |
|                      |       | until RX FIFO has more entries    |
|                      |       | than the value specified by       |
|                      |       | SpiRxTriggerLevel.                |
| SpiIntrRxUnderflow   | false | When the SCB mode is SPI,         |
| Opiniti Condemow     | laise | this parameter enables the        |
|                      |       | SCB.INTR RX.UNDERFLOW             |
|                      |       | interrupt source.                 |
|                      |       | SCB.INTR_RX.UNDERFLOW             |
|                      |       | trigger condition: attempt to     |
|                      |       | read from an empty RX FIFO.       |
| SpiIntrSlaveBusError | false | When the SCB mode is SPI,         |
| OpinitiolaveBusEnoi  | laise | this parameter enables the        |
|                      |       | SCB.INTR SLAVE.BUS -              |
|                      |       | ERROR interrupt source.           |
|                      |       | SCB.INTR SLAVE.BUS -              |
|                      |       | ERROR trigger condition: slave    |
|                      |       | select line is deselected at an   |
|                      |       | unexpected time in the SPI        |
|                      |       | transfer.                         |
|                      |       | Only applicable for SPI Slave     |
|                      |       | mode.                             |
| SpiIntrTxEmpty       | false | When the SCB mode is SPI,         |
|                      | laise | this parameter enables the        |
|                      |       | SCB.INTR_TX.EMPTY interrupt       |
|                      |       | source.                           |
|                      |       | SCB.INTR_TX.EMPTY trigger         |
|                      |       | condition: TX FIFO is empty.      |
| SpiIntrTxNotFull     | false | When the SCB mode is SPI,         |
|                      |       | this parameter enables the        |
|                      |       | SCB.INTR TX.NOT FULL              |
|                      |       | interrupt source.                 |
|                      |       | SCB.INTR_TX.NOT_FULL              |
|                      |       | trigger condition: TX FIFO is not |
|                      |       | full. There is at least one entry |
|                      |       | to put data.                      |
| SpiIntrTxOverflow    | false | When the SCB mode is SPI,         |
|                      |       | this parameter enables the        |
|                      |       | SCB.INTR_TX.OVERFLOW              |
|                      |       | interrupt source.                 |
|                      |       | SCB.INTR_TX.OVERFLOW              |
|                      |       | trigger condition: attempt to     |
|                      |       | write to a full TX FIFO.          |
| SpiIntrTxTrigger     | false | When the SCB mode is SPI,         |
|                      |       | this parameter enables the        |
|                      |       | SCB.INTR_TX.TRIGGER               |
|                      |       | interrupt source.                 |
|                      |       | SCB.INTR_TX.TRIGGER               |
|                      |       | trigger condition: remains active |
|                      |       | until TX FIFO has fewer entries   |
|                      |       | than the value specified by       |
|                      |       | SpiTxTriggerLevel.                |



| Parameter Name          | Value | Description  |
|-------------------------|-------|--|
| SpilntrTxUnderflow      | false | When the SCB mode is SPI, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.   |
| SpiLateMisoSampleEnable | false | When the SCB mode is SPI,<br>this parameter enables late<br>sampling of the MISO line by<br>the master.  |
| SpiMedianFilterEnable   | false | When the SCB mode is SPI,<br>this parameter applies a digital<br>3 tap median filter to the SPI<br>input line.   |
| SpiMode                 | Slave | When the SCB mode is SPI,<br>this parameter selects SPI<br>mode of operation as: Slave or<br>Master.   |
| SpiNumberOfRxDataBits   | 8     | When the SCB mode is SPI, this parameter specifies the number of data bits inside the SPI byte/word for RX direction.  |
| SpiNumberOfSelectLines  | 1     | When the SCB mode is SPI,<br>this parameter defines the<br>number of slave select lines.<br>The SPI Slave has only one<br>slave select line. The SPI<br>Master has up to 4 lines.  |
| SpiNumberOfTxDataBits   | 8     | When the SCB mode is SPI,<br>this parameter define the<br>number of data bits inside the<br>SPI byte/word for TX direction.  |
| SpiOvsFactor            | 16    | When the SCB mode is SPI,<br>this parameter defines the<br>oversampling factor of<br>SCBCLK.   |
| SpiRemoveMiso           | false | When the SCB mode is SPI,<br>this parameter removes the<br>MISO pin.   |
| SpiRemoveMosi           | false | When the SCB mode is SPI,<br>this parameter removes the<br>MOSI pin.   |
| SpiRemoveSclk           | false | When the SCB mode is SPI,<br>this parameter removes the<br>SCLK pin.   |
| SpiRxBufferSize         | 8     | When the SCB mode is SPI,<br>this parameter defines the size<br>of the RX buffer.  |
| SpiRxOutputEnable       | false | When the SCB mode is SPI, this parameter enables the RX trigger output terminal of the component. This terminal must be connected to the DMA input trigger or left unconnected. Only applicable for devices which have a DMA controller. |



|                         |                | C I PRI   |
|-------------------------|----------------|---|
| Parameter Name          | Value          | Description   |
| SpiRxTriggerLevel       | 7              | When the SCB mode is SPI,                             |
|                         |                | this parameter defines the                            |
|                         |                | number of entries in the RX                           |
|                         |                | FIFO to control the SCB.INTR                          |
|                         |                | RX.TRIGGER interrupt event or                         |
|                         |                | RX DMA trigger output.                                |
| SpiSclkMode             | CPHA = 0, CPOL | When the SCB mode is SPI,                             |
|                         | = 0            | this parameter defines the serial                     |
|                         |                | clock phase (CPHA) and                                |
|                         |                | polarity (CPOL).                                      |
| SpiSs0Polarity          | Active Low     | When the SCB mode is SPI,                             |
|                         |                | this parameter specifies active                       |
|                         |                | polarity of slave select 0.                           |
|                         |                |   |
|                         |                | Applicable only for devices                           |
|                         |                | other than PSoC 4000/PSoC                             |
|                         |                | 4100/PSoC 4200.                                       |
| SpiSs1Polarity          | Active Low     | When the SCB mode is SPI,                             |
|                         |                | this parameter specifies active                       |
|                         |                | polarity of slave select 1.                           |
|                         |                |   |
|                         |                | Applicable only for devices                           |
|                         |                | other than PSoC 4000/PSoC                             |
|                         |                | 4100/PSoC 4200.                                       |
| SpiSs2Polarity          | Active Low     | When the SCB mode is SPI,                             |
|                         |                | this parameter specifies active                       |
|                         |                | polarity of slave select 2.                           |
|                         |                | Analisada andreferada de de                           |
|                         |                | Applicable only for devices                           |
|                         |                | other than PSoC 4000/PSoC                             |
| 0.10.00.1.11            |                | 4100/PSoC 4200.                                       |
| SpiSs3Polarity          | Active Low     | When the SCB mode is SPI,                             |
|                         |                | this parameter specifies active                       |
|                         |                | polarity of slave select 3.                           |
|                         |                | Applicable only for devices                           |
|                         |                | Applicable only for devices other than PSoC 4000/PSoC |
|                         |                | 4100/PSoC 4200.                                       |
| SpiSubModo              | Motorola       | When the SCB mode is SPI,                             |
| SpiSubMode              | IVIOLOTOIA     | this parameter defines the sub                        |
|                         |                | mode of the SPI as: Motorola,                         |
|                         |                | TI(Start Coincides), TI(Start                         |
|                         |                | Precedes), or National                                |
|                         |                | Semiconductor.  |
| SpiTransferSeparation   | Continuous     | When the SCB mode is SPI,                             |
| Opi i ansierooparation  | Continuous     | this parameter defines the type                       |
|                         |                | of SPI transfers separation as:                       |
|                         |                | continuous or separated.                              |
| SpiTxBufferSize         | 8              | When the SCB mode is SPI,                             |
| OpiTABalloloizo         |                | this parameter defines the size                       |
|                         |                | of the TX buffer.                                     |
| SpiTxOutputEnable       | false          | When the SCB mode is SPI,                             |
| Op. 1 A Output Liliabio | idise          | this parameter enables the TX                         |
|                         |                | trigger output terminal of the                        |
|                         |                | component. This terminal must                         |
|                         |                | be connected to the DMA input                         |
|                         |                | trigger or left unconnected. Only                     |
|                         |                | applicable for devices which                          |
|                         |                | have a DMA controller.                                |
|                         |                |   |



| Parameter Name     | Value        | Description   |
|--------------------|--------------|---|
| SpiTxTriggerLevel  | 0            | When the SCB mode is SPI,                             |
| , 55               |              | this parameter defines the                            |
|                    |              | number of entries in the TX                           |
|                    |              | FIFO to control the SCB.INTR                          |
|                    |              | TX.TRIGGER interrupt event or                         |
|                    |              | TX DMA trigger output.                                |
| SpiWakeEnable      | false        | When the SCB mode is SPI,                             |
|                    |              | this parameter enables wakeup                         |
|                    |              | from Deep Sleep on slave                              |
|                    |              | select event.   |
| UartByteModeEnable | false        | When the SCB mode is UART,                            |
|                    |              | this parameter specifies the                          |
|                    |              | number of bits per FIFO data                          |
|                    |              | element.  |
|                    |              | The byte mode – false: a 16-bit                       |
|                    |              | FIFO data element. The FIFO                           |
|                    |              | depth is 8 entries.                                   |
|                    |              | The byte mode – true: an 8-bit                        |
|                    |              | FIFO data element. The FIFO                           |
|                    |              | depth is 16 entries.                                  |
|                    |              | Applicable only for devices                           |
|                    |              | Applicable only for devices other than PSoC 4000/PSoC |
|                    |              | 4100/PSoC 4200.                                       |
| UartClockFromTerm  | false        | When the SCB mode is UART,                            |
| CartClockFromTerm  | laise        | this parameter provides a clock                       |
|                    |              | terminal to connect a clock                           |
|                    |              | outside the component.                                |
| UartCtsEnable      | false        | When the SCB mode is UART,                            |
| OartoisEriable     | laise        | this parameter enables the cts                        |
|                    |              | input.  |
|                    |              | input.  |
|                    |              | Only applicable for devices                           |
|                    |              | other than PSoC 4000/PSoC                             |
|                    |              | 4100/PSoC 4200.                                       |
| UartCtsPolarity    | Active Low   | When the SCB mode is UART,                            |
| Cartotol clarity   | 7101170 2011 | this parameter specifies active                       |
|                    |              | polarity of an input cts signal.                      |
|                    |              | poranity or an impart ore orginal.                    |
|                    |              | Only applicable for devices                           |
|                    |              | other than PSoC 4000/PSoC                             |
|                    |              | 4100/PSoC 4200.                                       |
| UartDataRate       | 115200       | When the SCB mode is UART,                            |
|                    |              | this parameter specifies the                          |
|                    |              | Baud rate in bps (up to 1000                          |
|                    |              | kbps); the actual rate may differ                     |
|                    |              | based on available clock                              |
|                    |              | frequency and component                               |
|                    |              | settings. This parameter has no                       |
|                    |              | effect if the Clock from terminal                     |
|                    |              | parameter is enabled.                                 |
| UartDirection      | TX + RX      | When the SCB mode is UART,                            |
|                    |              | this parameter enables RX or                          |
|                    |              | TX direction or both                                  |
|                    |              | simultaneously.                                       |
| UartDropOnFrameErr | false        | When the SCB mode is UART,                            |
|                    |              | this parameter defines whether                        |
|                    |              | the data is dropped from RX                           |
|                    |              | FIFO on a frame error event.                          |



