PSoC® Creator™
Project Datasheet for 04_CAN_Basic

Creation Time: 03/21/2017 12:49:54
User: DESKTOP-4FI6UM3\lilu1
Project: 04_CAN_Basic
Tool: PSoC Creator 4.0 Update 1

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

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

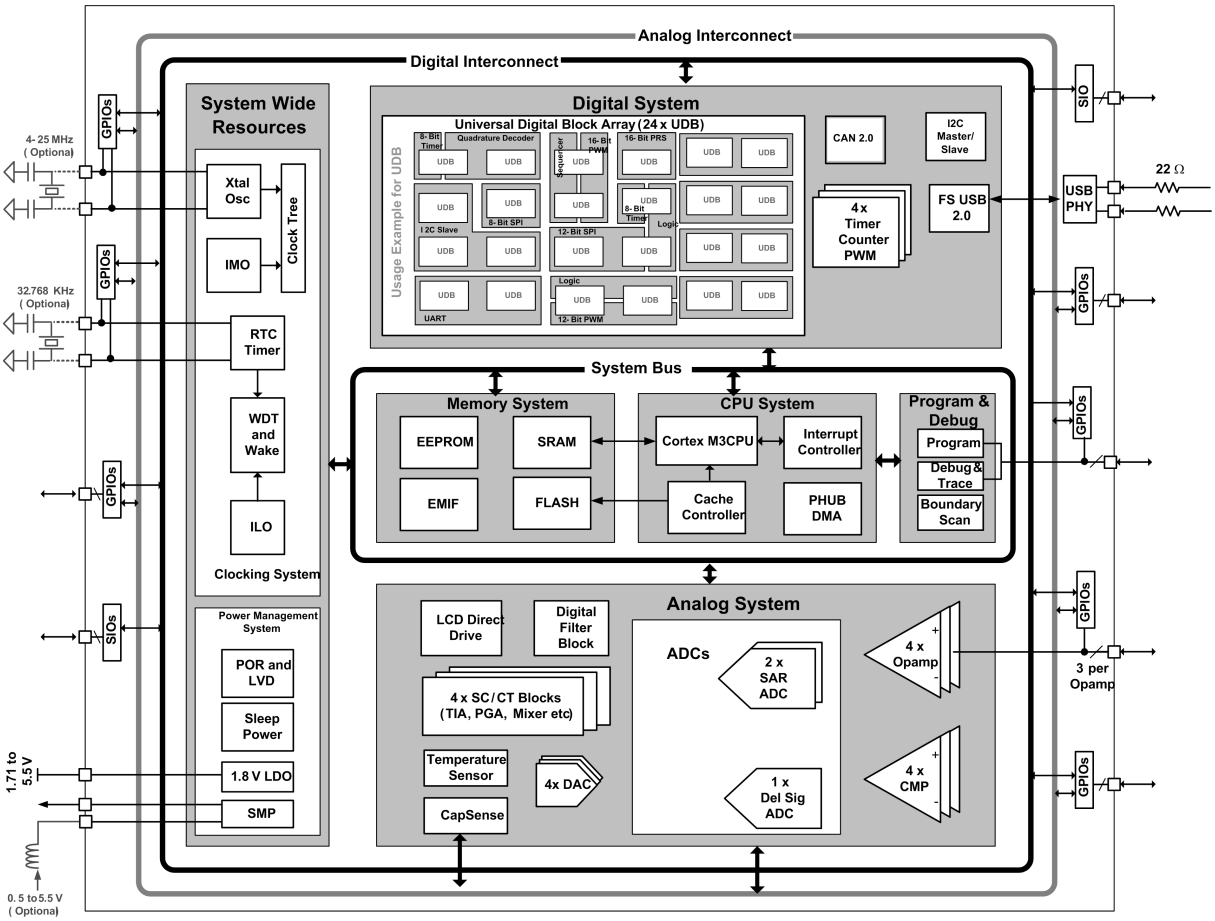


Table 1 lists the key characteristics of this device.

Table 1. Device Characteristics

Name	Value
Part Number	CY8C5888AXI-LP096
Package Name	100-TQFP
Family	PSoC 5LP
Series	CY8C58LP
CPU speed (MHz)	80
Flash size (kBytes)	256
SRAM size (kBytes)	64
EEPROM size (Bytes)	2048
Vdd range (V)	1.71 to 5.5
Automotive qualified	No (Industrial Grade Only)
Temp range (Celcius)	-40 to 85
JTAG ID	0x2E160069

