PSoC® Creator™
Project Datasheet for Project-DAQ_for_RaspberryPi

Creation Time: 05/01/2024 18:24:50
User: DESKTOP-9SIMMM9\Gemma
Project: Project-DAQ_for_RaspberryPi
Tool: PSoC Creator 4.2

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 © 2024 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

The Cypress PSoC 5 is a family of 32-bit devices with the following characteristics:

- High-performance 32-bit ARM Cortex-M3 core with a nested vectored interrupt controller (NVIC) and a high-performance DMA controller
- Digital system that includes configurable Universal Digital Blocks (UDBs) and specific function peripherals, such as USB, I2C and SPI
- Analog subsystem that includes 20-bit Delta Sigma converters (ADC), SAR ADCs, 8-bit DACs that can be configured for 12-bit operation, comparators, op amps and configurable switched capacitor (SC) and continuous time (CT) blocks to create PGAs, TIAs, mixers, and more
- Several types of memory elements, including SRAM, flash, and EEPROM
- Programming and debug system through JTAG, serial wire debug (SWD), and single wire viewer (SWV)
- Flexible routing to all pins

Figure 1 shows the major components of a typical CY8C58LP series member PSoC 5LP device. For details on all the systems listed above, please refer to the PSoC 5LP Technical Reference Manual.

Figure 1. CY8C58LP Device Series Block Diagram

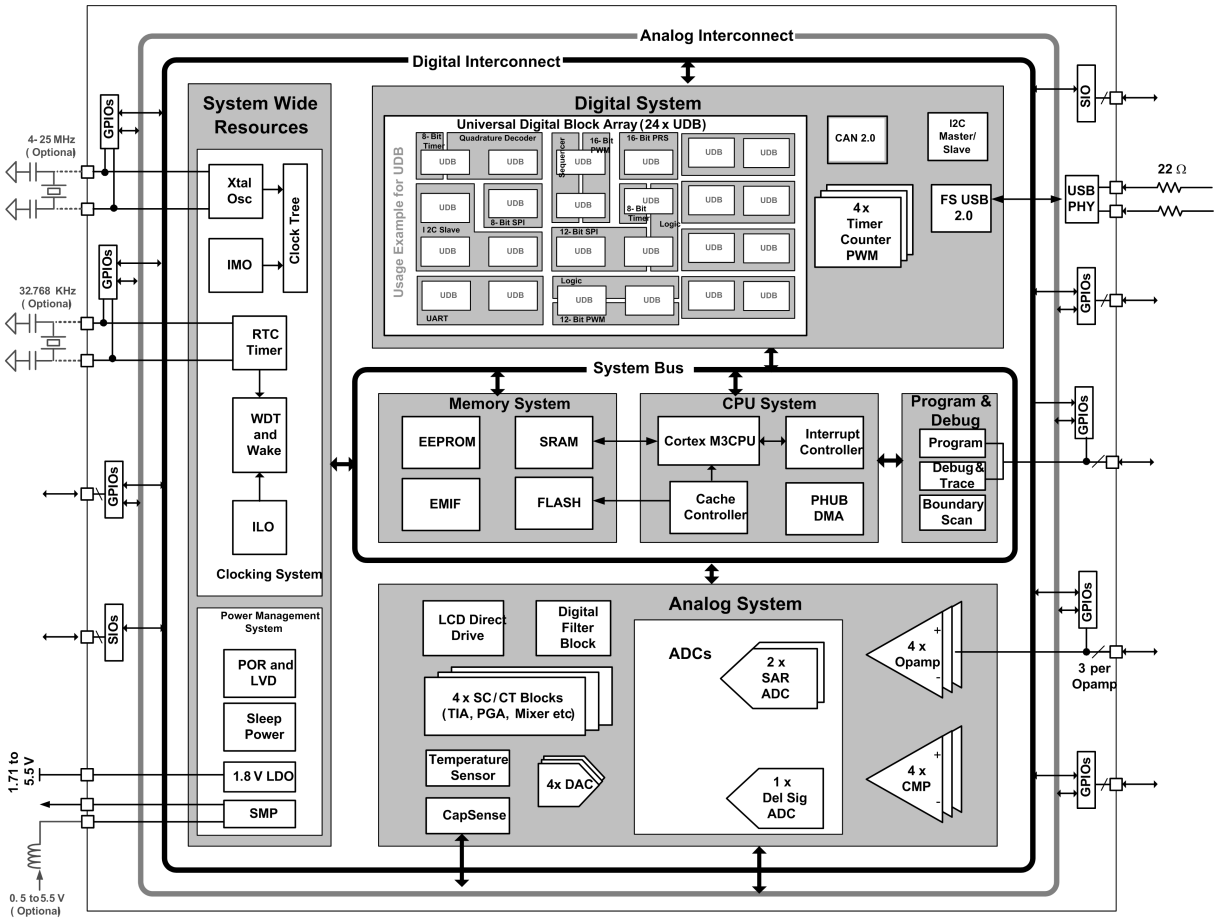


Table 1 lists the key characteristics of this device.

Table 1. Device Characteristics

Name	Value
Part Number	CY8C5888LTI-LP097
Package Name	68-QFN
Family	PSoC 5LP
Series	CY8C58LP
Max CPU speed (MHz)	0
Flash size (kB)	256
SRAM size (kB)	64
EEPROM size (bytes)	2048
Vdd range (V)	1.71 to 5.5
Automotive qualified	No (Industrial Grade Only)
Temp range (Celsius)	-40 to 85
JTAG ID	0x2E161069

NOTE: The CPU speed noted above is the maximum available speed. The CPU is clocked by Bus Clock, listed in the System Clocks section below.