| Parameter Name          | Value | Description EMBEDDED IN TO             |
|-------------------------|-------|--|
| UartDropOnParityErr     | false | When the SCB mode is UART,             |
|                         |       | this parameter determines              |
|                         |       | whether the data is dropped            |
|                         |       | from RX FIFO on a parity error         |
|                         |       | event.                                 |
| UartInterruptMode       | None  | When the SCB mode is UART,             |
| ·                       |       | this parameter specifies the           |
|                         |       | interrupt mode. None: Removes          |
|                         |       | all interrupt support. Internal:       |
|                         |       | Leaves the interrupt SCBIRQ            |
|                         |       | inside the component - the             |
|                         |       | interrupt terminal becomes             |
|                         |       | invisible. External: Provides an       |
|                         |       | interrupt terminal to connect an       |
|                         |       | interrupt outside component.           |
| UartIntrRxBreakDetected | false | This parameter enables the RX          |
|                         |       | break detection interrupt source       |
|                         |       | to trigger the interrupt output.       |
| UartIntrRxFrameErr      | false | When the SCB mode is UART,             |
|                         |       | this parameter enables the             |
|                         |       | SCB.INTR_RX.FRAME                      |
|                         |       | ERROR interrupt source.                |
|                         |       | SCB.INTR_RX.FRAME                      |
|                         |       | ERROR trigger condition: frame         |
|                         |       | error in received data frame.          |
| UartIntrRxFull          | false | When the SCB mode is UART,             |
|                         |       | this parameter enables the             |
|                         |       | SCB.INTR_RX.FULL interrupt             |
|                         |       | source.                                |
|                         |       | SCB.INTR_RX.FULL trigger               |
|                         |       | condition: RX FIFO is full.            |
| UartIntrRxNotEmpty      | false | When the SCB mode is UART,             |
|                         |       | this parameter enables the             |
|                         |       | SCB.INTR_RX.NOT_EMPTY                  |
|                         |       | interrupt source.                      |
|                         |       | SCB.INTR_RX.NOT_EMPTY                  |
|                         |       | trigger condition: RX FIFO is not      |
|                         |       | empty. There is at least one           |
| LL attach Da O and to a | £.1.  | entry to get data from.                |
| UartIntrRxOverflow      | false | When the SCB mode is UART,             |
|                         |       | this parameter enables the             |
|                         |       | SCB.INTR_RX.OVERFLOW                   |
|                         |       | interrupt source. SCB.INTR RX.OVERFLOW |
|                         |       | trigger condition: attempt to          |
|                         |       | write to a full RX FIFO.               |
|                         | false | When the SCB mode is UART,             |
| UartIntrRxParityErr     | iaise | this parameter enables the             |
|                         |       | SCB.INTR_RX.PARITY                     |
|                         |       | ERROR interrupt source.                |
|                         |       | SCB.INTR_RX.PARITY                     |
|                         |       | ERROR trigger condition: parity        |
|                         |       | error in received data frame.          |
|                         |       | entor in received data maille.         |



| Parameter Name      | Value | Description  |
|---------------------|-------|--|
| UartIntrRxTrigger   | false | When the SCB mode is UART,                                       |
|                     |       | this parameter enables the                                       |
|                     |       | SCB.INTR_RX.TRIGGER  |
|                     |       | interrupt source.  |
|                     |       | SCB.INTR_RX.TRIGGER  |
|                     |       | trigger condition: remains active until RX FIFO has more entries |
|                     |       | than the value specified by                                      |
|                     |       | UartRxTriggerLevel.  |
| UartIntrRxUnderflow | false | When the SCB mode is UART,                                       |
|                     |       | this parameter enables the                                       |
|                     |       | SCB.INTR_RX.UNDERFLOW  |
|                     |       | interrupt source.  |
|                     |       | SCB.INTR_RX.UNDERFLOW  |
|                     |       | trigger condition: attempt to                                    |
|                     |       | read from an empty RX FIFO.                                      |
| UartIntrTxEmpty     | false | When the SCB mode is UART,                                       |
|                     |       | this parameter enables the                                       |
|                     |       | SCB.INTR_TX.EMPTY interrupt                                      |
|                     |       | SOURCE.  |
|                     |       | SCB.INTR_TX.EMPTY trigger condition: TX FIFO is empty.           |
| UartIntrTxNotFull   | false | When the SCB mode is UART,                                       |
| Oartinu i xivotruli | laise | this parameter enables the                                       |
|                     |       | SCB.INTR_TX.NOT_FULL   |
|                     |       | interrupt source.  |
|                     |       | SCB.INTR_TX.NOT_FULL   |
|                     |       | trigger condition: TX FIFO is not                                |
|                     |       | full. There is at least one entry                                |
|                     |       | to put data.   |
| UartIntrTxOverflow  | false | When the SCB mode is UART,                                       |
|                     |       | this parameter enables the                                       |
|                     |       | SCB.INTR_TX.OVERFLOW   |
|                     |       | interrupt source.  |
|                     |       | SCB.INTR_TX.OVERFLOW   |
|                     |       | trigger condition: attempt to                                    |
| UartIntrTxTrigger   | false | write to a full TX FIFO.  When the SCB mode is UART,             |
| Oatulu 1X1119961    | laise | this parameter enables the                                       |
|                     |       | SCB.INTR TX.TRIGGER  |
|                     |       | interrupt source.  |
|                     |       | SCB.INTR_TX.TRIGGER  |
|                     |       | trigger condition: remains active                                |
|                     |       | until TX FIFO has fewer entries                                  |
|                     |       | than the value specified by                                      |
|                     |       | UartTxTriggerLevel.  |
| UartIntrTxUartDone  | false | When the SCB mode is UART,                                       |
|                     |       | this parameter enables the                                       |
|                     |       | SCB.INTR_TX.UART_DONE  |
|                     |       | interrupt source. SCB.INTR_TX.UART_DONE                          |
|                     |       | trigger condition: all data are                                  |
|                     |       | sent in to TX FIFO and the                                       |
|                     |       | transmit FIFO and the shifter                                    |
|                     |       | register are emptied.  |
|                     |       | 5. 210. 5.10 5111ptious  |



| UartIntrTxUartLostArb  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_ARB_LOST interrupt source. SCB.INTR_TX.UART_ARB_LOST trigger condition: UART lost arbitration, the value driven on the TX line is not the same as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the UART into parameter inverts the incoming RX line signal. Only applicable for UART InDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter enables the UART incoming RX line signal. Only applicable for UART InDA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART incoming RX line signal. Only applicable for UART InDA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-processor mode. Only applicable for UART  | Parameter Name         | Value              | Description EMBEDDED IN TO              |
|--|------------------------|--------------------|---|
| this parameter enables the SCB.INTR_TX.UART_ARB_LOST interrupt source.  SCB.INTR_TX.UART_ARB_LOST interrupt source.  SCB.INTR_TX.UART_ARB_LOST interrupt source.  SCB.INTR_TX.UART_ARB_LOST tirger condition: UART lost arbitration, the value driven on the TX line is not the same as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line.  Only applicable for UART SmartCard mode.  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source.  SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement.  Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source.  SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option.  Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal.  Only applicable for UART IrDA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input inpu |                        |                    |   |
| SCB.INTR_TX.UART_ARB_LOST interrupt source. SCB.INTR_TX.UART_ARB_LOST interrupt source. SCB.INTR_TX.UART_ARB_LOST trigger condition. UART lost arbitration, the value driven on the TX line is not the same as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the condition: attempt to read from an empty TX FIFO.  Only applicable for UART in DA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART in DA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART in DA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART in DA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter inverts the UART multi-processor mode. Only applicable for UART Standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-processor mode. Only applicable for UART address into RX FIFO. Only applicable for UART multi-  |                        | laise              |   |
| LOST interrupt source. SCB.INTR. TX.UART_ARB LOST trigger condition: UART lost arbitration, the value driven on the TX line is not the same as the value observed on the TX line is not the same as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART inDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART inDA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART input line.  UartIMpEnable  false  When the SCB mode is UART Standard mode.  When the SCB mode is UART Standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| SCB.INTR_TX_UART_ARB LOST trigger condition: UART lost arbitration, the value driven on the TX line is not the same as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX_UART_NACK interrupt source. SCB.INTR_TX_UART_NACK interrupt source. SCB.INTR_TX_UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX_UNDERFLOW interrupt source. SCB.INTR_TX_UNDERFLOW trigger condition: attempt to read from an empty TX_FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartImpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartImpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartImpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART standard mode.  UartImpEnable  UartImpEnable  Talse  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO.  Only applicable for UART multi-  |                        |                    |   |
| LOST trigger condition: UART lost arbitration, the value driven on the TX line is not the same as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX_UART_NACK interrupt source. SCB.INTR_TX_UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX_UNDERFLOW interrupt source. SCB.INTR_TX_ |                        |                    |   |
| Iost arbitration, the value driven on the TX line is not the same as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.    UartIntrTxUartNack  |                        |                    |   |
| as the value observed on the RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART liDA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartImpEnable  false  When the SCB mode is UART, this parameter applies and digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART input line.   |                        |                    |   |
| RX line. This event is useful when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack false When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow false When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLO |                        |                    | on the TX line is not the same          |
| when the transmitter and the receiver share a TX/RX line. Only applicable for UART SmartCard mode. When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART irDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-processor mode.  |                        |                    | as the value observed on the            |
| Treceiver share a TX/RX line. Only applicable for UART SmartCard mode.  UartIntrTxUartNack  Idalse  UartIntrTxUartNack  Idalse  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source.  SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  Idalse  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source.  SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  Idalse  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART irDA mode.  UartIMedianFilterEnable  Idalse  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  Idalse  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  Idalse  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  Idalse  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  Idalse  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    | RX line. This event is useful           |
| UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART inDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART inDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART inDA mode.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line. UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    | when the transmitter and the            |
| UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source.  SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART irDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART irDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi- parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi- processor mode. Only applicable for UART multi- parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    | receiver share a TX/RX line.            |
| UartIntrTxUartNack  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UART_NACK interrupt source.  SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX_FIFO.  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line. UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-incoming RX FIFO. Only applicable for UART multi-increasor mode.  |                        |                    |   |
| this parameter enables the SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line. UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-incensor in UART multi-incen |                        |                    | SmartCard mode.                         |
| SCB.INTR_TX.UART_NACK interrupt source. SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  Only applicable for UART address into RX FIFO. Only applicable for UART multi-  | UartIntrTxUartNack     | false              | When the SCB mode is UART,              |
| interrupt source.  SCB.INTR_TX_UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX_UNDERFLOW interrupt source. SCB.INTR_TX_UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    | this parameter enables the              |
| SCB.INTR_TX.UART_NACK trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| trigger condition: UART transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartIMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartIMpEnable  false  When the SCB mode is UART, this parameter applies and it is parameter enables the UART multi-processor mode. Only applicable for UART standard mode.  UartIMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    |   |
| transmitter received a negative acknowledgement. Only applicable for UART SmartCard mode.  UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    |   |
| Continue   |                        |                    |   |
| UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART and mode.  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-processor into RX FIFO. Only applicable for UART multi-   |                        |                    |   |
| UartIntrTxUnderflow  false  When the SCB mode is UART, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    | 1                                       |
| this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source.  SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| SCB.INTR_TX.UNDERFLOW interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  Men the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  | UartIntrTxUnderflow    | talse              | , i                                     |
| interrupt source. SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  Mhen the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    |   |
| SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    | _                                       |
| trigger condition: attempt to read from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| Tead from an empty TX FIFO.  UartIrdaLowPower  false  When the SCB mode is UART, this parameter enables the low power receiver option.  Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal.  Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode.  Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode.  Only applicable for UART standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO.  Only applicable for UART multi-  |                        |                    |   |
| UartIrdaLowPower   false   When the SCB mode is UART, this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  |                        |                    |   |
| this parameter enables the low power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   | Hartirdal awDower      | folso              |   |
| power receiver option. Only applicable for UART IrDA mode.  UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART standard mode.  UartMpRxAcceptAddress  false  Only applicable for UART address into RX FIFO. Only applicable for UART multi-   | OartiidaLowFowei       | laise              | 1                                       |
| UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    | · ·                                     |
| UartIrdaPolarity  Non-Inverting  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| UartIrdaPolarity  Non-Inverting  When the SCB mode is UART, this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    | 1                                       |
| this parameter inverts the incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   | LlartIrdaPolarity      | Non-Inverting      |   |
| incoming RX line signal. Only applicable for UART IrDA mode.  UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  | Curtifical clarity     | 14011 IIIVOI aliig |   |
| UartMedianFilterEnable  false  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    | · · · · · · · · · · · · · · · · · · ·   |
| UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| UartMedianFilterEnable  false  When the SCB mode is UART, this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    | 1 ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' |
| this parameter applies a digital 3 tap median filter to the UART input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  | UartMedianFilterEnable | false              |   |
| UartMpEnable  false  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        | 15.100             |   |
| input line.  UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    |   |
| UartMpEnable  false  When the SCB mode is UART, this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO.  Only applicable for UART multi-   |                        |                    |   |
| this parameter enables the UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  | UartMpEnable           | false              | •                                       |
| UART multi-processor mode. Only applicable for UART Standard mode.  UartMpRxAcceptAddress false When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   | '                      |                    | , i                                     |
| Only applicable for UART Standard mode.  UartMpRxAcceptAddress  false  When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| UartMpRxAcceptAddress false When the SCB mode is UART, this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    |   |
| this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  |                        |                    |   |
| this parameter define whether to put the matched UART address into RX FIFO. Only applicable for UART multi-  | UartMpRxAcceptAddress  | false              | When the SCB mode is UART,              |
| put the matched UART address into RX FIFO. Only applicable for UART multi-   |                        |                    | , i                                     |
| Only applicable for UART multi-  |                        |                    |   |
|  |                        |                    | I ·                                     |
| nrocessor mode   |                        |                    | Only applicable for UART multi-         |
| processor mode.  |                        |                    | processor mode.                         |