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

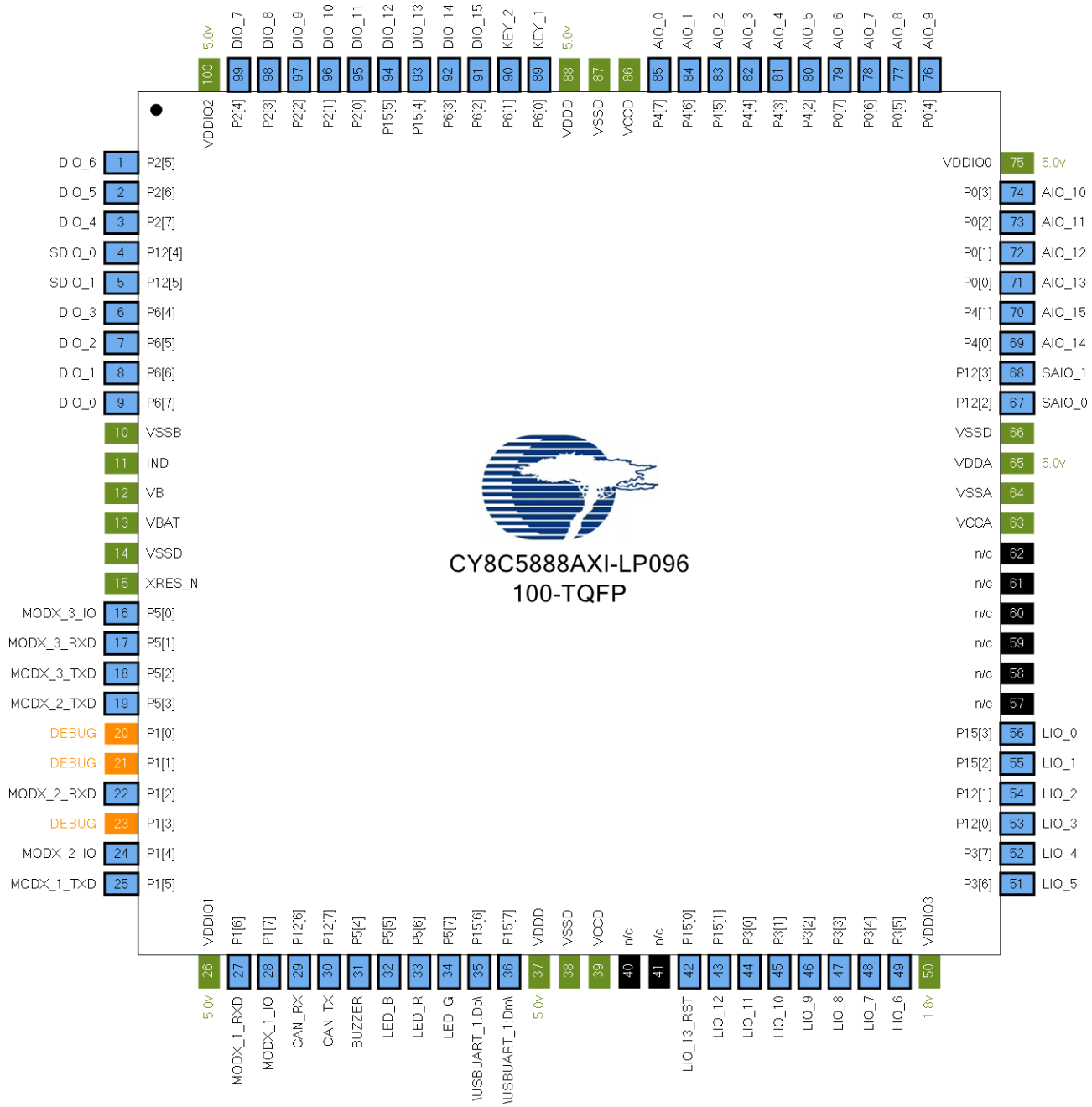
Resource Type	Used	Free	Max	% Used
Digital Clocks	2	6	8	25.00 %
Analog Clocks	1	3	4	25.00 %
CapSense Buffers	0	2	2	0.00 %
Digital Filter Block	0	1	1	0.00 %
Interrupts	9	23	32	28.13 %
IO	72	0	72	100.00 %
Segment LCD	0	1	1	0.00 %
CAN 2.0b	1	0	1	100.00 %
I2C	0	1	1	0.00 %
USB	1	0	1	100.00 %
DMA Channels	0	24	24	0.00 %
Timer	0	4	4	0.00 %
UDB				
Macrocells	126	66	192	65.63 %
Unique P-terms	229	155	384	59.64 %
Total P-terms	274			
Datapath Cells	16	8	24	66.67 %
Status Cells	16	8	24	66.67 %
StatusI Registers	11			
Routed Count7 Load/Enable	5			
Control Cells	6	18	24	25.00 %
Control Registers	1			
Count7 Cells	5			
Opamp	0	4	4	0.00 %
Comparator	0	4	4	0.00 %
Delta-Sigma ADC	0	1	1	0.00 %
LPF	0	2	2	0.00 %
SAR ADC	0	2	2	0.00 %
Analog (SC/CT) Blocks	0	4	4	0.00 %
DAC				

Resource Type	Used	Free	Max	% Used
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[5]	DIO_6	Software In/Out	OD, DL	HiZ Analog Unb
2	P2[6]	DIO_5	Software In/Out	OD, DL	HiZ Analog Unb
3	P2[7]	DIO_4	Software In/Out	OD, DL	HiZ Analog Unb
4	P12[4]	SDIO_0	Dgtl In	OD, DL	HiZ Analog Unb
5	P12[5]	SDIO_1	Dgtl Out	Strong drive	HiZ Analog Unb
6	P6[4]	DIO_3	Software In/Out	OD, DL	HiZ Analog Unb
7	P6[5]	DIO_2	Software In/Out	OD, DL	HiZ Analog Unb
8	P6[6]	DIO_1	Software In/Out	OD, DL	HiZ Analog Unb
9	P6[7]	DIO_0	Software In/Out	OD, DL	HiZ Analog Unb
10	VSSB	VSSB	Dedicated		
11	IND	IND	Dedicated		
12	VB	VB	Dedicated		
13	VBAT	VBAT	Dedicated		
14	VSSD	VSSD	Power		
15	XRES_N	XRES_N	Dedicated		
16	P5[0]	MODX_3_IO	Software In/Out	OD, DL	HiZ Analog Unb
17	P5[1]	MODX_3_RXD	Dgtl In	HiZ digital	HiZ Analog Unb
18	P5[2]	MODX_3_TXD	Dgtl Out	Strong drive	HiZ Analog Unb
19	P5[3]	MODX_2_TXD	Dgtl Out	Strong drive	HiZ Analog Unb
20	P1[0]	Debug:SWD_IO	Reserved		
21	P1[1]	Debug:SWD_CK	Reserved		
22	P1[2]	MODX_2_RXD	Dgtl In	HiZ digital	HiZ Analog Unb
23	P1[3]	Debug:SWV	Reserved		
24	P1[4]	MODX_2_IO	Software In/Out	OD, DL	HiZ Analog Unb
25	P1[5]	MODX_1_TXD	Dgtl Out	Strong drive	HiZ Analog Unb
26	VDDIO1	VDDIO1	Power		
27	P1[6]	MODX_1_RXD	Dgtl In	HiZ digital	HiZ Analog Unb
28	P1[7]	MODX_1_IO	Software In/Out	OD, DL	HiZ Analog Unb
29	P12[6]	CAN_RX	Dgtl In	HiZ digital	HiZ Analog Unb
30	P12[7]	CAN_TX	Dgtl Out	Strong drive	HiZ Analog Unb
31	P5[4]	BUZZER	Dgtl Out	Strong drive	HiZ Analog Unb
32	P5[5]	LED_B	Dgtl Out	Strong drive	HiZ Analog Unb
33	P5[6]	LED_R	Dgtl Out	Strong drive	HiZ Analog Unb
34	P5[7]	LED_G	Dgtl Out	Strong drive	HiZ Analog Unb
35	P15[6]	\USBUART_1:Dp\	Analog	HiZ analog	HiZ Analog Unb
36	P15[7]	\USBUART_1:Dm\	Analog	HiZ analog	HiZ Analog Unb
37	VDDD	VDDD	Power		