Table 2 lists the device resources that this design uses:

Table 2. Device Resources

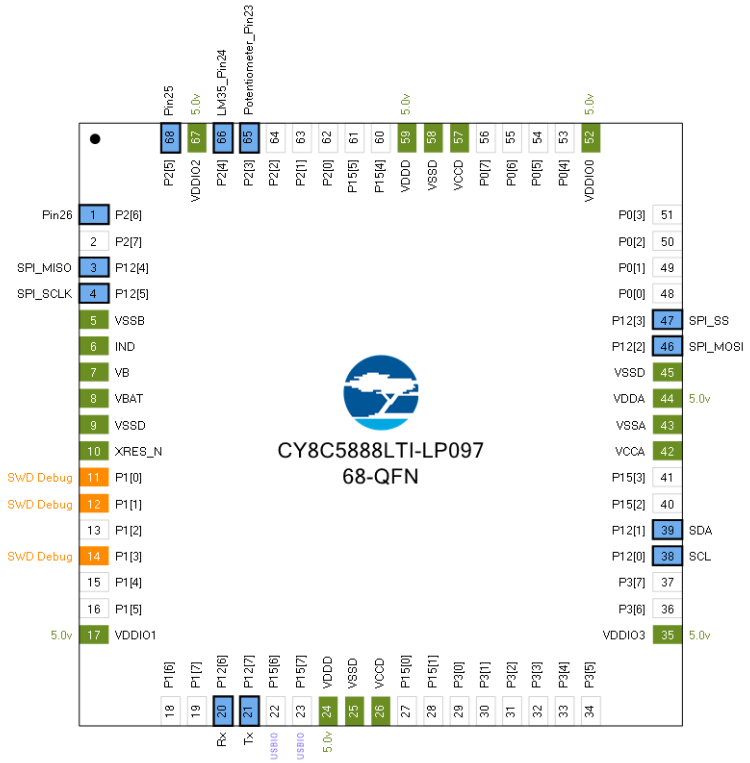
Resource Type	Used	Free	Max	% Used
Digital Clocks	4	4	8	50.00 %
Analog Clocks	1	3	4	25.00 %
CapSense Buffers	0	2	2	0.00 %
Digital Filter Block	0	1	1	0.00 %
Interrupts	7	25	32	21.88 %
IO	15	33	48	31.25 %
Segment LCD	0	1	1	0.00 %
CAN 2.0b	0	1	1	0.00 %
I2C	0	1	1	0.00 %
USB	0	1	1	0.00 %
DMA Channels	0	24	24	0.00 %
Timer	0	4	4	0.00 %
UDB				
Macrocells	62	130	192	32.29 %
Unique P-terms	119	265	384	30.99 %
Total P-terms	133			
Datapath Cells	5	19	24	20.83 %
Status Cells	9	15	24	37.50 %
StatusI Registers	5			
Sync Cells (x4)	1			
Routed Count7 Load/Enable	3			
Control Cells	4	20	24	16.67 %
Control Registers	1			
Count7 Cells	3			
Opamp	0	4	4	0.00 %
Comparator	0	4	4	0.00 %
Delta-Sigma ADC	1	0	1	100.00 %
LPF	0	2	2	0.00 %
SAR ADC	0	2	2	0.00 %
Analog (SC/CT) Blocks	0	4	4	0.00 %

Resource Type	Used	Free	Max	% Used
DAC				
VIDAC	0	4	4	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	Type	Drive Mode	Reset State
1	P2[6]	Pin26	Analog	HiZ analog	HiZ Analog Unb
2	P2[7]	GPIO [unused]			HiZ Analog Unb
3	P12[4]	SPI_MISO	Dgtl Out	Strong drive	HiZ Analog Unb
4	P12[5]	SPI_SCLK	Dgtl In	HiZ digital	HiZ Analog Unb
5	VSSB	VSSB	Dedicated		
6	IND	IND	Dedicated		
7	VB	VB	Dedicated		
8	VBAT	VBAT	Dedicated		
9	VSSD	VSSD	Power		
10	XRES_N	XRES_N	Dedicated		
11	P1[0]	Debug:SWD_IO	Reserved		
12	P1[1]	Debug:SWD_CK	Reserved		
13	P1[2]	GPIO [unused]			HiZ Analog Unb
14	P1[3]	Debug:SWV	Reserved		
15	P1[4]	GPIO [unused]			HiZ Analog Unb
16	P1[5]	GPIO [unused]			HiZ Analog Unb
17	VDDIO1	VDDIO1	Power		
18	P1[6]	GPIO [unused]			HiZ Analog Unb
19	P1[7]	GPIO [unused]			HiZ Analog Unb
20	P12[6]	Rx	Dgtl In	HiZ digital	HiZ Analog Unb
21	P12[7]	Tx	Dgtl Out	Strong drive	HiZ Analog Unb
22	P15[6]	USB IO [unused]			HiZ Analog Unb
23	P15[7]	USB IO [unused]			HiZ Analog Unb
24	VDDD	VDDD	Power		
25	VSSD	VSSD	Power		
26	VCCD	VCCD	Power		
27	P15[0]	GPIO [unused]			HiZ Analog Unb
28	P15[1]	GPIO [unused]			HiZ Analog Unb
29	P3[0]	GPIO [unused]			HiZ Analog Unb
30	P3[1]	GPIO [unused]			HiZ Analog Unb
31	P3[2]	GPIO [unused]			HiZ Analog Unb
32	P3[3]	GPIO [unused]			HiZ Analog Unb
33	P3[4]	GPIO [unused]			HiZ Analog Unb
34	P3[5]	GPIO [unused]			HiZ Analog Unb
35	VDDIO3	VDDIO3	Power		
36	P3[6]	GPIO [unused]			HiZ Analog Unb
37	P3[7]	GPIO [unused]			HiZ Analog Unb
38	P12[0]	SCL	Dgtl I/O	Res pull up	HiZ Analog Unb
39	P12[1]	SDA	Dgtl I/O	Res pull up	HiZ Analog Unb
40	P15[2]	GPIO [unused]			HiZ Analog Unb
41	P15[3]	GPIO [unused]			HiZ Analog Unb
42	VCCA	VCCA	Power		
43	VSSA	VSSA	Power		
44	VDDA	VDDA	Power		
45	VSSD	VSSD	Power		