| Parameter Name       | Value      | Description  |
|----------------------|------------|--|
| UartMpRxAddress      | 2          | When the SCB mode is UART,<br>this parameter defines the<br>UART address.<br>Only applicable for UART multi-<br>processor mode.  |
| UartMpRxAddressMask  | 255        | When the SCB mode is UART, this parameter defines the address mask in multiprocessor operation mode.  Bit value 0 – excludes bit from address comparison.  Bit value 1 – the bit needs to match with the corresponding bit of the UART address.  Only applicable for UART multiprocessor mode.               |
| UartNumberOfDataBits | 8 bits     | When the SCB mode is UART, this parameter defines the number of data bits inside the UART byte/word.   |
| UartNumberOfStopBits | 1 bit      | When the SCB mode is UART,<br>this parameter defines the<br>number of Stop bits.   |
| UartOvsFactor        | 12         | When the SCB mode is UART, this parameter defines the oversampling factor of SCBCLK.   |
| UartParityType       | None       | When the SCB mode is UART, this parameter applies UART parity check as Odd or Even or discards the parity entirely.  |
| UartRtsEnable        | false      | When the SCB mode is UART, this parameter enables the rts output.  Applicable only for devices other than PSoC 4000/PSoC 4100/PSoC 4200.   |
| UartRtsPolarity      | Active Low | When the SCB mode is UART, this parameter specifies active polarity of the output rts signal.  Applicable only for devices other than PSoC 4000/PSoC 4100/PSoC 4200.   |
| UartRtsTriggerLevel  | 4          | When the SCB mode is UART, this parameter specifies the number of entries in the RX FIFO to activate the rts output signal. When the receiver FIFO has fewer entries than the UartRtsTriggerLevel, an rts output signal is activated.  Applicable only for devices other than PSoC 4000/PSoC 4100/PSoC 4200. |
| UartRxBreakWidth     | 11         | This parameter specifies the break width in bits.  |



| Value     | Description   |
|-----------|---|
| 8         | When the SCB mode is UART,                                  |
|           | this parameter defines the size                             |
|           | of the RX buffer.   |
| false     | When the SCB mode is UART,                                  |
|           | this parameter enables the RX                               |
|           | trigger output terminal of the                              |
|           | component. This terminal must                               |
|           | be connected to the DMA input                               |
|           | trigger or left unconnected. Only                           |
|           | applicable for devices which have a DMA controller.         |
| 7         | When the SCB mode is UART,                                  |
|           | this parameter defines the                                  |
|           | number of entries in the RX                                 |
|           | FIFO to trigger control the                                 |
|           | SCB.INTR_RX.TRIGGER   |
|           | interrupt event or RX DMA                                   |
|           | trigger output.   |
| false     | When the SCB mode is UART,                                  |
|           | this parameter defines whether                              |
|           | to send a message again when a NACK response is received.   |
|           | Only applicable for UART                                    |
|           | SmartCard mode.   |
| Standard  | When the SCB mode is UART,                                  |
| Otaridard | this parameter defines the sub                              |
|           | mode of UART as: Standard,                                  |
|           | SmartCard or IrDA.  |
| 8         | When the SCB mode is UART,                                  |
|           | this parameter defines the size                             |
|           | of the TX buffer.   |
| false     | When the SCB mode is UART,                                  |
|           | this parameter enables the TX                               |
|           | trigger output terminal of the                              |
|           | component. This terminal must be connected to the DMA input |
|           | trigger or left unconnected. Only                           |
|           | applicable for devices which                                |
|           | have a DMA controller.                                      |
| 0         | When the SCB mode is UART,                                  |
|           | this parameter defines the                                  |
|           | number of entries in the TX                                 |
|           | FIFO to control the SCB.INTR                                |
|           | TX.TRIGGER interrupt event or                               |
|           | TX DMA trigger output.                                      |
| talse     | When the SCB mode is UART,                                  |
|           | this parameter enables the                                  |
|           | wakeup from Deep Sleep on start bit event. The actual       |
|           | wakeup source is RX GPIO.                                   |
|           | The skip start UART feature                                 |
|           |   |
|           | •   |
|           | allows it to continue receiving bytes.                      |
|           | false  false  Standard  8  false                            |

## 9.5.2 Instance UART

**Description: Serial Communication Block (SCB)** 



Instance type: SCB\_P4 [v4.0]
Datasheet: online component datasheet for SCB\_P4

Table 22. Component Parameters for UART

| Parameter Name             | Value | Description  |
|----------------------------|-------|--|
| Ezl2cByteModeEnable        | false | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter specifies the                                 |
|                            |       | number of bits per FIFO data                                 |
|                            |       | element.   |
|                            |       | The byte mode – false: a 16-bit                              |
|                            |       | FIFO data element. The FIFO                                  |
|                            |       | depth is 8 entries.  |
|                            |       | The byte mode – true: an 8-bit                               |
|                            |       | FIFO data element. The FIFO                                  |
|                            |       | depth is 16 entries.   |
|                            |       |  |
|                            |       | Applicable only for devices                                  |
|                            |       | other than PSoC 4000/PSoC                                    |
|                            |       | 4100/PSoC 4200.  |
| Ezl2cClockFromTerm         | false | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter provides a clock                              |
|                            |       | terminal to connect a clock                                  |
|                            |       | outside the component.                                       |
| Ezl2cClockStretching       | true  | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter specifies                                     |
|                            |       | whether the SCL is stretched                                 |
|                            |       | while in EZI2C operation.                                    |
| Ezl2cDataRate              | 100   | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter defines EZI2C                                 |
|                            |       | Data rate in kbps. The standard                              |
|                            |       | data rates are: 100, 400 and                                 |
|                            |       | 1000 kbps.   |
| Ezl2cNumberOfAddresses     | 1     | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter defines the                                   |
|                            |       | number of I2C slave addresses                                |
|                            |       | that device respond to.                                      |
| Ezl2cPrimarySlaveAddress   | 8     | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter specifies EZI2C                               |
|                            |       | primary 7-bits slave address                                 |
|                            | _     | (MSB ignored).   |
| Ezl2cSecondarySlaveAddress | 9     | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter specifies EZI2C                               |
|                            |       | secondary 7-bits slave address                               |
|                            |       | (MSB ignored).   |
|                            |       | Only applicable when EZI2C                                   |
| F-10, O. I. A. I. I O.     | _     | clock stretching option is set.                              |
| Ezl2cSubAddressSize        | 8     | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter specifies the                                 |
|                            |       | maximum size of the slave                                    |
|                            |       | buffer that is exposed to the                                |
|                            |       | master: 8bits – maximum buffer                               |
|                            |       | size is 256 bytes, 16 bits –<br>maximum buffer size is 65535 |
|                            |       | bytes.   |
| Ezi2eWekeEneble            | foloo | -  |
| Ezl2cWakeEnable            | false | When the SCB mode is EZI2C,                                  |
|                            |       | this parameter enables wakeup                                |
|                            |       | from Deep Sleep on I2C address match event.                  |
|                            |       | address match event.   |



| Parameter Name       | Value | Doscription                                     |
|----------------------|-------|---|
| I2C Bus Voltage      | 3.3   | Description When the SCB mode is I2C, this      |
| IZO bus voltage      | 3.3   | parameter specifies the voltage                 |
|                      |       | applied to the pull-up resistors                |
|                      |       | on the I2C bus.                                 |
|                      |       | 011 the 120 bde.                                |
|                      |       | Only applicable for devices                     |
|                      |       | other than PSoC 4000/PSoC                       |
|                      |       | 4100/PSoC 4200.                                 |
| I2C Bus Voltage      | 3.3   | When the SCB mode is EZI2C,                     |
|                      |       | this parameter specifies the                    |
|                      |       | voltage applied to the pull-up                  |
|                      |       | resistors on the I2C bus.                       |
|                      |       |   |
|                      |       | Only applicable for devices                     |
|                      |       | other than PSoC 4000/PSoC                       |
|                      |       | 4100/PSoC 4200.                                 |
| I2cAcceptAddress     | false | When the SCB mode is I2C, this                  |
|                      |       | parameter specifies whether to                  |
|                      |       | accept the match slave address                  |
|                      |       | in RX FIFO or not. All slave                    |
|                      |       | matched addresses are ACKed.                    |
|                      |       | The user has to register the                    |
|                      |       | callback function to handle                     |
|                      |       | accepted addresses. This                        |
|                      |       | feature has to be used when                     |
|                      |       | more than one address support                   |
|                      |       | is required.                                    |
| I2cAcceptGeneralCall | false | When the SCB mode is I2C, this                  |
|                      |       | parameter specifies whether to                  |
|                      |       | accept the general call address.                |
|                      |       | The general call address is                     |
|                      |       | ACKed when accepted and                         |
|                      |       | NAKed otherwise. The user has                   |
|                      |       | to register the callback function               |
|                      |       | to handle the general call                      |
| 10.5 / 14 / 5 / /    |       | address.  |
| I2cByteModeEnable    | false | When the SCB mode is I2C, this                  |
|                      |       | parameter specifies the number                  |
|                      |       | of bits per FIFO data element.                  |
|                      |       | The byte mode – false: a 16-bit                 |
|                      |       | FIFO data element. The FIFO depth is 8 entries. |
|                      |       | The byte mode – true: an 8-bit                  |
|                      |       | FIFO data element. The FIFO                     |
|                      |       | depth is 16 entries.                            |
|                      |       | dopario io citales.                             |
|                      |       | Applicable only for devices                     |
|                      |       | other than PSoC 4000/PSoC                       |
|                      |       | 4100/PSoC 4200.                                 |
| I2cClockFromTerm     | false | When the SCB mode is I2C, this                  |
| 120.3300 13001       | 13100 | parameter provides a clock                      |
|                      |       | terminal to connect a clock                     |
|                      |       | outside the component.                          |
| I2cDataRate          | 100   | When the SCB mode is I2C, this                  |
|                      | 100   | parameter specifies the data                    |
|                      |       | rate in kbps. The standard data                 |
|                      |       | rates are: 100, 400 and 1000                    |
|                      |       | kbps.   |
|                      | 1     | Nopo.   |



| Parameter Name             | Value | Description  |
|----------------------------|-------|--|
| I2cExternIntrHandler       | false | When the SCB mode is I2C, this parameter specifies whether the I2C interrupt handler is  |
|                            |       | configured in SCB_I2CInit(). This parameter is intended to be used by the PM/SM bus component. The modification  |
| 10 M 10                    | 4     | parameter default value causes I2C mode failures.  |
| I2cManualOversampleControl | true  | When the SCB mode is I2C, this parameter specifies the method of calculating the oversampling as manual or automatic.  |
| I2cMode                    | Slave | When the SCB mode is I2C, this parameter defines the I2C operation mode as: Slave, Master, Multi-Master or Multi-Master-Slave.   |
| I2cOvsFactor               | 16    | When the SCB mode is I2C, this parameter defines the oversampling factor of SCBCLK.  |
| I2cOvsFactorHigh           | 8     | When the SCB mode is I2C, this parameter defines the high oversampling factor of SCBCLK. Only applicable for I2C Master modes.   |
| I2cOvsFactorLow            | 8     | When the SCB mode is I2C, this parameter defines the low oversampling factor of SCBCLK. Only applicable for I2C Master modes.  |
| I2cSlaveAddress            | 8     | When the SCB mode is I2C, this parameter specifies the I2C 7-bits slave address (MSB ignored).   |
| I2cSlaveAddressMask        | 254   | When the SCB mode is I2C, this parameter specifies the I2C Slave address mask.  Bit value 0 – excludes bit from address comparison.  Bit value 1 – the bit needs to match with the corresponding bit of the I2C slave address. |
| I2cWakeEnable              | false | When the SCB mode is I2C, this parameter enables wakeup from Deep Sleep on an I2C address match event.   |
| ScbMisoSdaTxEnable         | true  | This parameter defines the availability of the spi_miso_i2csda_uart_tx pin.  |
| ScbMode                    | UART  | This parameter defines the mode of operation for the SCB component.  |
| ScbMosiSclRxEnable         | true  | This parameter defines the availability of the spi_mosi_i2cscl_uart_rx pin.  |