Pin	Port	Name	Type	Drive Mode	Reset State
38	VSSD	VSSD	Power		
39	VCCD	VCCD	Power		
42	P15[0]	LIO_13_RST	Software In/Out	OD, DL	HiZ Analog Unb
43	P15[1]	LIO_12	Software In/Out	OD, DL	HiZ Analog Unb
44	P3[0]	LIO_11	Software In/Out	OD, DL	HiZ Analog Unb
45	P3[1]	LIO_10	Software In/Out	OD, DL	HiZ Analog Unb
46	P3[2]	LIO_9	Software In/Out	OD, DL	HiZ Analog Unb
47	P3[3]	LIO_8	Software In/Out	OD, DL	HiZ Analog Unb
48	P3[4]	LIO_7	Software In/Out	OD, DL	HiZ Analog Unb
49	P3[5]	LIO_6	Software In/Out	OD, DL	HiZ Analog Unb
50	VDDIO3	VDDIO3	Power		
51	P3[6]	LIO_5	Software In/Out	OD, DL	HiZ Analog Unb
52	P3[7]	LIO_4	Software In/Out	OD, DL	HiZ Analog Unb
53	P12[0]	LIO_3	Software In/Out	OD, DL	HiZ Analog Unb
54	P12[1]	LIO_2	Software In/Out	OD, DL	HiZ Analog Unb
55	P15[2]	LIO_1	Software In/Out	OD, DL	HiZ Analog Unb
56	P15[3]	LIO_0	Dgtl In	Res pull up/down	HiZ Analog Unb
63	VCCA	VCCA	Power		
64	VSSA	VSSA	Power		
65	VDDA	VDDA	Power		
66	VSSD	VSSD	Power		
67	P12[2]	SAIO_0	Dgtl In	OD, DL	HiZ Analog Unb
68	P12[3]	SAIO_1	Dgtl Out	Strong drive	HiZ Analog Unb
69	P4[0]	AIO_14	Analog	HiZ analog	HiZ Analog Unb
70	P4[1]	AIO_15	Analog	HiZ analog	HiZ Analog Unb
71	P0[0]	AIO_13	Analog	HiZ analog	HiZ Analog Unb
72	P0[1]	AIO_12	Analog	HiZ analog	HiZ Analog Unb
73	P0[2]	AIO_11	Analog	HiZ analog	HiZ Analog Unb
74	P0[3]	AIO_10	Analog	HiZ analog	HiZ Analog Unb
75	VDDIO0	VDDIO0	Power		
76	P0[4]	AIO_9	Analog	HiZ analog	HiZ Analog Unb
77	P0[5]	AIO_8	Analog	HiZ analog	HiZ Analog Unb
78	P0[6]	AIO_7	Analog	HiZ analog	HiZ Analog Unb
79	P0[7]	AIO_6	Analog	HiZ analog	HiZ Analog Unb
80	P4[2]	AIO_5	Analog	HiZ analog	HiZ Analog Unb
81	P4[3]	AIO_4	Analog	HiZ analog	HiZ Analog Unb
82	P4[4]	AIO_3	Analog	HiZ analog	HiZ Analog Unb
83	P4[5]	AIO_2	Analog	HiZ analog	HiZ Analog Unb
84	P4[6]	AIO_1	Analog	HiZ analog	HiZ Analog Unb
85	P4[7]	AIO_0	Analog	HiZ analog	HiZ Analog Unb
86	VCCD	VCCD	Power		

Pin	Port	Name	Type	Drive Mode	Reset State
87	VSSD	VSSD	Power		
88	VDDD	VDDD	Power		
89	P6[0]	KEY_1	Software Input	Res pull up	HiZ Analog Unb
90	P6[1]	KEY_2	Software Input	Res pull up	HiZ Analog Unb
91	P6[2]	DIO_15	Software In/Out	OD, DL	HiZ Analog Unb
92	P6[3]	DIO_14	Software In/Out	OD, DL	HiZ Analog Unb
93	P15[4]	DIO_13	Software In/Out	OD, DL	HiZ Analog Unb
94	P15[5]	DIO_12	Software In/Out	OD, DL	HiZ Analog Unb
95	P2[0]	DIO_11	Software In/Out	OD, DL	HiZ Analog Unb
96	P2[1]	DIO_10	Software In/Out	OD, DL	HiZ Analog Unb
97	P2[2]	DIO_9	Software In/Out	OD, DL	HiZ Analog Unb
98	P2[3]	DIO_8	Software In/Out	OD, DL	HiZ Analog Unb
99	P2[4]	DIO_7	Software In/Out	OD, DL	HiZ Analog Unb
100	VDDIO2	VDDIO2	Power		

Abbreviations used in Table 3 have the following meanings:

- OD, DL = Open drain, drives low
- HiZ Analog Unb = Hi-Z Analog Unbuffered
- Dgtl In = Digital Input
- Dgtl Out = Digital Output
- HiZ digital = High impedance digital
- HiZ analog = High impedance analog
- Res pull up/down = Resistive pull up/down
- 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]	71	AIO_13	Analog	HiZ analog	HiZ Analog Unb
P0[1]	72	AIO_12	Analog	HiZ analog	HiZ Analog Unb
P0[2]	73	AIO_11	Analog	HiZ analog	HiZ Analog Unb
P0[3]	74	AIO_10	Analog	HiZ analog	HiZ Analog Unb
P0[4]	76	AIO_9	Analog	HiZ analog	HiZ Analog Unb
P0[5]	77	AIO_8	Analog	HiZ analog	HiZ Analog Unb
P0[6]	78	AIO_7	Analog	HiZ analog	HiZ Analog Unb
P0[7]	79	AIO_6	Analog	HiZ analog	HiZ Analog Unb
P1[0]	20	Debug:SWD_IO	Reserved		
P1[1]	21	Debug:SWD_CK	Reserved		
P1[2]	22	MODX_2_RXD	Dgtl In	HiZ digital	HiZ Analog Unb
P1[3]	23	Debug:SWV	Reserved		
P1[4]	24	MODX_2_IO	Software In/Out	OD, DL	HiZ Analog Unb
P1[5]	25	MODX_1_TXD	Dgtl Out	Strong drive	HiZ Analog Unb
P1[6]	27	MODX_1_RXD	Dgtl In	HiZ digital	HiZ Analog Unb
P1[7]	28	MODX_1_IO	Software In/Out	OD, DL	HiZ Analog Unb
P12[0]	53	LIO_3	Software In/Out	OD, DL	HiZ Analog Unb
P12[1]	54	LIO_2	Software In/Out	OD, DL	HiZ Analog Unb
P12[2]	67	SAIO_0	Dgtl In	OD, DL	HiZ Analog Unb
P12[3]	68	SAIO_1	Dgtl Out	Strong drive	HiZ Analog Unb
P12[4]	4	SDIO_0	Dgtl In	OD, DL	HiZ Analog Unb
P12[5]	5	SDIO_1	Dgtl Out	Strong drive	HiZ Analog Unb
P12[6]	29	CAN_RX	Dgtl In	HiZ digital	HiZ Analog Unb
P12[7]	30	CAN_TX	Dgtl Out	Strong drive	HiZ Analog Unb
P15[0]	42	LIO_13_RST	Software In/Out	OD, DL	HiZ Analog Unb
P15[1]	43	LIO_12	Software In/Out	OD, DL	HiZ Analog Unb
P15[2]	55	LIO_1	Software In/Out	OD, DL	HiZ Analog Unb
P15[3]	56	LIO_0	Dgtl In	Res pull up/down	HiZ Analog Unb
P15[4]	93	DIO_13	Software In/Out	OD, DL	HiZ Analog Unb
P15[5]	94	DIO_12	Software In/Out	OD, DL	HiZ Analog Unb
P15[6]	35	\USBUART_1:Dp\	Analog	HiZ analog	HiZ Analog Unb
P15[7]	36	\USBUART_1:Dm\	Analog	HiZ analog	HiZ Analog Unb
P2[0]	95	DIO_11	Software In/Out	OD, DL	HiZ Analog Unb
P2[1]	96	DIO_10	Software In/Out	OD, DL	HiZ Analog Unb

Port	Pin	Name	Type	Drive Mode	Reset State
P2[2]	97	DIO_9	Software In/Out	OD, DL	HiZ Analog Unb
P2[3]	98	DIO_8	Software In/Out	OD, DL	HiZ Analog Unb
P2[4]	99	DIO_7	Software In/Out	OD, DL	HiZ Analog Unb
P2[5]	1	DIO_6	Software In/Out	OD, DL	HiZ Analog Unb
P2[6]	2	DIO_5	Software In/Out	OD, DL	HiZ Analog Unb
P2[7]	3	DIO_4	Software In/Out	OD, DL	HiZ Analog Unb
P3[0]	44	LIO_11	Software In/Out	OD, DL	HiZ Analog Unb
P3[1]	45	LIO_10	Software In/Out	OD, DL	HiZ Analog Unb
P3[2]	46	LIO_9	Software In/Out	OD, DL	HiZ Analog Unb
P3[3]	47	LIO_8	Software In/Out	OD, DL	HiZ Analog Unb
P3[4]	48	LIO_7	Software In/Out	OD, DL	HiZ Analog Unb
P3[5]	49	LIO_6	Software In/Out	OD, DL	HiZ Analog Unb
P3[6]	51	LIO_5	Software In/Out	OD, DL	HiZ Analog Unb
P3[7]	52	LIO_4	Software In/Out	OD, DL	HiZ Analog Unb
P4[0]	69	AIO_14	Analog	HiZ analog	HiZ Analog Unb
P4[1]	70	AIO_15	Analog	HiZ analog	HiZ Analog Unb
P4[2]	80	AIO_5	Analog	HiZ analog	HiZ Analog Unb
P4[3]	81	AIO_4	Analog	HiZ analog	HiZ Analog Unb
P4[4]	82	AIO_3	Analog	HiZ analog	HiZ Analog Unb
P4[5]	83	AIO_2	Analog	HiZ analog	HiZ Analog Unb
P4[6]	84	AIO_1	Analog	HiZ analog	HiZ Analog Unb
P4[7]	85	AIO_0	Analog	HiZ analog	HiZ Analog Unb
P5[0]	16	MODX_3_IO	Software In/Out	OD, DL	HiZ Analog Unb
P5[1]	17	MODX_3_RXD	Dgtl In	HiZ digital	HiZ Analog Unb
P5[2]	18	MODX_3_TXD	Dgtl Out	Strong drive	HiZ Analog Unb
P5[3]	19	MODX_2_TXD	Dgtl Out	Strong drive	HiZ Analog Unb
P5[4]	31	BUZZER	Dgtl Out	Strong drive	HiZ Analog Unb
P5[5]	32	LED_B	Dgtl Out	Strong drive	HiZ Analog Unb
P5[6]	33	LED_R	Dgtl Out	Strong drive	HiZ Analog Unb
P5[7]	34	LED_G	Dgtl Out	Strong drive	HiZ Analog Unb
P6[0]	89	KEY_1	Software Input	Res pull up	HiZ Analog Unb
P6[1]	90	KEY_2	Software Input	Res pull up	HiZ Analog Unb
P6[2]	91	DIO_15	Software In/Out	OD, DL	HiZ Analog Unb
P6[3]	92	DIO_14	Software In/Out	OD, DL	HiZ Analog Unb
P6[4]	6	DIO_3	Software In/Out	OD, DL	HiZ Analog Unb

Port	Pin	Name	Type	Drive Mode	Reset State
P6[5]	7	DIO_2	Software In/Out	OD, DL	HiZ Analog Unb
P6[6]	8	DIO_1	Software In/Out	OD, DL	HiZ Analog Unb
P6[7]	9	DIO_0	Software In/Out	OD, DL	HiZ Analog Unb

Abbreviations used in Table 4 have the following meanings:

- HiZ analog = High impedance analog
- HiZ Analog Unb = Hi-Z Analog Unbuffered
- Dgtl In = Digital Input
- HiZ digital = High impedance digital
- OD, DL = Open drain, drives low
- Dgtl Out = Digital Output
- Res pull up/down = Resistive pull up/down
- Res pull up = Resistive pull up