Pin	Port	Name	Type	Drive Mode	Reset State
46	P12[2]	SPI_MOSI	Dgtl In	HiZ digital	HiZ Analog Unb
47	P12[3]	SPI_SS	Dgtl In	HiZ digital	HiZ Analog Unb
48	P0[0]	GPIO [unused]			HiZ Analog Unb
49	P0[1]	GPIO [unused]			HiZ Analog Unb
50	P0[2]	GPIO [unused]			HiZ Analog Unb
51	P0[3]	GPIO [unused]			HiZ Analog Unb
52	VDDIO0	VDDIO0	Power		
53	P0[4]	GPIO [unused]			HiZ Analog Unb
54	P0[5]	GPIO [unused]			HiZ Analog Unb
55	P0[6]	GPIO [unused]			HiZ Analog Unb
56	P0[7]	GPIO [unused]			HiZ Analog Unb
57	VCCD	VCCD	Power		
58	VSSD	VSSD	Power		
59	VDDD	VDDD	Power		
60	P15[4]	GPIO [unused]			HiZ Analog Unb
61	P15[5]	GPIO [unused]			HiZ Analog Unb
62	P2[0]	GPIO [unused]			HiZ Analog Unb
63	P2[1]	GPIO [unused]			HiZ Analog Unb
64	P2[2]	GPIO [unused]			HiZ Analog Unb
65	P2[3]	Potentiometer_Pin23	Analog	HiZ analog	HiZ Analog Unb
66	P2[4]	LM35_Pin24	Analog	HiZ analog	HiZ Analog Unb
67	VDDIO2	VDDIO2	Power		
68	P2[5]	Pin25	Analog	HiZ analog	HiZ Analog Unb

Abbreviations used in Table 3 have the following meanings:

- HiZ analog = High impedance analog
- HiZ Analog Unb = Hi-Z Analog Unbuffered
- Dgtl Out = Digital Output
- Dgtl In = Digital Input
- HiZ digital = High impedance digital
- Dgtl I/O = Digital In/Out
- 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	Type	Drive Mode	Reset State
P0[0]	48	GPIO [unused]			HiZ Analog Unb
P0[1]	49	GPIO [unused]			HiZ Analog Unb
P0[2]	50	GPIO [unused]			HiZ Analog Unb
P0[3]	51	GPIO [unused]			HiZ Analog Unb
P0[4]	53	GPIO [unused]			HiZ Analog Unb
P0[5]	54	GPIO [unused]			HiZ Analog Unb
P0[6]	55	GPIO [unused]			HiZ Analog Unb
P0[7]	56	GPIO [unused]			HiZ Analog Unb
P1[0]	11	Debug:SWD_IO	Reserved		
P1[1]	12	Debug:SWD_CK	Reserved		
P1[2]	13	GPIO [unused]			HiZ Analog Unb
P1[3]	14	Debug:SWV	Reserved		
P1[4]	15	GPIO [unused]			HiZ Analog Unb
P1[5]	16	GPIO [unused]			HiZ Analog Unb
P1[6]	18	GPIO [unused]			HiZ Analog Unb
P1[7]	19	GPIO [unused]			HiZ Analog Unb
P12[0]	38	SCL	Dgtl I/O	Res pull up	HiZ Analog Unb
P12[1]	39	SDA	Dgtl I/O	Res pull up	HiZ Analog Unb
P12[2]	46	SPI_MOSI	Dgtl In	HiZ digital	HiZ Analog Unb
P12[3]	47	SPI_SS	Dgtl In	HiZ digital	HiZ Analog Unb
P12[4]	3	SPI_MISO	Dgtl Out	Strong drive	HiZ Analog Unb
P12[5]	4	SPI_SCLK	Dgtl In	HiZ digital	HiZ Analog Unb
P12[6]	20	Rx	Dgtl In	HiZ digital	HiZ Analog Unb
P12[7]	21	Tx	Dgtl Out	Strong drive	HiZ Analog Unb
P15[0]	27	GPIO [unused]			HiZ Analog Unb
P15[1]	28	GPIO [unused]			HiZ Analog Unb
P15[2]	40	GPIO [unused]			HiZ Analog Unb
P15[3]	41	GPIO [unused]			HiZ Analog Unb
P15[4]	60	GPIO [unused]			HiZ Analog Unb
P15[5]	61	GPIO [unused]			HiZ Analog Unb
P15[6]	22	USB IO [unused]			HiZ Analog Unb
P15[7]	23	USB IO [unused]			HiZ Analog Unb
P2[0]	62	GPIO [unused]			HiZ Analog Unb
P2[1]	63	GPIO [unused]			HiZ Analog Unb
P2[2]	64	GPIO [unused]			HiZ Analog Unb
P2[3]	65	Potentiometer_Pin23	Analog	HiZ analog	HiZ Analog Unb
P2[4]	66	LM35_Pin24	Analog	HiZ analog	HiZ Analog Unb
P2[5]	68	Pin25	Analog	HiZ analog	HiZ Analog Unb
P2[6]	1	Pin26	Analog	HiZ analog	HiZ Analog Unb
P2[7]	2	GPIO [unused]			HiZ Analog Unb
P3[0]	29	GPIO [unused]			HiZ Analog Unb
P3[1]	30	GPIO [unused]			HiZ Analog Unb
P3[2]	31	GPIO [unused]			HiZ Analog Unb
P3[3]	32	GPIO [unused]			HiZ Analog Unb
P3[4]	33	GPIO [unused]			HiZ Analog Unb