| Parameter Name             | Value | Description  |
|----------------------------|-------|--|
| ScbRxWakeIrqEnable         | false | This parameter defines the                               |
| - Cost attraction quitable | laio  | availability of the spi_mosi_i2c                         |
|                            |       | scl_uart_rx_wake pin.                                    |
| ScbSclkEnable              | false | This parameter defines the                               |
| CODCORLINGBIO              | laise | availability of the sclk pin.                            |
| ScbSs0Enable               | false | This parameter defines the                               |
| SCDOSULTIABLE              | laise | availability of the ss0 pin.                             |
| ScbSs1Enable               | false | This parameter defines the                               |
| Schostellable              | laise | availability of the ss1 pin.                             |
| ScbSs2Enable               | false |  |
| SCDSSZENADIE               | laise | This parameter defines the                               |
| 0.1.0.0511.                | f.1.  | availability of the ss2 pin.                             |
| ScbSs3Enable               | false | This parameter defines the                               |
|                            |       | availability of the ss3 pin.                             |
| Show EZI2C Terminals       | false | When the SCB mode is EZI2C,                              |
|                            |       | this parameter removes internal                          |
|                            |       | pins and expose signals to                               |
|                            |       | terminals. The exposed                                   |
|                            |       | terminals must be connected to                           |
| OL 100 T : 1               |       | the pins.  |
| Show I2C Terminals         | false | When the SCB mode is I2C, this                           |
|                            |       | parameter removes internal pins                          |
|                            |       | and expose signals to terminals.                         |
|                            |       | The exposed terminals must be                            |
| 01 001 7 1                 |       | connected to the pins.                                   |
| Show SPI Terminals         | false | When the SCB mode is SPI,                                |
|                            |       | this parameter removes internal                          |
|                            |       | pins and expose signals to                               |
|                            |       | terminals. The exposed                                   |
|                            |       | terminals must be connected to                           |
| Observation II             | 6.1.  | the pins or SmartIO component.                           |
| Show UART Terminals        | false | When the SCB mode is UART,                               |
|                            |       | this parameter removes internal                          |
|                            |       | pins and expose signals to terminals. The exposed        |
|                            |       | terminals. The exposed                                   |
|                            |       | the pins or SmartIO component.                           |
| Slew Rate                  | Fast  |  |
| Siew Rate                  | rasi  | When the SCB mode is EZI2C, this parameter specifies the |
|                            |       | slew rate settings of the I2C                            |
|                            |       | pins.  |
|                            |       | For devices supporting GPIO                              |
|                            |       | Over-Voltage Tolerance                                   |
|                            |       | (GPIO_OVT) pins, I2C FM+                                 |
|                            |       | options should be used when                              |
|                            |       | I2C data rate is greater than                            |
|                            |       | 400 kbps. This option also                               |
|                            |       | requires the I2C bus voltage to                          |
|                            |       | be defined.Refer to the Device                           |
|                            |       | Datasheet to determine which                             |
|                            |       | pins are GPIO_OVT capable.                               |
|                            |       |  |



| Parameter Name     | Value     | Description                       |
|--------------------|-----------|-----------------------------------|
| Slew Rate          | Fast      | When the SCB mode is I2C, this    |
|                    |           | parameter specifies the slew      |
|                    |           | rate settings of the I2C pins.    |
|                    |           | For devices supporting GPIO       |
|                    |           | Over-Voltage Tolerance            |
|                    |           | (GPIO_OVT) pins, I2C FM+          |
|                    |           | options should be used when       |
|                    |           | I2C data rate is greater than     |
|                    |           | 400 kbps. This option also        |
|                    |           | requires the I2C bus voltage to   |
|                    |           | be defined. Refer to the Device   |
|                    |           | Datasheet to determine which      |
|                    |           | pins are GPIO_OVT capable.        |
| SpiBitRate         | 1000      | When the SCB mode is SPI,         |
| Opibilitate        | 1000      | this parameter specifies the Bit  |
|                    |           | rate in kbps (up to 8000 kbps);   |
|                    |           | the actual rate may differ based  |
|                    |           | on available clock frequency      |
|                    |           | and component settings. This      |
|                    |           | parameter has no effect if the    |
|                    |           | Clock from terminal parameter     |
|                    |           | is enabled.                       |
| CniDitaOrdor       | MSB First | When the SCB mode is SPI,         |
| SpiBitsOrder       | MSB FIISI | · 1                               |
|                    |           | this parameter defines the bit    |
| 0.10 + M + 5 + H   |           | order as: MSB first or LSB first. |
| SpiByteModeEnable  | false     | When the SCB mode is SPI,         |
|                    |           | this parameter specifies the      |
|                    |           | number of bits per FIFO data      |
|                    |           | element.                          |
|                    |           | The byte mode – false: a 16-bit   |
|                    |           | FIFO data element. The FIFO       |
|                    |           | depth is 8 entries.               |
|                    |           | The byte mode – true: an 8-bit    |
|                    |           | FIFO data element. The FIFO       |
|                    |           | depth is 16 entries.              |
|                    |           | Ameliaahla amkufan dayisaa        |
|                    |           | Applicable only for devices       |
|                    |           | other than PSoC 4000/PSoC         |
| 0.:0115            | 6.1       | 4100/PSoC 4200.                   |
| SpiClockFromTerm   | false     | When the SCB mode is SPI,         |
|                    |           | this parameter provides a clock   |
|                    |           | terminal to connect a clock       |
|                    | 1         | outside the component.            |
| SpiFreeRunningSclk | false     | When the SCB mode is SPI,         |
|                    |           | this parameter specifies the      |
|                    |           | SCLK generation by the master     |
|                    |           | as: gated or free running         |
|                    |           | (continuous).                     |
|                    |           |                                   |
|                    |           | Applicable only for devices       |
|                    |           | other than PSoC 4000/PSoC         |
|                    |           | 4100/PSoC 4200.                   |



| Parameter Name             | Value | Description Description           |
|----------------------------|-------|-----------------------------------|
| SpilnterruptMode           | None  | When the SCB mode is SPI,         |
| Opiniterruptiviode         | None  | this parameter specifies the      |
|                            |       | interrupt mode. None: Removes     |
|                            |       | all interrupt support. Internal:  |
|                            |       | Leaves the interrupt SCBIRQ       |
|                            |       | inside the component - the        |
|                            |       | interrupt terminal becomes        |
|                            |       | invisible. External: Provides an  |
|                            |       | interrupt terminal to connect an  |
|                            |       | interrupt outside the component.  |
| SpiIntrMasterSpiDone       | false | When the SCB mode is SPI,         |
| - Spiriti Master Spizsorio | 14100 | this parameter enables the        |
|                            |       | SCB.INTR M. SPI DONE              |
|                            |       | interrupt source.                 |
|                            |       | SCB.INTR M. SPI DONE: all         |
|                            |       | data are sent into TX FIFO and    |
|                            |       | the TX FIFO and the shifter       |
|                            |       | register are emptied.             |
|                            |       | Only applicable for SPI Master    |
|                            |       | mode.                             |
| SpiIntrRxFull              | false | When the SCB mode is SPI,         |
| Opiniar (XI dii            | laise | this parameter enables the        |
|                            |       | SCB.INTR_RX.FULL interrupt        |
|                            |       | source.                           |
|                            |       | SCB.INTR_RX.FULL trigger          |
|                            |       | condition: RX FIFO is full.       |
| SpiIntrRxNotEmpty          | false | When the SCB mode is SPI,         |
| Opiniartxittotempty        | laise | this parameter enables the        |
|                            |       | SCB.INTR RX.NOT EMPTY             |
|                            |       | interrupt source.                 |
|                            |       | SCB.INTR RX.NOT EMPTY             |
|                            |       | trigger condition: RX FIFO is not |
|                            |       | empty. There is at least one      |
|                            |       | entry to get data from.           |
| SpiIntrRxOverflow          | false | When the SCB mode is SPI,         |
|                            | 14.55 | this parameter enables the        |
|                            |       | SCB.INTR RX.OVERFLOW              |
|                            |       | interrupt source.                 |
|                            |       | SCB.INTR RX.OVERFLOW              |
|                            |       | trigger condition: attempt to     |
|                            |       | write to a full RX FIFO.          |
| SpilntrRxTrigger           | false | When the SCB mode is SPI,         |
| - F 2                      |       | this parameter enables the        |
|                            |       | SCB.INTR RX.TRIGGER               |
|                            |       | interrupt source.                 |
|                            |       | SCB.INTR RX.TRIGGER               |
|                            |       | trigger condition: remains active |
|                            |       | until RX FIFO has more entries    |
|                            |       | than the value specified by       |
|                            |       | SpiRxTriggerLevel.                |
| SpiIntrRxUnderflow         | false | When the SCB mode is SPI,         |
| •                          |       | this parameter enables the        |
|                            |       | SCB.INTR RX.UNDERFLOW             |
|                            |       | interrupt source.                 |
|                            |       | SCB.INTR RX.UNDERFLOW             |
|                            |       |                                   |
|                            |       | trigger condition: attempt to     |



| Parameter Name          | Value | Description  |
|-------------------------|-------|--|
| SpiIntrSlaveBusError    | false | When the SCB mode is SPI, this parameter enables the SCB.INTR_SLAVE.BUSERROR interrupt source. SCB.INTR_SLAVE.BUSERROR trigger condition: slave select line is deselected at an unexpected time in the SPI transfer. Only applicable for SPI Slave mode. |
| SpilntrTxEmpty          | false | When the SCB mode is SPI, this parameter enables the SCB.INTR_TX.EMPTY interrupt source.  SCB.INTR_TX.EMPTY trigger condition: TX FIFO is empty.   |
| SpiIntrTxNotFull        | false | When the SCB mode is SPI, this parameter enables the SCB.INTR_TX.NOT_FULL interrupt source.  SCB.INTR_TX.NOT_FULL trigger condition: TX FIFO is not full. There is at least one entry to put data.   |
| SpiIntrTxOverflow       | false | When the SCB mode is SPI, this parameter enables the SCB.INTR_TX.OVERFLOW interrupt source. SCB.INTR_TX.OVERFLOW trigger condition: attempt to write to a full TX FIFO.  |
| SpiIntrTxTrigger        | false | When the SCB mode is SPI, this parameter enables the SCB.INTR_TX.TRIGGER interrupt source. SCB.INTR_TX.TRIGGER trigger condition: remains active until TX FIFO has fewer entries than the value specified by SpiTxTriggerLevel.                          |
| SpilntrTxUnderflow      | false | When the SCB mode is SPI, this parameter enables the SCB.INTR_TX.UNDERFLOW interrupt source.  SCB.INTR_TX.UNDERFLOW trigger condition: attempt to read from an empty TX FIFO.  |
| SpiLateMisoSampleEnable | false | When the SCB mode is SPI,<br>this parameter enables late<br>sampling of the MISO line by<br>the master.  |
| SpiMedianFilterEnable   | false | When the SCB mode is SPI,<br>this parameter applies a digital<br>3 tap median filter to the SPI<br>input line.   |