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
\USBUART_1:Dm\	P15[7]	Analog	HiZ Analog Unb
\USBUART_1:Dp\	P15[6]	Analog	HiZ Analog Unb
AIO_0	P4[7]	Analog	HiZ Analog Unb
AIO_1	P4[6]	Analog	HiZ Analog Unb
AIO_10	P0[3]	Analog	HiZ Analog Unb
AIO_11	P0[2]	Analog	HiZ Analog Unb
AIO_12	P0[1]	Analog	HiZ Analog Unb
AIO_13	P0[0]	Analog	HiZ Analog Unb
AIO_14	P4[0]	Analog	HiZ Analog Unb
AIO_15	P4[1]	Analog	HiZ Analog Unb
AIO_2	P4[5]	Analog	HiZ Analog Unb
AIO_3	P4[4]	Analog	HiZ Analog Unb
AIO_4	P4[3]	Analog	HiZ Analog Unb
AIO_5	P4[2]	Analog	HiZ Analog Unb
AIO_6	P0[7]	Analog	HiZ Analog Unb
AIO_7	P0[6]	Analog	HiZ Analog Unb
AIO_8	P0[5]	Analog	HiZ Analog Unb
AIO_9	P0[4]	Analog	HiZ Analog Unb
BUZZER	P5[4]	Dgtl Out	HiZ Analog Unb
CAN_RX	P12[6]	Dgtl In	HiZ Analog Unb
CAN_TX	P12[7]	Dgtl Out	HiZ Analog Unb
Debug:SWD_CK	P1[1]	Reserved	
Debug:SWD_IO	P1[0]	Reserved	
Debug:SWV	P1[3]	Reserved	
DIO_0	P6[7]	Software In/Out	HiZ Analog Unb
DIO_1	P6[6]	Software In/Out	HiZ Analog Unb
DIO_10	P2[1]	Software In/Out	HiZ Analog Unb
DIO_11	P2[0]	Software In/Out	HiZ Analog Unb
DIO_12	P15[5]	Software In/Out	HiZ Analog Unb
DIO_13	P15[4]	Software In/Out	HiZ Analog Unb
DIO_14	P6[3]	Software In/Out	HiZ Analog Unb
DIO_15	P6[2]	Software In/Out	HiZ Analog Unb
DIO_2	P6[5]	Software In/Out	HiZ Analog Unb
DIO_3	P6[4]	Software In/Out	HiZ Analog Unb
DIO_4	P2[7]	Software In/Out	HiZ Analog Unb

Name	Port	Type	Reset State
DIO_5	P2[6]	Software In/Out	HiZ Analog Unb
DIO_6	P2[5]	Software In/Out	HiZ Analog Unb
DIO_7	P2[4]	Software In/Out	HiZ Analog Unb
DIO_8	P2[3]	Software In/Out	HiZ Analog Unb
DIO_9	P2[2]	Software In/Out	HiZ Analog Unb
KEY_1	P6[0]	Software Input	HiZ Analog Unb
KEY_2	P6[1]	Software Input	HiZ Analog Unb
LED_B	P5[5]	Dgtl Out	HiZ Analog Unb
LED_G	P5[7]	Dgtl Out	HiZ Analog Unb
LED_R	P5[6]	Dgtl Out	HiZ Analog Unb
LIO_0	P15[3]	Dgtl In	HiZ Analog Unb
LIO_1	P15[2]	Software In/Out	HiZ Analog Unb
LIO_10	P3[1]	Software In/Out	HiZ Analog Unb
LIO_11	P3[0]	Software In/Out	HiZ Analog Unb
LIO_12	P15[1]	Software In/Out	HiZ Analog Unb
LIO_13_RST	P15[0]	Software In/Out	HiZ Analog Unb
LIO_2	P12[1]	Software In/Out	HiZ Analog Unb
LIO_3	P12[0]	Software In/Out	HiZ Analog Unb
LIO_4	P3[7]	Software In/Out	HiZ Analog Unb
LIO_5	P3[6]	Software In/Out	HiZ Analog Unb
LIO_6	P3[5]	Software In/Out	HiZ Analog Unb
LIO_7	P3[4]	Software In/Out	HiZ Analog Unb
LIO_8	P3[3]	Software In/Out	HiZ Analog Unb
LIO_9	P3[2]	Software In/Out	HiZ Analog Unb
MODX_1_IO	P1[7]	Software In/Out	HiZ Analog Unb
MODX_1_RXD	P1[6]	Dgtl In	HiZ Analog Unb
MODX_1_TXD	P1[5]	Dgtl Out	HiZ Analog Unb
MODX_2_IO	P1[4]	Software In/Out	HiZ Analog Unb
MODX_2_RXD	P1[2]	Dgtl In	HiZ Analog Unb
MODX_2_TXD	P5[3]	Dgtl Out	HiZ Analog Unb
MODX_3_IO	P5[0]	Software In/Out	HiZ Analog Unb
MODX_3_RXD	P5[1]	Dgtl In	HiZ Analog Unb
MODX_3_TXD	P5[2]	Dgtl Out	HiZ Analog Unb

Name	Port	Type	Reset State
SAIO_0	P12[2]	Dgtl In	HiZ Analog Unb
SAIO_1	P12[3]	Dgtl Out	HiZ Analog Unb
SDIO_0	P12[4]	Dgtl In	HiZ Analog Unb
SDIO_1	P12[5]	Dgtl Out	HiZ Analog Unb

Abbreviations used in Table 5 have the following meanings:

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

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)	0x80
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