Port	Pin	Name	Type	Drive Mode	Reset State
P3[5]	34	GPIO [unused]			HiZ Analog Unb
P3[6]	36	GPIO [unused]			HiZ Analog Unb
P3[7]	37	GPIO [unused]			HiZ Analog Unb

Abbreviations used in Table 4 have the following meanings:

- HiZ Analog Unb = Hi-Z Analog Unbuffered
- Dgtl I/O = Digital In/Out
- Res pull up = Resistive pull up
- Dgtl In = Digital Input
- HiZ digital = High impedance digital
- Dgtl Out = Digital Output
- HiZ analog = High impedance analog

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	Type	Reset State
Debug:SWD_CK	P1[1]	Reserved	
Debug:SWD_IO	P1[0]	Reserved	
Debug:SWV	P1[3]	Reserved	
GPIO [unused]	P0[4]		HiZ Analog Unb
GPIO [unused]	P3[1]		HiZ Analog Unb
GPIO [unused]	P3[3]		HiZ Analog Unb
GPIO [unused]	P3[2]		HiZ Analog Unb
GPIO [unused]	P15[0]		HiZ Analog Unb
GPIO [unused]	P15[1]		HiZ Analog Unb
GPIO [unused]	P3[0]		HiZ Analog Unb
GPIO [unused]	P0[3]		HiZ Analog Unb
GPIO [unused]	P3[7]		HiZ Analog Unb
GPIO [unused]	P3[6]		HiZ Analog Unb
GPIO [unused]	P15[3]		HiZ Analog Unb
GPIO [unused]	P0[0]		HiZ Analog Unb
GPIO [unused]	P0[2]		HiZ Analog Unb
GPIO [unused]	P3[4]		HiZ Analog Unb
GPIO [unused]	P15[2]		HiZ Analog Unb
GPIO [unused]	P0[1]		HiZ Analog Unb
GPIO [unused]	P1[2]		HiZ Analog Unb
GPIO [unused]	P1[4]		HiZ Analog Unb
GPIO [unused]	P1[5]		HiZ Analog Unb
GPIO [unused]	P2[1]		HiZ Analog Unb
GPIO [unused]	P3[5]		HiZ Analog Unb
GPIO [unused]	P2[7]		HiZ Analog Unb
GPIO [unused]	P2[2]		HiZ Analog Unb
GPIO [unused]	P2[0]		HiZ Analog Unb
GPIO [unused]	P0[7]		HiZ Analog Unb
GPIO [unused]	P0[6]		HiZ Analog Unb
GPIO [unused]	P0[5]		HiZ Analog Unb
GPIO [unused]	P15[4]		HiZ Analog Unb
GPIO [unused]	P1[7]		HiZ Analog Unb
GPIO [unused]	P1[6]		HiZ Analog Unb
GPIO [unused]	P15[5]		HiZ Analog Unb
LM35_Pin24	P2[4]	Analog	HiZ Analog Unb
Pin25	P2[5]	Analog	HiZ Analog Unb
Pin26	P2[6]	Analog	HiZ Analog Unb
Potentiometer_Pin23	P2[3]	Analog	HiZ Analog Unb
Rx	P12[6]	Dgtl In	HiZ Analog Unb
SCL	P12[0]	Dgtl I/O	HiZ Analog Unb
SDA	P12[1]	Dgtl I/O	HiZ Analog Unb
SPI_MISO	P12[4]	Dgtl Out	HiZ Analog Unb
SPI_MOSI	P12[2]	Dgtl In	HiZ Analog Unb
SPI_SCLK	P12[5]	Dgtl In	HiZ Analog Unb
SPI_SS	P12[3]	Dgtl In	HiZ Analog Unb

Name	Port	Type	Reset State
Tx	P12[7]	Dgtl Out	HiZ Analog Unb
USB IO [unused]	P15[6]		HiZ Analog Unb
USB IO [unused]	P15[7]		HiZ Analog Unb

Abbreviations used in Table 5 have the following meanings:

- HiZ Analog Unb = Hi-Z Analog Unbuffered
- Dgtl In = Digital Input
- Dgtl I/O = Digital In/Out
- Dgtl Out = Digital Output

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

- Pins chapter in the System Reference Guide
  - 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
Enable Error Correcting Code (ECC)	False
Store Configuration Data in ECC Memory	True
Instruction Cache Enabled	True
Enable Fast IMO During Startup	True
Unused Bonded IO	Allow but warn
Heap Size (bytes)	0x0800
Stack Size (bytes)	0x0800
Include CMSIS Core Peripheral Library Files	True

3.2 System Debug Settings

Table 7. System Debug Settings

Name	Value
Debug Select	SWD+SWV (serial wire debug and viewer)
Enable Device Protection	False
Embedded Trace (ETM)	False
Use Optional XRES	False

3.3 System Operating Conditions

Table 8. System Operating Conditions

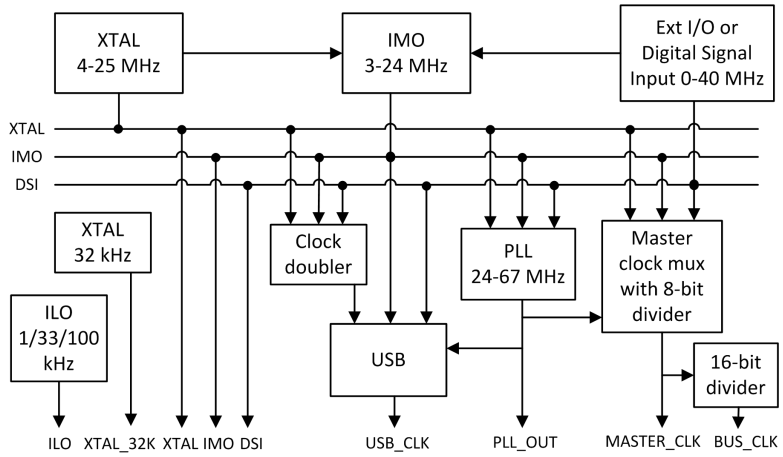
Name	Value
VDDA (V)	5.0
VDDD (V)	5.0
VDDIO0 (V)	5.0
VDDIO1 (V)	5.0
VDDIO2 (V)	5.0
VDDIO3 (V)	5.0
Variable VDDA	False
Temperature Range	-40C - 85/125C