|                        |                | EMBEDDED IN TO                    |
|------------------------|----------------|-----------------------------------|
| Parameter Name         | Value          | Description                       |
| SpiMode                | Slave          | When the SCB mode is SPI,         |
|                        |                | this parameter selects SPI        |
|                        |                | mode of operation as: Slave or    |
|                        |                | Master.                           |
| SpiNumberOfRxDataBits  | 8              | When the SCB mode is SPI,         |
|                        |                | this parameter specifies the      |
|                        |                | number of data bits inside the    |
|                        |                | SPI byte/word for RX direction.   |
| SpiNumberOfSelectLines | 1              | When the SCB mode is SPI,         |
|                        |                | this parameter defines the        |
|                        |                | number of slave select lines.     |
|                        |                | The SPI Slave has only one        |
|                        |                | slave select line. The SPI        |
|                        |                | Master has up to 4 lines.         |
| SpiNumberOfTxDataBits  | 8              | When the SCB mode is SPI,         |
|                        |                | this parameter define the         |
|                        |                | number of data bits inside the    |
|                        |                | SPI byte/word for TX direction.   |
| SpiOvsFactor           | 16             | When the SCB mode is SPI,         |
|                        |                | this parameter defines the        |
|                        |                | oversampling factor of            |
|                        |                | SĆBCĽK.                           |
| SpiRemoveMiso          | false          | When the SCB mode is SPI,         |
| '                      |                | this parameter removes the        |
|                        |                | MISO pin.                         |
| SpiRemoveMosi          | false          | When the SCB mode is SPI,         |
|                        | laice          | this parameter removes the        |
|                        |                | MOSI pin.                         |
| SpiRemoveSclk          | false          | When the SCB mode is SPI,         |
| Spir tomove Som        | laice          | this parameter removes the        |
|                        |                | SCLK pin.                         |
| SpiRxBufferSize        | 8              | When the SCB mode is SPI,         |
| Opii WBuiloroi20       |                | this parameter defines the size   |
|                        |                | of the RX buffer.                 |
| SpiRxOutputEnable      | false          | When the SCB mode is SPI,         |
|                        | laise          | this parameter enables the RX     |
|                        |                | trigger output terminal of the    |
|                        |                | component. This terminal must     |
|                        |                | be connected to the DMA input     |
|                        |                | trigger or left unconnected. Only |
|                        |                | applicable for devices which      |
|                        |                | have a DMA controller.            |
| SpiRxTriggerLevel      | 7              | When the SCB mode is SPI,         |
| Opil Williggor Level   | '              | this parameter defines the        |
|                        |                | number of entries in the RX       |
|                        |                | FIFO to control the SCB.INTR -    |
|                        |                | RX.TRIGGER interrupt event or     |
|                        |                | RX DMA trigger output.            |
| SpiSclkMode            | CPHA = 0, CPOL | When the SCB mode is SPI,         |
| - Opioolitivious       | = 0            | this parameter defines the serial |
|                        |                | clock phase (CPHA) and            |
|                        |                | polarity (CPOL).                  |
| SpiSs0Polarity         | Active Low     | When the SCB mode is SPI,         |
| Opiosor clarity        | VOUAG FOM      | this parameter specifies active   |
|                        |                | polarity of slave select 0.       |
|                        |                | polarity of slave select o.       |
|                        |                | Applicable only for devices       |
|                        |                | other than PSoC 4000/PSoC         |
|                        |                | 4100/PSoC 4200.                   |
|                        | 1              | 1100/1 000 4200.                  |



| Parameter Name        | Value      | Description  |
|-----------------------|------------|--|
| SpiSs1Polarity        | Active Low | When the SCB mode is SPI,<br>this parameter specifies active<br>polarity of slave select 1.  |
|                       |            | Applicable only for devices other than PSoC 4000/PSoC 4100/PSoC 4200.  |
| SpiSs2Polarity        | Active Low | When the SCB mode is SPI, this parameter specifies active polarity of slave select 2.  |
|                       |            | Applicable only for devices other than PSoC 4000/PSoC 4100/PSoC 4200.  |
| SpiSs3Polarity        | Active Low | When the SCB mode is SPI, this parameter specifies active polarity of slave select 3.  Applicable only for devices other than PSoC 4000/PSoC   |
|                       |            | 4100/PSoC 4200.  |
| SpiSubMode            | Motorola   | When the SCB mode is SPI, this parameter defines the sub mode of the SPI as: Motorola, TI(Start Coincides), TI(Start Precedes), or National Semiconductor.   |
| SpiTransferSeparation | Continuous | When the SCB mode is SPI,<br>this parameter defines the type<br>of SPI transfers separation as:<br>continuous or separated.  |
| SpiTxBufferSize       | 8          | When the SCB mode is SPI,<br>this parameter defines the size<br>of the TX buffer.  |
| SpiTxOutputEnable     | false      | When the SCB mode is SPI, this parameter enables the TX trigger output terminal of the component. This terminal must be connected to the DMA input trigger or left unconnected. Only applicable for devices which have a DMA controller. |
| SpiTxTriggerLevel     | 0          | When the SCB mode is SPI, this parameter defines the number of entries in the TX FIFO to control the SCB.INTRTX.TRIGGER interrupt event or TX DMA trigger output.  |
| SpiWakeEnable         | false      | When the SCB mode is SPI,<br>this parameter enables wakeup<br>from Deep Sleep on slave<br>select event.  |



| Parameter Name         | Value      | Description Description                                     |
|------------------------|------------|---|
| UartByteModeEnable     | false      | When the SCB mode is UART,                                  |
| oartbyteiwode⊑nable    | laise      | this parameter specifies the                                |
|                        |            | number of bits per FIFO data                                |
|                        |            | element.  |
|                        |            |   |
|                        |            | The byte mode – false: a 16-bit FIFO data element. The FIFO |
|                        |            |   |
|                        |            | depth is 8 entries.   |
|                        |            | The byte mode – true: an 8-bit                              |
|                        |            | FIFO data element. The FIFO                                 |
|                        |            | depth is 16 entries.  |
|                        |            | Applicable only for devices                                 |
|                        |            | Applicable only for devices                                 |
|                        |            | other than PSoC 4000/PSoC                                   |
|                        |            | 4100/PSoC 4200.   |
| UartClockFromTerm      | false      | When the SCB mode is UART,                                  |
|                        |            | this parameter provides a clock                             |
|                        |            | terminal to connect a clock                                 |
|                        |            | outside the component.                                      |
| UartCtsEnable          | false      | When the SCB mode is UART,                                  |
|                        |            | this parameter enables the cts                              |
|                        |            | input.  |
|                        |            |   |
|                        |            | Only applicable for devices                                 |
|                        |            | other than PSoC 4000/PSoC                                   |
|                        |            | 4100/PSoC 4200.   |
| UartCtsPolarity        | Active Low | When the SCB mode is UART,                                  |
|                        |            | this parameter specifies active                             |
|                        |            | polarity of an input cts signal.                            |
|                        |            |   |
|                        |            | Only applicable for devices                                 |
|                        |            | other than PSoC 4000/PSoC                                   |
|                        |            | 4100/PSoC 4200.   |
| UartDataRate           | 115200     | When the SCB mode is UART,                                  |
|                        |            | this parameter specifies the                                |
|                        |            | Baud rate in bps (up to 1000                                |
|                        |            | kbps); the actual rate may differ                           |
|                        |            | based on available clock                                    |
|                        |            | frequency and component                                     |
|                        |            | settings. This parameter has no                             |
|                        |            | effect if the Clock from terminal                           |
|                        |            | parameter is enabled.                                       |
| UartDirection          | TX + RX    | When the SCB mode is UART,                                  |
|                        |            | this parameter enables RX or                                |
|                        |            | TX direction or both  |
|                        |            | simultaneously.   |
| UartDropOnFrameErr     | false      | When the SCB mode is UART,                                  |
| Ca. as a point rainoun | 10100      | this parameter defines whether                              |
|                        |            | the data is dropped from RX                                 |
|                        |            | FIFO on a frame error event.                                |
| UartDropOnParityErr    | false      | When the SCB mode is UART,                                  |
|                        | laise      | this parameter determines                                   |
|                        |            | whether the data is dropped                                 |
|                        |            | from RX FIFO on a parity error                              |
|                        |            | event.  |
|                        |            | EVELIL.   |



| Parameter Name          | Value | Description   |
|-------------------------|-------|---|
| UartInterruptMode       | None  | When the SCB mode is UART, this parameter specifies the interrupt mode. None: Removes all interrupt support. Internal: Leaves the interrupt SCBIRQ inside the component - the interrupt terminal becomes invisible. External: Provides an interrupt terminal to connect an interrupt outside component. |
| UartIntrRxBreakDetected | false | This parameter enables the RX break detection interrupt source to trigger the interrupt output.   |
| UartIntrRxFrameErr      | false | When the SCB mode is UART, this parameter enables the SCB.INTR_RX.FRAMEERROR interrupt source. SCB.INTR_RX.FRAMEERROR trigger condition: frame error in received data frame.  |
| UartIntrRxFull          | false | When the SCB mode is UART, this parameter enables the SCB.INTR_RX.FULL interrupt source.  SCB.INTR_RX.FULL trigger condition: RX FIFO is full.  |
| UartIntrRxNotEmpty      | false | When the SCB mode is UART, this parameter enables the SCB.INTR_RX.NOT_EMPTY interrupt source.  SCB.INTR_RX.NOT_EMPTY trigger condition: RX FIFO is not empty. There is at least one entry to get data from.   |
| UartIntrRxOverflow      | false | When the SCB mode is UART, this parameter enables the SCB.INTR_RX.OVERFLOW interrupt source. SCB.INTR_RX.OVERFLOW trigger condition: attempt to write to a full RX FIFO.  |
| UartIntrRxParityErr     | false | When the SCB mode is UART, this parameter enables the SCB.INTR_RX.PARITY ERROR interrupt source. SCB.INTR_RX.PARITY ERROR trigger condition: parity error in received data frame.   |
| UartIntrRxTrigger       | false | When the SCB mode is UART, this parameter enables the SCB.INTR_RX.TRIGGER interrupt source.  SCB.INTR_RX.TRIGGER trigger condition: remains active until RX FIFO has more entries than the value specified by UartRxTriggerLevel.   |



| =                              | 1                | EMBEDDED IN TO  |
|--------------------------------|------------------|---|
| Parameter Name                 | Value            | Description   |
| UartIntrRxUnderflow            | false            | When the SCB mode is UART,                            |
|                                |                  | this parameter enables the                            |
|                                |                  | SCB.INTR_RX.UNDERFLOW                                 |
|                                |                  | interrupt source.                                     |
|                                |                  | SCB.INTR_RX.UNDERFLOW                                 |
|                                |                  | trigger condition: attempt to                         |
| L Louis Institutor Consensitor | false            | read from an empty RX FIFO.                           |
| UartIntrTxEmpty                | laise            | When the SCB mode is UART, this parameter enables the |
|                                |                  | SCB.INTR_TX.EMPTY interrupt                           |
|                                |                  | source.   |
|                                |                  | SCB.INTR_TX.EMPTY trigger                             |
|                                |                  | condition: TX FIFO is empty.                          |
| UartIntrTxNotFull              | false            | When the SCB mode is UART,                            |
|                                | 15.155           | this parameter enables the                            |
|                                |                  | SCB.INTR TX.NOT FULL                                  |
|                                |                  | interrupt source.                                     |
|                                |                  | SCB.INTR TX.NOT FULL                                  |
|                                |                  | trigger condition: TX FIFO is not                     |
|                                |                  | full. There is at least one entry                     |
|                                |                  | to put data.  |
| UartIntrTxOverflow             | false            | When the SCB mode is UART,                            |
|                                |                  | this parameter enables the                            |
|                                |                  | SCB.INTR_TX.OVERFLOW                                  |
|                                |                  | interrupt source.                                     |
|                                |                  | SCB.INTR_TX.OVERFLOW                                  |
|                                |                  | trigger condition: attempt to                         |
| 11.0.4.7.7.                    |                  | write to a full TX FIFO.                              |
| UartIntrTxTrigger              | false            | When the SCB mode is UART,                            |
|                                |                  | this parameter enables the                            |
|                                |                  | SCB.INTR_TX.TRIGGER interrupt source.                 |
|                                |                  | SCB.INTR_TX.TRIGGER                                   |
|                                |                  | trigger condition: remains active                     |
|                                |                  | until TX FIFO has fewer entries                       |
|                                |                  | than the value specified by                           |
|                                |                  | UartTxTriggerLevel.                                   |
| UartIntrTxUartDone             | false            | When the SCB mode is UART,                            |
|                                |                  | this parameter enables the                            |
|                                |                  | SCB.INTR_TX.UART_DONE                                 |
|                                |                  | interrupt source.                                     |
|                                |                  | SCB.INTR_TX.UART_DONE                                 |
|                                |                  | trigger condition: all data are                       |
|                                |                  | sent in to TX FIFO and the                            |
|                                |                  | transmit FIFO and the shifter register are emptied.   |
| UartIntrTxUartLostArb          | false            | When the SCB mode is UART,                            |
| Cartifili i XOai (LOStAID      | laise            | this parameter enables the                            |
|                                |                  | SCB.INTR_TX.UART_ARB                                  |
|                                |                  | LOST interrupt source.                                |
|                                |                  | SCB.INTR_TX.UART_ARB                                  |
|                                |                  | LOST trigger condition: UART                          |
|                                |                  | lost arbitration, the value driven                    |
|                                |                  | on the TX line is not the same                        |
|                                |                  | as the value observed on the                          |
|                                |                  | RX line. This event is useful                         |
|                                |                  | when the transmitter and the                          |
|                                |                  | receiver share a TX/RX line.                          |
|                                |                  | Only applicable for UART                              |
|                                | 07/05/0000 40:00 | SmartCard mode.                                       |



| Parameter Name         | Value         | Description   |
|------------------------|---------------|---|
| UartIntrTxUartNack     | false         | When the SCB mode is UART,                                    |
|                        |               | this parameter enables the                                    |
|                        |               | SCB.INTR_TX.UART_NACK   |
|                        |               | interrupt source. SCB.INTR_TX.UART_NACK                       |
|                        |               | trigger condition: UART                                       |
|                        |               | transmitter received a negative                               |
|                        |               | acknowledgement.  |
|                        |               | Only applicable for UART                                      |
|                        |               | SmartCard mode.   |
| UartIntrTxUnderflow    | false         | When the SCB mode is UART, this parameter enables the         |
|                        |               | SCB.INTR TX.UNDERFLOW   |
|                        |               | interrupt source.   |
|                        |               | SCB.INTR_TX.UNDERFLOW   |
|                        |               | trigger condition: attempt to                                 |
| Hardada Danie          | 6.1.          | read from an empty TX FIFO.                                   |
| UartIrdaLowPower       | false         | When the SCB mode is UART, this parameter enables the low     |
|                        |               | power receiver option.  |
|                        |               | Only applicable for UART IrDA                                 |
|                        |               | mode.   |
| UartIrdaPolarity       | Non-Inverting | When the SCB mode is UART,                                    |
|                        |               | this parameter inverts the                                    |
|                        |               | incoming RX line signal. Only applicable for UART IrDA        |
|                        |               | mode.   |
| UartMedianFilterEnable | false         | When the SCB mode is UART,                                    |
|                        |               | this parameter applies a digital                              |
|                        |               | 3 tap median filter to the UART                               |
| Llouth Au Euroble      | f-1           | input line.   |
| UartMpEnable           | false         | When the SCB mode is UART, this parameter enables the         |
|                        |               | UART multi-processor mode.                                    |
|                        |               | Only applicable for UART                                      |
|                        |               | Standard mode.  |
| UartMpRxAcceptAddress  | false         | When the SCB mode is UART,                                    |
|                        |               | this parameter define whether to put the matched UART address |
|                        |               | into RX FIFO.   |
|                        |               | Only applicable for UART multi-                               |
|                        |               | processor mode.   |
| UartMpRxAddress        | 2             | When the SCB mode is UART,                                    |
|                        |               | this parameter defines the                                    |
|                        |               | UART address. Only applicable for UART multi-                 |
|                        |               | processor mode.   |
| UartMpRxAddressMask    | 255           | When the SCB mode is UART,                                    |
|                        |               | this parameter defines the                                    |
|                        |               | address mask in multi-  |
|                        |               | processor operation mode.                                     |
|                        |               | Bit value 0 – excludes bit from address comparison.           |
|                        |               | Bit value 1 – the bit needs to                                |
|                        |               | match with the corresponding                                  |
|                        |               | bit of the UART address.                                      |
|                        |               | Only applicable for UART multi-                               |
|                        |               | processor mode.   |



| Parameter Name       | Value      | Description  |
|----------------------|------------|--|
| UartNumberOfDataBits | 8 bits     | When the SCB mode is UART,   |
|                      |            | this parameter defines the number of data bits inside the UART byte/word.  |
| UartNumberOfStopBits | 1 bit      | When the SCB mode is UART,<br>this parameter defines the<br>number of Stop bits.   |
| UartOvsFactor        | 12         | When the SCB mode is UART,<br>this parameter defines the<br>oversampling factor of<br>SCBCLK.  |
| UartParityType       | None       | When the SCB mode is UART, this parameter applies UART parity check as Odd or Even or discards the parity entirely.  |
| UartRtsEnable        | false      | When the SCB mode is UART, this parameter enables the rts output.  Applicable only for devices other than PSoC 4000/PSoC   |
|                      |            | 4100/PSoC 4200.  |
| UartRtsPolarity      | Active Low | When the SCB mode is UART, this parameter specifies active polarity of the output rts signal.  Applicable only for devices other than PSoC 4000/PSoC 4100/PSoC 4200.   |
| UartRtsTriggerLevel  | 4          | When the SCB mode is UART, this parameter specifies the number of entries in the RX FIFO to activate the rts output signal. When the receiver FIFO has fewer entries than the UartRtsTriggerLevel, an rts output signal is activated.  Applicable only for devices other than PSoC 4000/PSoC 4100/PSoC 4200. |
| UartRxBreakWidth     | 11         | This parameter specifies the break width in bits.  |
| UartRxBufferSize     | 8          | When the SCB mode is UART, this parameter defines the size of the RX buffer.   |
| UartRxOutputEnable   | false      | When the SCB mode is UART, this parameter enables the RX trigger output terminal of the component. This terminal must be connected to the DMA input trigger or left unconnected. Only applicable for devices which have a DMA controller.  |



| Parameter Name        | Value    | Description   |
|-----------------------|----------|---|
| UartRxTriggerLevel    | 7        | When the SCB mode is UART, this parameter defines the number of entries in the RX FIFO to trigger control the SCB.INTR_RX.TRIGGER interrupt event or RX DMA trigger output.   |
| UartSmCardRetryOnNack | false    | When the SCB mode is UART, this parameter defines whether to send a message again when a NACK response is received. Only applicable for UART SmartCard mode.  |
| UartSubMode           | Standard | When the SCB mode is UART,<br>this parameter defines the sub<br>mode of UART as: Standard,<br>SmartCard or IrDA.  |
| UartTxBufferSize      | 8        | When the SCB mode is UART, this parameter defines the size of the TX buffer.  |
| UartTxOutputEnable    | false    | When the SCB mode is UART, this parameter enables the TX trigger output terminal of the component. This terminal must be connected to the DMA input trigger or left unconnected. Only applicable for devices which have a DMA controller. |
| UartTxTriggerLevel    | 0        | When the SCB mode is UART, this parameter defines the number of entries in the TX FIFO to control the SCB.INTRTX.TRIGGER interrupt event or TX DMA trigger output.  |
| UartWakeEnable        | false    | When the SCB mode is UART, this parameter enables the wakeup from Deep Sleep on start bit event. The actual wakeup source is RX GPIO. The skip start UART feature allows it to continue receiving bytes.                                  |
| User Comments         |          | Instance-specific comments.   |

9.6 Component type: TCPWM\_P4 [v2.10]

### 9.6.1 Instance Timer\_1

**Description: 16-bit Timer Counter PWM (TCPWM)** 

Instance type: TCPWM\_P4 [v2.10]

Datasheet: online component datasheet for TCPWM\_P4

Table 23. Component Parameters for Timer\_1

| Parameter Name | Value | Description                     |
|----------------|-------|---------------------------------|
| PWMCompare     | 65535 | The initial value for the       |
|                |       | comparison register when in the |
|                |       | PWM mode                        |



| Parameter Name               | Value                | Description EMBEDDED IN TO                                     |
|------------------------------|----------------------|--|
| PWMCompareBuf                | 65535                | The initial value for the second                               |
| 1 WWGGMpareBar               | 00000                | comparison register when in the                                |
|                              |                      | PWM mode   |
| PWMCompareSwap               | Disable swap         | Determines whether the PWM                                     |
|                              | •                    | swap check box is enabled or                                   |
|                              |                      | disabled   |
| PWMCountMode                 | Level                | Determines whether the PWM                                     |
|                              |                      | counter counts at level detection                              |
|                              |                      | or in various modes of edge                                    |
|                              |                      | detection  |
| PWMCountPresent              | false                | Determines if the PWM count                                    |
|                              |                      | signal is present and controls                                 |
| DIAMAD IT' O I               |                      | the visibility of the count pin                                |
| PWMDeadTimeCycle             | 0                    | Sets the number of cycles of dead time insertion               |
| DVA/A Alesta www.mstA Alesta | Tamainal accept made |  |
| PWMInterruptMask             | Terminal count mask  | The mask used for enabling the interrupt bit in the PWM mode   |
| PWMKillEvent                 | Asynchronous         | Selects whether a PWM kill                                     |
| FVVIVIKIIIEVEIII             | Asyliciliollous      | event is synchronous or  |
|                              |                      | asynchronous to the input clock                                |
| PWMLinenSignal               | Direct Output        | Selects whether the PWM  |
| T TTIME ITE I GITAL          | 2oct Gatpat          | line_n signal is inverted or is                                |
|                              |                      | directly output  |
| PWMLineSignal                | Direct Output        | Selects whether the PWM line                                   |
|                              |                      | signal is inverted or is directly                              |
|                              |                      | output   |
| PWMMode                      | PWM                  | Selects one of the three PWM                                   |
|                              |                      | modes - PWM, PWM with dead                                     |
|                              |                      | time insertion, or Pseudo                                      |
| DVA/AAD                      | 05505                | random PWM   |
| PWMPeriod                    | 65535                | The initial value for the period register when in the PWM mode |
| PWMPeriodBuf                 | 65535                | The initial value for the second                               |
| T VVIVIT ETIOGENI            | 03333                | period register when in the                                    |
|                              |                      | PWM mode   |
| PWMPeriodSwap                | Disable swap         | Enables swap between the                                       |
|                              |                      | PWM period and period_buf                                      |
|                              |                      | registers  |
| PWMPrescaler                 | 0                    | Defines the prescaler used to                                  |
|                              |                      | divide the TCPWM clock to                                      |
|                              |                      | create the counter clock                                       |
| PWMReloadMode                | Rising edge          | Determines whether the PWM                                     |
|                              |                      | reload signal is accepted at                                   |
|                              |                      | level detection or in various                                  |
| DW/MD aloadDrassst           | folos                | modes of edge detection  |
| PWMReloadPresent             | false                | Determines whether the PWM reload signal is present and        |
|                              |                      | controls its pin visibility                                    |
| PWMRunMode                   | Continuous           | Selects between continuous                                     |
| - Trivii (driiviodo          | Continuous           | and one shot run mode for the                                  |
|                              |                      | PWM  |
| PWMSetAlign                  | Left align           | Selects the alignment of the                                   |
|                              |                      | PWM waveform to be either left,                                |
|                              |                      | right, center or asymmetrically                                |
|                              |                      | aligned  |
|                              | -                    |  |



| Parameter Name    | Value               | Description  |
|-------------------|---------------------|--|
| PWMStartMode      | Rising edge         | Determines whether the PWM start signal is accepted at level detection or in various modes of edge detection                           |
| PWMStartPresent   | false               | Determines whether the PWM start signal is present and controls its pin visibility   |
| PWMStopEvent      | Don't stop on Kill  | Selects whether to kill the PWM on a stop signal or not  |
| PWMStopMode       | Rising edge         | Determines whether the PWM stop signal is accepted at level detection or in various modes of edge detection                            |
| PWMStopPresent    | false               | Determines whether the PWM stop signal is present and controls its pin visibility  |
| PWMSwitchMode     | Rising edge         | Determines whether the PWM switch signal is accepted at level detection or in various modes of edge detection                          |
| PWMSwitchPresent  | false               | Determines whether the PWM switch signal is present and controls its pin visibility  |
| QuadEncodingModes | x1 Encoding mode    | Selects one of the three<br>quadrature decoder modes –<br>x1, x2, or x4 encoding mode  |
| QuadIndexMode     | Rising edge         | Determines whether the Quadrature Decoder index signal is accepted at level detection or in various modes of edge detection            |
| QuadIndexPresent  | false               | Determines whether the Quadrature Decoder index signal is present and controls its pin visibility                                      |
| QuadInterruptMask | Terminal count mask | The mask used to configure which Quadrature Decoder event causes an interrupt  |
| QuadPhiAMode      | Level               | Determines whether the Quadrature Decoder PhiA signal is accepted at level detection or in various modes of edge detection             |
| QuadPhiBMode      | Level               | Determines whether the<br>Quadrature Decoder PhiB<br>signal is accepted at level<br>detection or in various modes of<br>edge detection |
| QuadStopMode      | Rising edge         | Determines whether the Quadrature Decoder stop signal is accepted at level detection or in various modes of edge detection             |
| QuadStopPresent   | false               | Determines whether the Quadrature Decoder stop signal is present and controls its pin visibility                                       |