4 Clocks

The clock system includes these clock resources:

- Four internal clock sources increase system integration:
  - 3 to 74.7 MHz Internal Main Oscillator (IMO) ±1% at 3 MHz
  - 1 kHz, 33 kHz, and 100 kHz Internal Low Speed Oscillator (ILO) outputs
  - 12 to 80 MHz clock doubler output, sourced from IMO, MHz External Crystal Oscillator (MHzECO), and Digital System Interconnect (DSI)
  - 24 to 80 MHz fractional Phase-Locked Loop (PLL) sourced from IMO, MHzECO, and DSI
- Clock generated using a DSI signal from an external I/O pin or other logic
- Two external clock sources provide high precision clocks:
  - 4 to 25 MHz External Crystal Oscillator (MHzECO)
  - 32.768 kHz External Crystal Oscillator (kHzECO) for Real Time Clock (RTC)
- Dedicated 16-bit divider for bus clock
- Eight individually sourced 16-bit clock dividers for the digital system peripherals
- Four individually sourced 16-bit clock dividers with skew for the analog system peripherals
- IMO has a USB mode that synchronizes to USB host traffic, requiring no external crystal for USB. (USB equipped parts only)

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 Freq	Nominal Freq	Accuracy (%)	Start at Reset	Enabled
BUS_CLK	DIGITAL	MASTER_CLK	? MHz	24 MHz	±1	True	True
PLL_OUT	DIGITAL	IMO	24 MHz	24 MHz	±1	True	True
MASTER_CLK	DIGITAL	PLL_OUT	? MHz	24 MHz	±1	True	True
IMO	DIGITAL		3 MHz	3 MHz	±1	True	True
ILO	DIGITAL		? MHz	1 kHz	-50,+100	True	True
USB_CLK	DIGITAL	IMO	48 MHz	? MHz	±0	False	False
XTAL	DIGITAL		24 MHz	? MHz	±0	False	False
XTAL 32kHz	DIGITAL		32.768 kHz	? MHz	±0	False	False
Digital Signal	DIGITAL		? 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

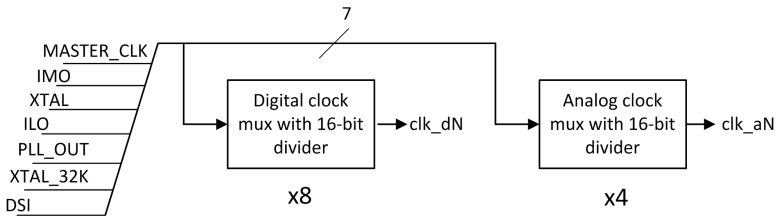


Table 10 lists the local clocks used in this design.

Table 10. Local Clocks

Name	Domain	Source	Desired Freq	Nominal Freq	Accuracy (%)	Start at Reset	Enabled
ADC_DelSig_-Ext_CP_Clk	DIGITAL	MASTER_CLK	? MHz	24 MHz	±1	True	True
I2C_Clock	DIGITAL	PLL_OUT	6.4 MHz	6 MHz	±1	True	True
SPI_IntClock	DIGITAL	MASTER_CLK	4 MHz	4 MHz	±1	True	True
ADC_Clock	ANALOG	MASTER_CLK	3 MHz	3 MHz	±1	True	True
DEBUG_-UART_IntClock	DIGITAL	MASTER_CLK	921.6 kHz	923.077 kHz	±1	True	True

For more information on clocking resources, please refer to:

- Clocking System chapter in the PSoC 5LP Technical Reference Manual
- Clocking chapter in the System Reference Guide
  - o CyPLL API routines
  - o CyIMO API routines
  - o CyILO API routines

- CyMaster API routines
- CyXTAL API routines

5 Interrupts and DMAs

5.1 Interrupts

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

Table 11. Interrupts

Name	Intr Num	Vector	Priority
SPI_RX_ISR	0	0	7
SPI_SS_ISR	1	1	7
SPI_TX_ISR	2	2	7
ADC_DelSig_IRQ	3	3	7
I2C_I2C_IRQ	4	4	7
SPI_TxInternalInterrupt	5	5	7
ADC_ISR	29	29	7

For more information on interrupts, please refer to:

- Interrupt Controller chapter in the PSoC 5LP Technical Reference Manual
- Interrupts chapter in the System Reference Guide
  - CyInt API routines and related registers
- Datasheet for cy_isr component

5.2 DMAs

This design contains no DMA components.