| Parameter Name      | Value               | Description Description   |
|---------------------|---------------------|---|
| TCCaptureMode       | Rising edge         | Determines whether the  |
| Todaptureivioue     | Trising edge        | Timer/Counter capture signal is accepted at level detection or in various modes of edge detection                     |
| TCCapturePresent    | false               | Determines whether the Timer/Counter capture signal is present and controls its pin visibility                        |
| TCCompare           | 20000               | The initial value for the comparison register when in the Timer/Counter mode  |
| TCCompareBuf        | 65535               | The initial value for the second comparison register when in the Timer/Counter mode                                   |
| TCCompareSwap       | Disable swap        | Determines whether the<br>Timer/Counter swap check box<br>is enabled or disabled                                      |
| TCCompCapMode       | Capture Mode        | Selects whether the<br>Timer/Counter capture or the<br>compare mode is enabled  |
| TCCountingModes     | Counts up           | Selects the count direction of the counter  |
| TCCountMode         | Level               | Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection |
| TCCountPresent      | false               | Determines whether the<br>Timer/Counter count signal is<br>present and controls its pin<br>visibility                 |
| TCInterruptMask     | Terminal count mask | The mask used to determine which Timer/Counter event causes an interrupt  |
| TCPeriod            | 200                 | The initial value for the Timer/Counter period register   |
| TCPrescaler         | 0                   | Selects the prescaler value to apply to the Timer/Counter clock   |
| TCPWMCapturePresent | false               | Determines whether the Unconfigured capture signal is present and controls its pin visibility                         |
| TCPWMConfig         | Timer Counter       | Selects the TCPWM mode -<br>Unconfigured, Timer/Counter,<br>PWM, or Quadrature Decoder                                |
| TCPWMCountPresent   | false               | Determines whether the Unconfigured count signal is present and controls its pin visibility                           |
| TCPWMReloadPresent  | false               | Determines whether the Unconfigured reload signal is present and controls its pin visibility                          |



| Parameter Name    | Value        | Description   |
|-------------------|--------------|---|
| TCPWMStartPresent | false        | Determines whether the Unconfigured start signal is present and controls its pin visibility                             |
| TCPWMStopPresent  | false        | Determines whether the<br>Unconfigured stop signal is<br>present and controls its pin<br>visibility                     |
| TCReloadMode      | Falling edge | Determines whether the Timer/Counter reload signal is accepted at level detection or in various modes of edge detection |
| TCReloadPresent   | true         | Determines whether the<br>Timer/Counter reload signal is<br>present and controls its pin<br>visibility                  |
| TCRunMode         | One Shot     | Selects whether the counter runs continuously or one shot   |
| TCStartMode       | Falling edge | Determines whether the start signal is accepted at level detection or in various modes of edge detection                |
| TCStartPresent    | true         | Determines whether the<br>Timer/Counter start signal is<br>present and controls its pin<br>visibility                   |
| TCStopMode        | Rising edge  | Determines whether the Timer/Counter stop signal is accepted at level detection or in various modes of edge detection   |
| TCStopPresent     | true         | Determines whether the<br>Timer/Counter stop signal is<br>present and controls its pin<br>visibility                    |
| User Comments     |              | Instance-specific comments.   |

# 9.6.2 Instance Timer\_2

Description: 16-bit Timer Counter PWM (TCPWM)

Instance type: TCPWM\_P4 [v2.10]

Datasheet: online component datasheet for TCPWM\_P4

Table 24. Component Parameters for Timer\_2

| Parameter Name | Value        | Description   |
|----------------|--------------|---|
| PWMCompare     | 65535        | The initial value for the comparison register when in the PWM mode        |
| PWMCompareBuf  | 65535        | The initial value for the second comparison register when in the PWM mode |
| PWMCompareSwap | Disable swap | Determines whether the PWM swap check box is enabled or disabled          |



| Parameter Name   | Value               | Description Description   |
|------------------|---------------------|---|
| PWMCountMode     | Level               | Determines whether the PWM counter counts at level detection or in various modes of edge detection            |
| PWMCountPresent  | false               | Determines if the PWM count signal is present and controls the visibility of the count pin                    |
| PWMDeadTimeCycle | 0                   | Sets the number of cycles of dead time insertion  |
| PWMInterruptMask | Terminal count mask | The mask used for enabling the interrupt bit in the PWM mode  |
| PWMKillEvent     | Asynchronous        | Selects whether a PWM kill event is synchronous or asynchronous to the input clock                            |
| PWMLinenSignal   | Direct Output       | Selects whether the PWM line_n signal is inverted or is directly output                                       |
| PWMLineSignal    | Direct Output       | Selects whether the PWM line signal is inverted or is directly output   |
| PWMMode          | PWM                 | Selects one of the three PWM modes - PWM, PWM with dead time insertion, or Pseudo random PWM                  |
| PWMPeriod        | 65535               | The initial value for the period register when in the PWM mode  |
| PWMPeriodBuf     | 65535               | The initial value for the second period register when in the PWM mode   |
| PWMPeriodSwap    | Disable swap        | Enables swap between the PWM period and period_buf registers  |
| PWMPrescaler     | 0                   | Defines the prescaler used to divide the TCPWM clock to create the counter clock                              |
| PWMReloadMode    | Rising edge         | Determines whether the PWM reload signal is accepted at level detection or in various modes of edge detection |
| PWMReloadPresent | false               | Determines whether the PWM reload signal is present and controls its pin visibility                           |
| PWMRunMode       | Continuous          | Selects between continuous and one shot run mode for the PWM  |
| PWMSetAlign      | Left align          | Selects the alignment of the PWM waveform to be either left, right, center or asymmetrically aligned          |
| PWMStartMode     | Rising edge         | Determines whether the PWM start signal is accepted at level detection or in various modes of edge detection  |
| PWMStartPresent  | false               | Determines whether the PWM start signal is present and controls its pin visibility                            |
| PWMStopEvent     | Don't stop on Kill  | Selects whether to kill the PWM on a stop signal or not   |



| Parameter Name    | Value               | Description   |
|-------------------|---------------------|---|
| PWMStopMode       | Rising edge         | Determines whether the PWM stop signal is accepted at level detection or in various modes of edge detection                 |
| PWMStopPresent    | false               | Determines whether the PWM stop signal is present and controls its pin visibility   |
| PWMSwitchMode     | Rising edge         | Determines whether the PWM switch signal is accepted at level detection or in various modes of edge detection               |
| PWMSwitchPresent  | false               | Determines whether the PWM switch signal is present and controls its pin visibility   |
| QuadEncodingModes | x1 Encoding mode    | Selects one of the three<br>quadrature decoder modes –<br>x1, x2, or x4 encoding mode                                       |
| QuadIndexMode     | Rising edge         | Determines whether the Quadrature Decoder index signal is accepted at level detection or in various modes of edge detection |
| QuadIndexPresent  | false               | Determines whether the Quadrature Decoder index signal is present and controls its pin visibility                           |
| QuadInterruptMask | Terminal count mask | The mask used to configure which Quadrature Decoder event causes an interrupt   |
| QuadPhiAMode      | Level               | Determines whether the Quadrature Decoder PhiA signal is accepted at level detection or in various modes of edge detection  |
| QuadPhiBMode      | Level               | Determines whether the Quadrature Decoder PhiB signal is accepted at level detection or in various modes of edge detection  |
| QuadStopMode      | Rising edge         | Determines whether the Quadrature Decoder stop signal is accepted at level detection or in various modes of edge detection  |
| QuadStopPresent   | false               | Determines whether the Quadrature Decoder stop signal is present and controls its pin visibility                            |
| TCCaptureMode     | Rising edge         | Determines whether the Timer/Counter capture signal is accepted at level detection or in various modes of edge detection    |
| TCCapturePresent  | false               | Determines whether the Timer/Counter capture signal is present and controls its pin visibility                              |