6 Flash Memory

PSoC 5LP 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 Address	End Address	Protection Level
0x0	0x3FFFF	U - Unprotected

Flash memory is organized as rows with each row of flash having 256 bytes. Each flash row can be assigned one of four protection levels:

- U - Unprotected
- F - Factory Upgrade
- R - Field Upgrade
- W - Full Protection

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

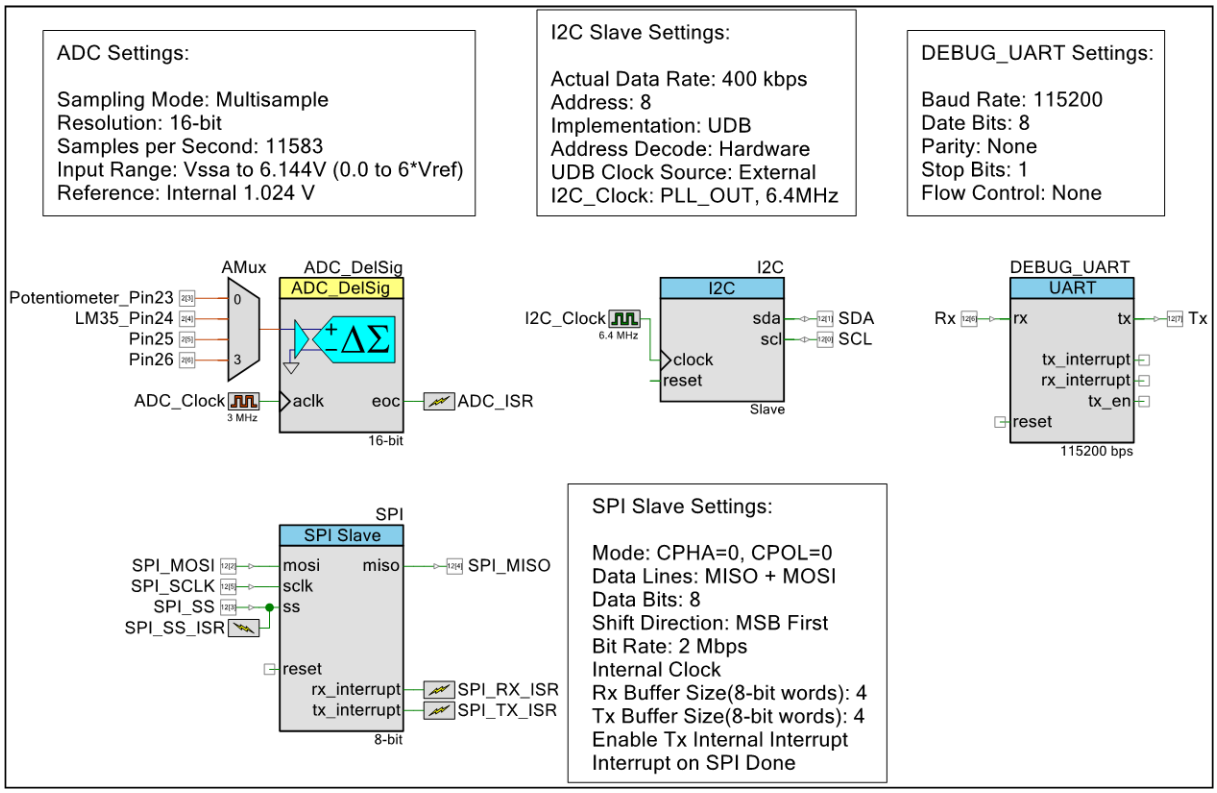
- Flash Protection chapter in the PSoC 5LP Technical Reference Manual
- Flash and EEPROM chapter in the System Reference Guide
  - CyWrite API routines
  - CyFlash API routines

7 Design Contents

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

7.1 Schematic Sheet: Page 1

Figure 5. Schematic Sheet: Page 1



This schematic sheet contains the following component instances:

- Instance ADC_DelSig (type: ADC_DelSig_v3_30)
- Instance AMux (type: AMux_v1_80)
- Instance DEBUG_UART (type: UART_v2_50)
- Instance I2C (type: I2C_v3_50)
- Instance SPI (type: SPI_Slave_v2_70)

8 Components

8.1 Component type: ADC_DelSig [v3.30]

8.1.1 Instance ADC_DelSig

Description: Delta-Sigma ADC
Instance type: ADC_DelSig [v3.30]
Datasheet: online component datasheet for ADC_DelSig

Table 13. Component Parameters for ADC_DelSig

Parameter Name	Value	Description
ADC_Alignment	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Alignment_Config2	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Alignment_Config3	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Alignment_Config4	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Charge_Pump_Clock	true	Low power charge pump clock selection
ADC_Clock	External	Parameter for selecting the ADC clock type.
ADC_Input_Mode	Single	Differential or Single ended input mode
ADC_Input_Range	0.0 to 6*Vref	Choose input operating mode that best supports the range of the signals being measured.
ADC_Input_Range_Config2	0.0 to Vref	Choose input operating mode that best supports the range of the signals being measured.
ADC_Input_Range_Config3	0.0 to Vref	Choose input operating mode that best supports the range of the signals being measured.
ADC_Input_Range_Config4	0.0 to Vref	Choose input operating mode that best supports the range of the signals being measured.
ADC_Power	Medium Power	Sets power level of ADC.
ADC_Reference	Internal 1.024 Volts	Selects voltage reference source and configuration.
ADC_Reference_Config2	Internal 1.024 Volts	Selects voltage reference source and configuration.
ADC_Reference_Config3	Internal 1.024 Volts	Selects voltage reference source and configuration.
ADC_Reference_Config4	Internal 1.024 Volts	Selects voltage reference source and configuration.
ADC_Resolution	16	ADC Resolution in bits
ADC_Resolution_Config2	16	ADC Resolution in bits
ADC_Resolution_Config3	16	ADC Resolution in bits
ADC_Resolution_Config4	16	ADC Resolution in bits

Parameter Name	Value	Description
Clock_Frequency	64000	Determines the ADC clock frequency.
Comment_Config1	Default Config	Parameter which holds the user comment for the config1.
Comment_Config2	Second Config	Parameter which holds the user comment for the config2.
Comment_Config3	Third Config	Parameter which holds the user comment for the config3.
Comment_Config4	Fourth Config	Parameter which holds the user comment for the config4.
Config1_Name	CFG1	This parameter is used to create constants in the header file for config 1.
Config2_Name	CFG2	This parameter is used to create constants in the header file for config 2.
Config3_Name	CFG3	This parameter is used to create constants in the header file for config 3.
Config4_Name	CFG4	This parameter is used to create constants in the header file for config 4.
Configs	4	Number of active configurations
Conversion_Mode	1 - Multi Sample	ADC conversion mode
Conversion_Mode_Config2	2 - Continuous	ADC conversion mode
Conversion_Mode_Config3	2 - Continuous	ADC conversion mode
Conversion_Mode_Config4	2 - Continuous	ADC conversion mode
Enable_Vref_Vss	false	Determines whether or not to connect ADC's reference Vssa to AGL[6].
EnableModulatorInput	false	When this parameter is enabled, the modulator input terminal will be enabled on the symbol.
Input_Buffer_Gain	1	Gain of input amplifier
Input_Buffer_Gain_Config2	1	Gain of input amplifier
Input_Buffer_Gain_Config3	1	Gain of input amplifier
Input_Buffer_Gain_Config4	1	Gain of input amplifier
Input_Buffer_Mode	Rail to Rail	Buffer Mode type selection
Input_Buffer_Mode_Config2	Rail to Rail	Buffer Mode type selection
Input_Buffer_Mode_Config3	Rail to Rail	Buffer Mode type selection
Input_Buffer_Mode_Config4	Rail to Rail	Buffer Mode type selection
Ref_Voltage	1.024	Set reference voltage
Ref_Voltage_Config2	1.024	Set reference voltage
Ref_Voltage_Config3	1.024	Set reference voltage
Ref_Voltage_Config4	1.024	Set reference voltage
rm_int	false	Removes internal interrupt (IRQ)
Sample_Rate	10000	Sample Rate in Hz
Sample_Rate_Config2	10000	Sample Rate in Hz
Sample_Rate_Config3	10000	Sample Rate in Hz
Sample_Rate_Config4	10000	Sample Rate in Hz
Start_of_Conversion	Software	Continuous conversions or hardware controlled
User Comments		Instance-specific comments.

8.2 Component type: AMux [v1.80]

8.2.1 Instance AMux

Description: Multiplexer used to route analog signals.
Instance type: AMux [v1.80]
Datasheet: online component datasheet for AMux

Table 14. Component Parameters for AMux

Parameter Name	Value	Description
AtMostOneActive	false	Limit to at most one active channel.
Channels	4	Channel count.
Isolation	Medium	Specify minimum, medium, or maximum switch control; affects channel isolation and switching time.
MuxType	Single	Select between single or differential inputs.
User Comments		Instance-specific comments.

8.3 Component type: I2C [v3.50]

8.3.1 Instance I2C

Description: Standard I2C communication interface
Instance type: I2C [v3.50]
Datasheet: online component datasheet for I2C

Table 15. Component Parameters for I2C

Parameter Name	Value	Description
Address_Decode	Hardware	Determines either hardware or software address match logic.
BusSpeed_kHz	100	I2C Data Rate in kbps. Standard settings are 50, 100, 400 or 1000. The value must be between 1 and 1000.
EnableWakeup	false	Determines if I2C is selected as wakeup source.
ExternalBuffer	false	Exposes scl and sda in and out terminals outside the component.
Externi2cIntrHandler	false	Allows I2C interrupt handler to be set outside the I2C component. This feature intended only for PM/SM bus usage.
ExternTmoutIntrHandler	false	Allows I2C timeout interrupt handler to be set outside the I2C component. This feature intended only for PM/SM bus usage.
Hex	true	Indicates that address has been input in hexadecimal format.

Parameter Name	Value	Description
I2C_Mode	Slave	Determines I2C mode (Slave/Master/Multi-Master/Multi-Master-Slave).
I2cBusPort	Any	Determines which I2C pins have been selected. Select I2C0/I2C1 and connect to corresponding pins to be able use I2C as wakeup source.
Implementation	UDB	Determines either I2C implementation Fixed Function or UDB.
NotSlaveClockMinusTolerance	25	Internal component clock negative tolerance value in Master, Multi-Master or Multi-Master-Slave mode.
NotSlaveClockPlusTolerance	5	Internal component clock positive tolerance value in Master, Multi-Master or Multi-Master-Slave mode.
PrescalerEnabled	false	Enables prescaler (7-bit counter) to expand timeout timer range.
PrescalerPeriod	3	Prescaler period of timeout timer.
SclTimeoutEnabled	false	Enables low time monitoring of scl line.
SdaTimeoutEnabled	false	Enables low time monitoring of sda line.
Slave_Address	72	7-bits I2C slave address.
SlaveClockMinusTolerance	5	Internal component clock negative tolerance value in Slave mode.
SlaveClockPlusTolerance	50	Internal component clock positive tolerance value in Slave mode.
TimeoutImplementation	UDB	Determines either timeout timer feature implementation as UDB or Fixed Function. The Fixed Function implementation only available for PSoC5LP.
TimeOutms	25	Determines maximum time allowed for scl or sda to be low state (in mS). The timeout timer generates interrupt after timeout expires.
TimeoutPeriodff	39999	Period of timeout timer (Fixed Function).
TimeoutPeriodUdb	39999	Period of timeout timer (UDB).
UdbInternalClock	false	Determines either internal or external clock source for I2C UDB.
UdbSlaveFixedPlacementEnable	false	Enables fixed placement for I2C UDB. Only available in slave mode.
User Comments		Instance-specific comments.