| TCCompare  20000 The initial value for the comparison register when in the Timer/Counter mode  TCCompareBuf  TCCompareSwap  Disable swap  TCCompareSwap  Disable swap  TCCompCapMode  Capture Mode  TCCountingModes  TCCountingModes  TCCountingModes  TCCountMode  TCCOu | Parameter Name           | Value               | Description EMBEDDED IN TO    |
|--|--------------------------|---------------------|-------------------------------|
| TCCompareBuf  TCCompareBuf  TCCompareBuf  TCCompareSwap  Disable swap  Disable swap  Determines whether the Timer/Counter mode  TCCompCapMode  Capture Mode  TCCompCapMode  Capture Mode  TCCountingModes  Counts up  TCCountMode  Level  TCCountMode  TCCountMode  TCCountMode  TCCOuntPresent  TCCountPresent  TCPWMCapturePresent  Talse  Determines whether the Timer/Counter capture or the compare mode is enabled of the counter of |                          |                     |                               |
| Timer/Counter mode TCCompareBuf TCCompareBuf TCCompareSwap Disable swap Disable swap Disable swap Disable swap TCCompCapMode TCCompCapMode TCCompCapMode TCCountingModes TCCou | 1 000mpare               | 20000               |                               |
| TCCompareBuf  TCCompareSwap  Disable swap  TCCompareSwap  Disable swap  TCCompCapMode  Capture Mode  TCCompCapMode  Capture Mode  TCCountingModes  Counts up  Counts up  Determines whether the Timer/Counter capture of the counter of |                          |                     |                               |
| TCCompareSwap  Disable swap  TCCompareSwap  Disable swap  Determines whether the Timer/Counter swap check box is enabled or disabled  TCCompCapMode  Capture Mode  TCCountingModes  Counts up  Determines whether the Timer/Counter capture or the compare mode is enabled  TCCountingModes  Counts up  Determines whether the Timer/Counter count direction of the counter  TCCountMode  Level  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter event causes an interrupt  TCPrescaler  O Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  False  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  False  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  False  Determines whether the Unconfigured tala signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility                     | TCCompareBuf             | 65535               |                               |
| TCCompareSwap  Disable swap  Determines whether the Timer/Counter swap check box is enabled or disabled  TCCompCapMode  Capture Mode  Selects whether the Timer/Counter capture or the compare mode is enabled  TCCountingModes  Counts up  Selects the count direction of the counter  TCCountMode  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  0  Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  Determines whether the Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured search signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  | TOCOMPAREDUI             | 05555               |                               |
| TCCompareSwap  Disable swap  Disable swap  Determines whether the Timer/Counter swap check box is enabled or disabled and disabled and disabled seems of the compare mode is enabled or disabled. Selects whether the Timer/Counter capture or the compare mode is enabled and disabled. TCCountingModes  Counts up  Determines whether the Timer/Counter count direction of the counter. TCCountMode  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection. TCCountPresent  False  TCCountPresent  False  Determines whether the Timer/Counter count signal is present and controls its pin visibility. TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter period register. TCPeriod  TCPeriod  Determines whether the Timer/Counter period register. TCPerscaler  O Selects the prescaler value to apply to the Timer/Counter colock  TCPWMCapturePresent  False  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  False  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  False  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  False  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured treload signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility   |                          |                     |                               |
| Timer/Counter swap check box is enabled or disabled TCCompCapMode  Capture Mode  Selects whether the Timer/Counter capture or the compare mode is enabled TCCountingModes  Counts up Selects the count direction of the counter TCCountMode  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  False  Terminal count mask  Terminal count mask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt TCPeriod  200  The initial value for the Timer/Counter event print in the print of the counter count signal is present and controls its pin visibility  TCPWMCapturePresent  TCPWMCapturePresent  False  TCPWMConfig  Timer Counter  TCPWMCountPresent  False  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMReloadPresent  False  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  False  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  False  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility   | TCComporeSwan            | Disable awar        |                               |
| TCCompCapMode  Capture Mode  Selects whether the Selects whether the Timer/Counter capture or the compare mode is enabled  TCCountingModes  Counts up  Selects the count direction of the counter  TCCountMode  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  0  Selects the prescaler value to apply to the Timer/Counter solock  TCPWMCapturePresent  false  Determines whether the Unconfigured apture signal is present and controls its pin visibility  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured tool signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility   | TCCompareSwap            | Disable Swap        |                               |
| TCCompCapMode Capture Mode Selects whether the Timer/Counter capture or the compare mode is enabled counting Modes Counts up Selects the count direction of the counter TCCountMode Level Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent False Terminal count mask Terminal count mask The mask used to determine which Timer/Counter event causes an interrupt causes an interrupt the initial value for the Timer/Counter period register TCPeriod Timer Counter Talse Timer Counter TCPeriod TCPeriod Timer Counter TCPeriod TCPeriod Timer Counter Timer Counter Timer/Counter Timer/Counter TCPeriod Timer Counter Timer Counter Timer/Counter TCPeriod Timer Counter Timer/Counter TCPeriod Timer Counter Timer/Counter Timer/Counter TCPeriod Timer Counter Timer/Counter Timer/ |                          |                     |                               |
| Timer/Counter capture or the compare mode is enabled Selects the count direction of the counter  TCCountMode  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  0  Selects the prescaler value to apply to the Timer/Counter counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured transpal signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility   | TOO O M I -              | Cantura Mada        |                               |
| TCCountingModes  Counts up  Selects the count direction of the counter  TCCountMode  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility   | 1 CCompCapiviode         | Capture Mode        |                               |
| TCCountingModes  Counts up  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPescaler  O Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured eload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent  False  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStartPresent false  Determines whether the Unconfigured start signal is present and controls its pin visibility  |                          |                     |                               |
| TCCountMode  Level  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  TCCOUNTPRESE |                          |                     |                               |
| TCCountMode  Level  Determines whether the Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  0  Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility  | TCCountingModes          | Counts up           |                               |
| Timer/Counter count signal is accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  0  Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  |                          |                     |                               |
| accepted at a level detect or at various modes of edge detection  TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  0  Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility   | TCCountMode              | Level               |                               |
| TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  TCPeriod  TCPeriod  TCPeriod  TCPeriod  TCPeriod  TCPeriod  Topic in initial value for the Timer/Counter period register  TCPrescaler  TCPrescaler  TCPWMCapturePresent  TCPWMCapturePresent  TCPWMConfig  Timer Counter  TCPWMConfig  Timer Counter  TCPWMCountPresent  TCPWMCountPresent  TCPWMCountPresent  TCPWMCountPresent  TCPWMCountPresent  TCPWMCountPresent  TCPWMCountPresent  TCPWMCountPresent  TCPWMReloadPresent  TCPWMReloadPresent  TCPWMStartPresent  |                          |                     |                               |
| TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  Oselects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  Determines whether the Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured rount signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  |                          |                     | ·                             |
| TCCountPresent  false  Determines whether the Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  200  The initial value for the Timer/Counter period register  TCPrescaler  0  Selects the prescaler value to apply to the Timer/Counter olock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  |                          |                     | _                             |
| Timer/Counter count signal is present and controls its pin visibility  TCInterruptMask  Terminal count mask  The mask used to determine which Timer/Counter event causes an interrupt  TCPeriod  TCPeriod  TCPrescaler  O  Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  TCPWMCountPresent  false  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured counts ginal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  |                          |                     |                               |
| TCInterruptMask Terminal count mask The mask used to determine which Timer/Counter event causes an interrupt TCPeriod TCPeriod The initial value for the Timer/Counter period register TCPrescaler TCPrescaler TCPWMCapturePresent Talse TCPWMConfig Timer Counter TCPWMCountPresent TCPWMReloadPresent TCPWMReloadPresent TCPWMStartPresent TCPWMStartPresent TCPWMStartPresent TCPWMStartPresent TCPWMStartPresent TCPWMStopPresent TCPWMStopPrese | TCCountPresent           | false               | Determines whether the        |
| TCInterruptMask Terminal count mask The mask used to determine which Timer/Counter event causes an interrupt TCPeriod TCPeriod TCPrescaler TCPrescaler TCPrescaler TCPWMCapturePresent TCPWMConfig Timer Counter TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMReloadPresent TCPWMReloadPresent TCPWMStartPresent TCPWMStartPresent TCPWMStartPresent TCPWMStartPresent TCPWMStopPresent TCPWMSt |                          |                     | Timer/Counter count signal is |
| TCInterruptMask Terminal count mask The mask used to determine which Timer/Counter event causes an interrupt TCPeriod TCPeriod TDenitial value for the Timer/Counter period register TCPrescaler TCPrescaler TCPWMCapturePresent TCPWMCapturePresent TCPWMConfig Timer Counter TCPWMConfig Timer Counter TCPWMConfig Timer Counter TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMCountPresent TCPWMReloadPresent TCPWMReloadPresent TCPWMStartPresent TCPWMStartPresent TCPWMStartPresent TCPWMStopPresent TCPWMStop |                          |                     | present and controls its pin  |
| TCPeriod 200 The initial value for the Timer/Counter period register  TCPrescaler 0 Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent false Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig Timer Counter Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent false Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent false Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent false Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility   |                          |                     | visibility                    |
| TCPeriod 200 The initial value for the Timer/Counter period register  TCPrescaler 0 Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent false Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig Timer Counter Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent false Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent false Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMStartPresent false Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility   | TCInterruptMask          | Terminal count mask | The mask used to determine    |
| TCPeriod  200 The initial value for the Timer/Counter period register  TCPrescaler  0 Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig  Timer Counter Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false Determines whether the Unconfigured start signal is present and controls its pin visibility   | '                        |                     | which Timer/Counter event     |
| TCPeriod 200 The initial value for the Timer/Counter period register  TCPrescaler 0 Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent false Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig Timer Counter Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent false Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent false Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility  |                          |                     | causes an interrupt           |
| Timer/Counter period register  TCPrescaler  O Selects the prescaler value to apply to the Timer/Counter clock  TCPWMCapturePresent  false Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig Timer Counter Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent false Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent false Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured start signal is present and controls its pin visibility   | TCPeriod                 | 200                 | -                             |
| TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMCountPresent  false  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility   |                          |                     |                               |
| apply to the Timer/Counter clock  TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility  | TCPrescaler              | 0                   |                               |
| TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility  | Tot research             |                     |                               |
| TCPWMCapturePresent  false  Determines whether the Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility  |                          |                     |                               |
| Unconfigured capture signal is present and controls its pin visibility  TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility   | TCPWMCanturePresent      | false               |                               |
| TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility   | 101 WWOaptarer resent    | laise               |                               |
| TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility   |                          |                     |                               |
| TCPWMConfig  Timer Counter  Selects the TCPWM mode - Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin  |                          |                     |                               |
| Unconfigured, Timer/Counter, PWM, or Quadrature Decoder  TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin   | TCDW/MConfig             | Timer Counter       |                               |
| TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin visibility   | TOP WINCOINING           | Timer Counter       |                               |
| TCPWMCountPresent  false  Determines whether the Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin  |                          |                     |                               |
| Unconfigured count signal is present and controls its pin visibility  TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin   | TCDW/MCountDropont       | foloo               | -                             |
| TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin   | TCP VVIVICOUTIEFTESETIL  | laise               |                               |
| TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin   |                          |                     |                               |
| TCPWMReloadPresent  false  Determines whether the Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin   |                          |                     |                               |
| Unconfigured reload signal is present and controls its pin visibility  TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin unconfigured stop signal is present and controls its pin  | TODWAD all a significant | f-1                 | ,                             |
| TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin  | I CPVVIVIKeloadPresent   | raise               |                               |
| TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin  |                          |                     |                               |
| TCPWMStartPresent  false  Determines whether the Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent  false  Determines whether the Unconfigured stop signal is present and controls its pin  |                          |                     | ·                             |
| Unconfigured start signal is present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured stop signal is present and controls its pin   | TODIAMAGE                |                     | ,                             |
| TCPWMStopPresent false present and controls its pin visibility  TCPWMStopPresent false Determines whether the Unconfigured stop signal is present and controls its pin   | I CPVVIVIStartPresent    | false               |                               |
| TCPWMStopPresent false Determines whether the Unconfigured stop signal is present and controls its pin   |                          |                     |                               |
| TCPWMStopPresent false Determines whether the Unconfigured stop signal is present and controls its pin   |                          |                     |                               |
| Unconfigured stop signal is present and controls its pin   |                          |                     |                               |
| present and controls its pin   | I CPWMStopPresent        | false               |                               |
|  |                          |                     |                               |
| visibility   |                          |                     |                               |
|  |                          |                     | visibility                    |



| Parameter Name  | Value        | Description                             |
|-----------------|--------------|---|
| TCReloadMode    | Falling edge | Determines whether the                  |
|                 |              | Timer/Counter reload signal is          |
|                 |              | accepted at level detection or in       |
|                 |              | various modes of edge detection         |
| TCReloadPresent | <b>*</b>     | Determines whether the                  |
| TCReloadPresent | true         | Timer/Counter reload signal is          |
|                 |              | present and controls its pin            |
|                 |              | visibility                              |
| TCRunMode       | One Shot     | Selects whether the counter             |
|                 |              | runs continuously or one shot           |
| TCStartMode     | Falling edge | Determines whether the start            |
|                 |              | signal is accepted at level             |
|                 |              | detection or in various modes of        |
|                 |              | edge detection                          |
| TCStartPresent  | true         | Determines whether the                  |
|                 |              | Timer/Counter start signal is           |
|                 |              | present and controls its pin visibility |
| TCStopMode      | Rising edge  | Determines whether the                  |
| ТСЭторинове     | Trising eage | Timer/Counter stop signal is            |
|                 |              | accepted at level detection or in       |
|                 |              | various modes of edge                   |
|                 |              | detection                               |
| TCStopPresent   | true         | Determines whether the                  |
|                 |              | Timer/Counter stop signal is            |
|                 |              | present and controls its pin            |
|                 |              | visibility                              |
| User Comments   |              | Instance-specific comments.             |

# 9.7 Component type: Timer [v2.80]

## 9.7.1 Instance DATALINK\_Timer

Description: 8, 16, 24 or 32-bit Timer

Instance type: Timer [v2.80]

Datasheet: online component datasheet for Timer

Table 25. Component Parameters for DATALINK\_Timer

| Parameter Name         | Value | Description  |
|------------------------|-------|--|
| CaptureAlternatingFall | false | Enables data capture on either edge but not until a valid falling edge is detected first.  |
| CaptureAlternatingRise | false | Enables data capture on either edge but not until a valid rising edge is detected first.   |
| CaptureCount           | 2     | The CaptureCount parameter works as a divider on the hardware input "capture". A CaptureCount value of 2 would result in an actual capture taking place every other time the input "capture" is changed. |
| CaptureCounterEnabled  | false | Enables the capture counter to count capture events (up to 127) before a capture is triggered.   |



| Parameter Name      | Value         | Description   |
|---------------------|---------------|---|
| CaptureMode         | Falling Edge  | This parameter defines the capture input signal requirements to trigger a valid capture event   |
| EnableMode          | Hardware Only | This parameter specifies the methods in enabling the component. Hardware mode makes the enable input pin visible. Software mode may reduce the resource usage if not enabled.     |
| FixedFunction       | false         | Configures the component to use fixed function HW block instead of the UDB implementation.  |
| InterruptOnCapture  | false         | Parameter to check whether interrupt on a capture event is enabled or disabled.   |
| InterruptOnFIFOFull | true          | Parameter to check whether interrupt on a FIFO Full event is enabled disabled.  |
| InterruptOnTC       | true          | Parameter to check whether interrupt on a TC is enabled or disabled.  |
| NumberOfCaptures    | 4             | Number of captures allowed until the counter is cleared or disabled.  |
| Period              | 79            | Defines the timer period (This is also the reload value when terminal count is reached)   |
| Resolution          | 8             | Defines the resolution of the hardware. This parameter affects how many bits are used in the Period counter and defines the maximum resolution of the internal component signals. |
| RunMode             | One Shot      | Defines the hardware to run continuously, run until a terminal count is reached or run until an interrupt event is triggered.   |
| TriggerMode         | Rising Edge   | Defines the required trigger input signal to cause a valid trigger enable of the timer  |
| User Comments       |               | Instance-specific comments.   |



### 10 Other Resources

The following documents contain important information on Cypress software APIs that might be relevant to this design:

- Standard Types and Defines chapter in the System Reference Guide
  - Software base types
  - Hardware register types
  - Compiler defines
  - Cypress API return codes
  - Interrupt types and macros
- Registers
  - o The full PSoC 4 register map is covered in the PSoC 4 Registers Technical Reference
  - o Register Access chapter in the System Reference Guide
    - § CY\_GET API routines§ CY\_SET API routines
- System Functions chapter in the **System Reference Guide** 
  - o General API routines
  - o CyDelay API routines
  - o CyVd Voltage Detect API routines
- Power Management
  - o Power Supply and Monitoring chapter in the PSoC 4 Technical Reference Manual
  - o Low Power Modes chapter in the PSoC 4 Technical Reference Manual
  - o Power Management chapter in the System Reference Guide
    - § CyPm API routines
- Watchdog Timer chapter in the System Reference Guide
  - CyWdt API routines