8.4 Component type: SPI_Slave [v2.70]

8.4.1 Instance SPI

Description: Serial Peripheral Interface Slave
Instance type: SPI_Slave [v2.70]
Datasheet: online component datasheet for SPI_Slave

Table 16. Component Parameters for SPI

Parameter Name	Value	Description
BidirectMode	false	Bidirectional mode setting
ClockInternal	true	Defines whether internal clock is used or not
DesiredBitRate	2000000	Desired Bit Rate in Hz
FixedPlacementEnabled	false	
InterruptOnByteComplete	false	Set Initial Interrupt Source to Enable Interrupt on Byte Transfer Complete
InterruptOnDone	true	Set Initial Interrupt Source to Enable Interrupt on SPI Done
InterruptOnRXEmpty	false	Set Initial Interrupt Source to Enable Interrupt on RX FIFO Empty
InterruptOnRXFull	false	Set Initial Interrupt Source to Enable Interrupt on RX FIFO full
InterruptOnRXNotEmpty	false	Set Initial Interrupt Source to Enable Interrupt on RX Not Empty
InterruptOnRXOverrun	false	Set Initial Interrupt Source to Enable Interrupt on RX FIFO overrun
InterruptOnTXEmpty	false	Set Initial Interrupt Source to Enable Interrupt on TX FIFO Empty
InterruptOnTXFull	false	Set Initial Interrupt Source to Enable Interrupt on TX FIFO full
InterruptOnTXNotFull	false	Set Initial Interrupt Source to Enable Interrupt on TX FIFO not full
Mode	CPHA = 0, CPOL = 0	Allows for setting the SPI Clock Polarity and Clock Phase from one of the four well known modes
MultiSlaveEnable	false	Allows using of the SPI MISO output enable terminal for multislave mode support
NumberOfDataBits	8	Data Width (3-16 bits)
RxBufferSize	4	RAM size used to store RX Data
ShiftDir	MSB First	Data Shift Direction (MSB First or LSB First)
TxBufferSize	4	RAM size used to store TX Data
UseInternalInterrupt	false	Defines whether internal interrupt is used or not
User Comments		Instance-specific comments.
UseRxInternalInterrupt	false	Defines whether Rx internal interrupt is used or not
UseTxInternalInterrupt	true	Defines whether Tx internal interrupt is used or not

8.5 Component type: UART [v2.50]

8.5.1 Instance DEBUG_UART

Description: Universal Asynchronous Receiver Transmitter
Instance type: UART [v2.50]
Datasheet: online component datasheet for UART

Table 17. Component Parameters for DEBUG_UART

Parameter Name	Value	Description
Address1	0	This parameter specifies the RX Hardware Address #1.
Address2	0	This parameter specifies the RX Hardware Address #2.
BaudRate	115200	Sets the target baud rate.
BreakBitsRX	13	Specifies the break signal length for the RX (detection) channel.
BreakBitsTX	13	Specifies the break signal length for the TX channel.
BreakDetect	false	Enables the break detect hardware.
CRCoutputsEn	false	Enables the CRC outputs.
EnIntRXInterrupt	false	Enables the internal RX interrupt configuration and the ISR.
EnIntTXInterrupt	false	Enables the internal TX interrupt configuration and the ISR.
FlowControl	None	Enable the flow control signals.
HalfDuplexEn	false	Enables half duplex mode on the RX Half of the UART module.
HwTXEnSignal	true	Enables the external TX enable signal output.
InternalClock	true	Enables the internal clock. This parameter removes the clock input pin.
InterruptOnTXComplete	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX complete' event.
InterruptOnTXFifoEmpty	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX FIFO empty' event.
InterruptOnTXFifoFull	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX FIFO full' event.
InterruptOnTXFifoNotFull	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX FIFO not full' event.
IntOnAddressDetect	false	Enables the interrupt on hardware address detected event by default
IntOnAddressMatch	false	Enables the interrupt on hardware address match detected event by default
IntOnBreak	false	Enables the interrupt on break signal detected event by default

Parameter Name	Value	Description
IntOnByteRcvd	true	Enables the interrupt on RX byte received event by default
IntOnOverrunError	false	Enables the interrupt on overrun error event by default
IntOnParityError	false	Enables the interrupt on parity error event by default
IntOnStopError	false	Enables the interrupt on stop error event by default
NumDataBits	8	Defines the number of data bits. Values can be 5, 6, 7 or 8 bits.
NumStopBits	1	Defines the number of stop bits. Values can be 1 or 2 bits.
OverSamplingRate	8	This parameter defines the over sampling rate.
ParityType	None	Sets the parity type as Odd, Even or Mark/Space
ParityTypeSw	false	This parameter allows the parity type to be changed through software by using the WriteControlRegister API
RXAddressMode	None	Configures the RX hardware address detection mode
RXBufferSize	4	The size of the RAM space allocated for the RX input buffer.
RXEnable	true	Enables the RX in the UART
TXBitClkGenDP	true	When enabled, this parameter enables the TX clock generation on DataPath resource. When disabled, TX clock is generated from Clock7.
TXBufferSize	4	The size of the RAM space allocated for the TX output buffer.
TXEnable	true	Enables the TX in the UART
Use23Polling	true	Allows the use of 2 out of 3 polling resources on the RX UART sampler.
User Comments		Instance-specific comments.

9 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
  - The full PSoC 5LP register map is covered in the PSoC 5LP Registers Technical Reference Manual
  - Register Access chapter in the System Reference Guide
    - CY_GET API routines
    - CY_SET API routines
- System Functions chapter in the System Reference Guide
  - General API routines
  - CyDelay API routines
  - CyVd Voltage Detect API routines
- Power Management
  - Power Supply and Monitoring chapter in the PSoC 5LP Technical Reference Manual
  - Low Power Modes chapter in the PSoC 5LP Technical Reference Manual
  - Power Management chapter in the System Reference Guide
    - CyPm API routines
- Watchdog Timer chapter in the System Reference Guide
  - CyWdt API routines
- Cache Management
  - Cache Controller chapter in the PSoC 5LP Technical Reference Manual
  - Cache chapter in the System Reference Guide
    - CyFlushCache() API routine