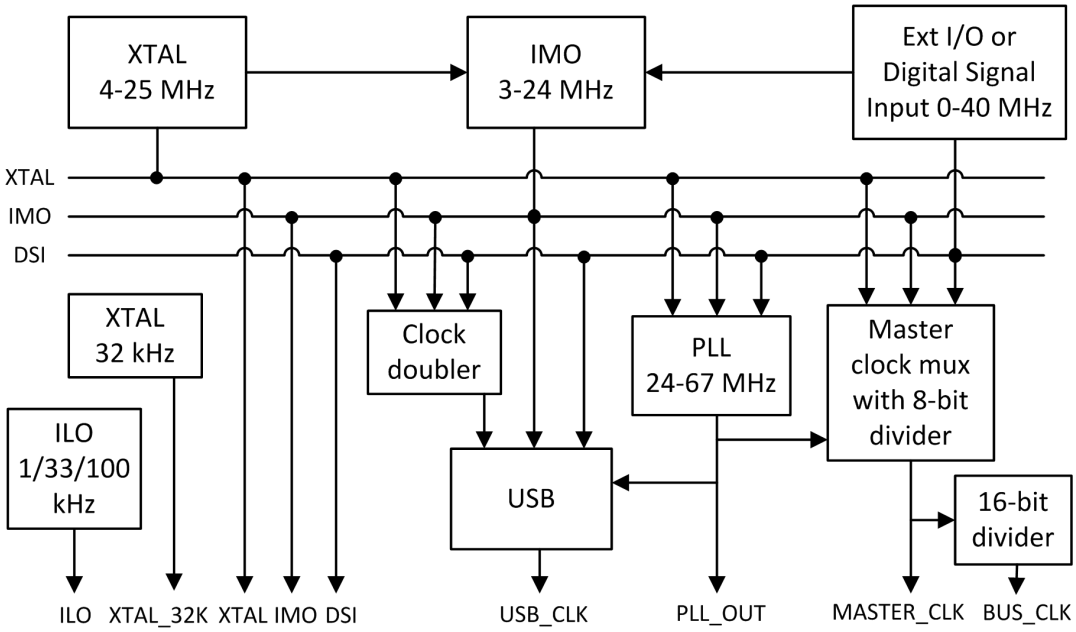
Name	Value
VDDA (V)	5.0
VDDD (V)	5.0
VDDIO0 (V)	5.0
VDDIO1 (V)	5.0
VDDIO2 (V)	5.0
VDDIO3 (V)	1.8
Variable VDDA	True
Temperature Range	-40C - 85/125C

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
USB_CLK	DIGITAL	IMO	48 MHz	48 MHz	±0.25	False	True
IMO	DIGITAL		24 MHz	24 MHz	±0.25	True	True
MASTER_CLK	DIGITAL	PLL_OUT	? MHz	24 MHz	±0.25	True	True
BUS_CLK	DIGITAL	MASTER_CLK	? MHz	24 MHz	±0.25	True	True
PLL_OUT	DIGITAL	IMO	24 MHz	24 MHz	±0.25	True	True
ILO	DIGITAL		? MHz	100 kHz	-55,+100	True	True
XTAL 32kHz	DIGITAL		32.768 kHz	? MHz	±0	False	False
Digital Signal	DIGITAL		? MHz	? MHz	±0	False	False
XTAL	DIGITAL		24 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

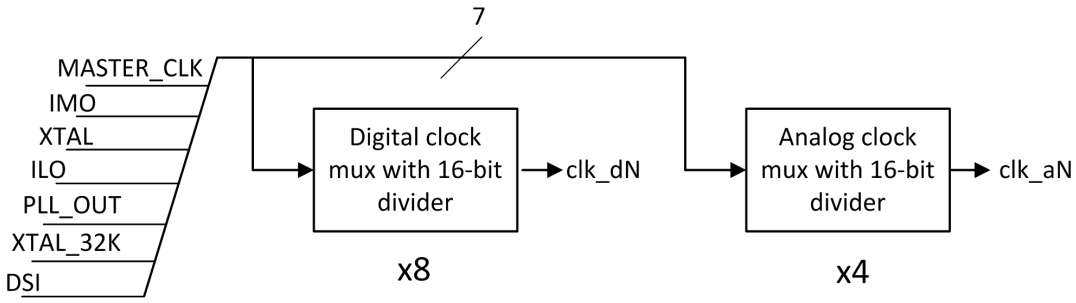


Table 10 lists the design wide clocks used in this design.

Table 10. Design Wide Clocks

Name	Domain	Source	Desired Freq	Nominal Freq	Accuracy (%)	Start at Reset	Enabled
ScBoostClk	ANALOG	IMO	10 MHz	12 MHz	±0.25	False	True

Table 11 lists the local clocks used in this design.

Table 11. Local Clocks

Name	Domain	Source	Desired Freq	Nominal Freq	Accuracy (%)	Start at Reset	Enabled
CAN_Clock	DIGITAL	BUS_CLK	? MHz	24 MHz	±0.25	True	True
Clock_1	DIGITAL	MASTER_CLK	100 kHz	100 kHz	±0.25	True	True
Clock_2	DIGITAL	MASTER_CLK	10 kHz	10 kHz	±0.25	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
  - CyPLL API routines
  - CyIMO API routines
  - 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 12. Interrupts

Name	Priority	Vector
CAN_isr	7	16
USBUART_1_arb_int	7	22
USBUART_1_bus_reset	7	23
USBUART_1_dp_int	7	12
USBUART_1_ep_0	7	24
USBUART_1_ep_1	7	0
USBUART_1_ep_2	7	1
USBUART_1_ep_3	7	2
USBUART_1_sof_int	7	21

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 13 lists the Flash protection settings for your design.

Table 13. 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:

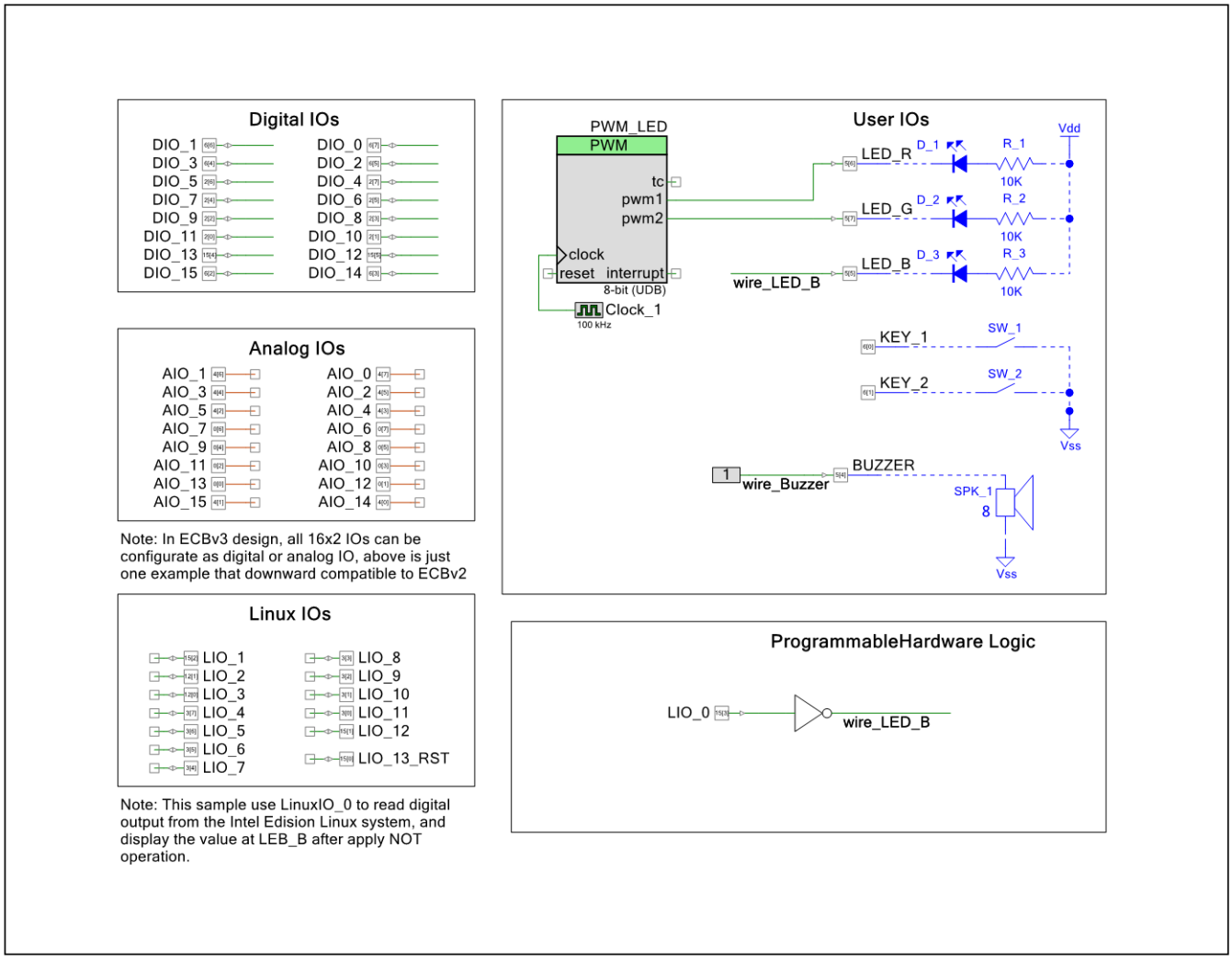
- Flash Protection chapter in the PSoC 5LP Technical Reference Manual
- Flash and EEPROM chapter in the System Reference Guide
  - CyWrite API routines
  - CyFlash API routines

7 Design Contents

This design's schematic content consists of the following 2 schematic sheets:

7.1 Schematic Sheet: IO Ports

Figure 5. Schematic Sheet: IO Ports

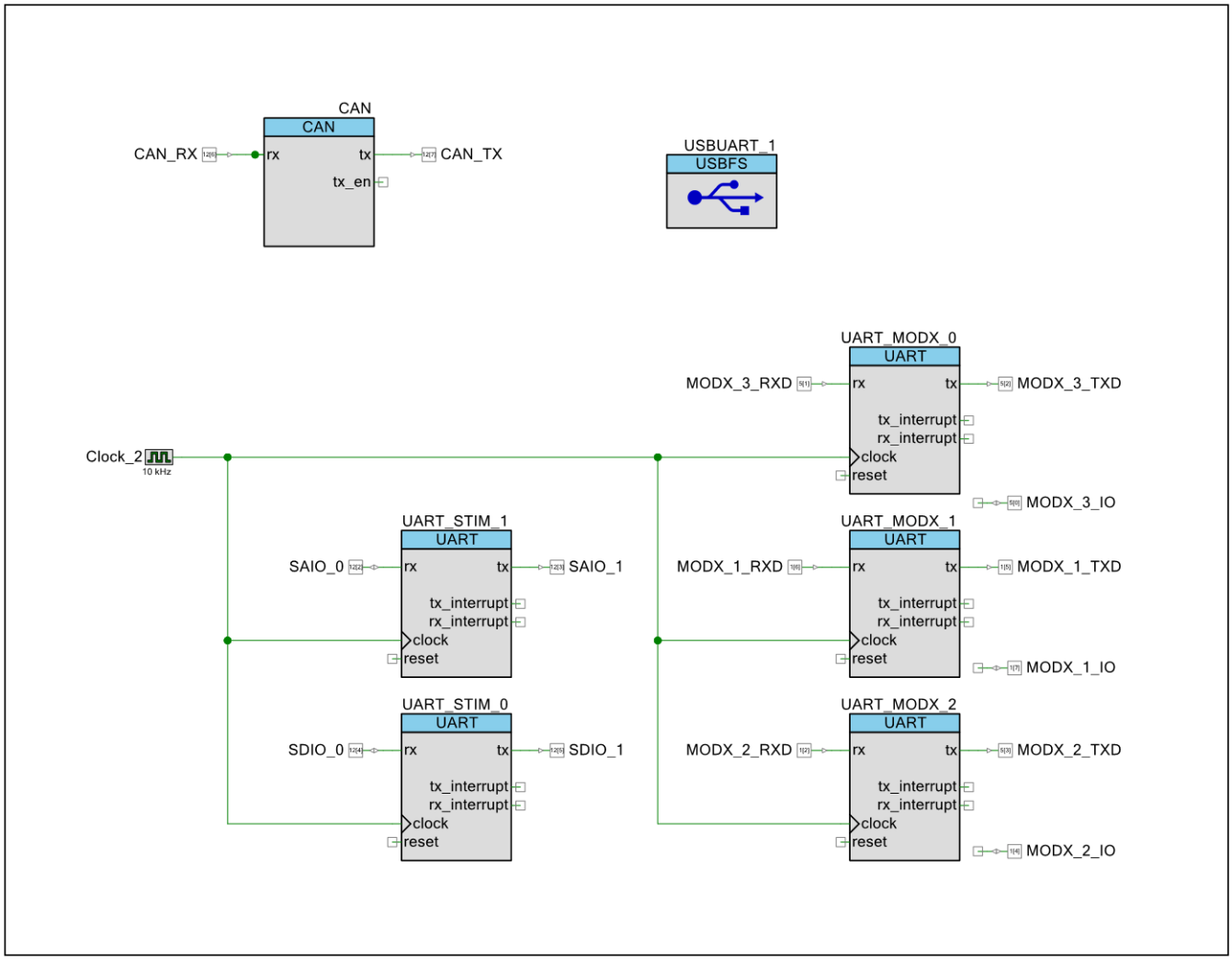


This schematic sheet contains the following component instances:

- Instance PWM_LED (type: PWM_v3_30)

7.2 Schematic Sheet: Comm

Figure 6. Schematic Sheet: Comm



This schematic sheet contains the following component instances:

- Instance CAN (type: CAN_v3_0)
- Instance UART_MODX_0 (type: UART_v2_50)
- Instance UART_MODX_1 (type: UART_v2_50)
- Instance UART_MODX_2 (type: UART_v2_50)
- Instance UART_STIM_0 (type: UART_v2_50)
- Instance UART_STIM_1 (type: UART_v2_50)
- Instance USBUART_1 (type: USBFS_v3_0)

8 Components

8.1 Component type: CAN [v3.0]

8.1.1 Instance CAN

Description: Controller Area Network (ISO-11898-1)
Instance type: CAN [v3.0]
Datasheet: online component datasheet for CAN

Table 14. Component Parameters for CAN

Parameter Name	Value	Description
AckError	false	Message acknowledge error detected interrupt
AckErrorUseHelper	true	Use ISR helper for Message acknowledge error detected interrupt
AdvancedInterruptConfig	true	Advanced interrupt configuration
Arbiter	0	Transmit buffer arbitration
ArbLost	false	Arbitration lost detected interrupt
ArbLostUseHelper	true	Use ISR helper for Arbitration lost detected interrupt
BaudRate	125	The desired baud rate
BitError	false	Bit error detected interrupt
BitErrorUseHelper	true	Use ISR helper for Bit error detected interrupt
Bitrate	11	Bit rate prescaler (BRP)
BussOff	false	Bus off state interrupt
BussOffUseHelper	true	Use ISR helper for Bus off state interrupt
ClkFrequency	24	The system clock frequency equal to BUS_CLK (PSoC 3/5LP) or SYSCLK (PSoC 4)
ConnectExtInterruptLine	false	Use External interrupt line as the third output
ConnectTxEn	true	Use external transceiver enable signal as the second output
CrcError	false	CRC error detected interrupt
CrcErrorUseHelper	true	Use ISR Helper for CRC error detected interrupt
EdgeMode	0	CAN bus synchronization logic
FormError	false	Message format error detected interrupt
FormErrorUseHelper	true	Use ISR helper for Message format error detected interrupt
FullCustomIntISR	true	Enable the internal ISR with fully custom code
IntEnable	true	Global Interrupt enable flag
IntISRDisable	false	Disable/bypass the internal ISR component
Overload	false	Overload frame received interrupt

Parameter Name	Value	Description
OverloadUseHelper	true	Use ISR helper for Overload frame received interrupt
Reset	0	Bus-off restart
RTRAutomaticReply	false	RTR automatic reply sent interrupt
RTRAutomaticReplyUseHelper	true	Use ISR helper for RTR automatic reply sent interrupt
RxMsg	true	Message received interrupt
RxMsgLost	false	Receive buffer full interrupt
RxMsgLostUseHelper	true	Use ISR helper for Receive buffer full interrupt
RxMsgUseHelper	true	Use ISR helper for Message received interrupt
Sjw	4	Synchronization Jump Width
Sm	0	CAN bus Bit sampling
SSTError	false	Single shot transmission failure interrupt
SSTErrorUseHelper	true	Use ISR helper for Single shot transmission failure interrupt
StuckAtZero	false	Stuck at zero interrupt
StuckAtZeroUseHelper	true	Use ISR helper for Stuck at zero interrupt
StuffError	false	Bit stuffing error detected interrupt
StuffErrorUseHelper	true	Use ISR helper for Bit stuffing error detected interrupt
SwapDataByteEndianness	false	The byte position of the CAN receive and transmit data field endianness
Tseg1	10	The length of time segment1
Tseg2	5	The length of time segment2
TxMsg	false	Message transmitted interrupt
TxMsgUseHelper	true	Use ISR helper for Message transmitted interrupt

8.2 Component type: PWM [v3.30]

8.2.1 Instance PWM_LED

Description: 8 or 16-bit Pulse Width Modulator
Instance type: PWM [v3.30]
Datasheet: online component datasheet for PWM

Table 15. Component Parameters for PWM_LED

Parameter Name	Value	Description
CaptureMode	None	Defines the functionality of the capture Input. The parameter determines which signal on the capture input is required to capture the current count value to the FIFO.
CompareStatusEdgeSense	true	Enables edge sense detection on compare outputs for use in edge sensitive interrupts

Parameter Name	Value	Description
CompareType1	Greater or Equal	Sets the compare value comparison type setting for the compare 1 output
CompareType2	Greater or Equal	Sets the compare value comparison type setting for the compare 2 output
CompareValue1	0	Compares Output 1 to value
CompareValue2	0	Compares Output 2 to value
DeadBand	Disabled	Defines whether dead band outputs are desired or not.
DeadTime	1	Defines the number of required dead band clock cycles
DitherOffset	0.00	Allows the user to implement dither to get more bits out of a 8 or 16 bit PWM.
EnableMode	Software Only	Specifies the method of enabling the PWM. This can be either hardware or software.
FixedFunction	false	Determines whether the fixed function counter timer is used or the UDB implementation is used.
InterruptOnCMP1	false	Enables the interrupt on compare1 true event
InterruptOnCMP2	false	Enables the interrupt on compare2 true event
InterruptOnKill	false	Enables the interrupt on a kill event
InterruptOnTC	false	Enables the interrupt on terminal count event
KillMode	Disabled	Parameter to select the kill mode for build time.
MinimumKillTime	1	Sets the minimum number of clock cycles that a kill must be active on the outputs when KillMode is set to Minimum Kill Time mode
Period	255	Defines the PWM period value
PWMMode	Two Outputs	Defines the overall mode of the PWM
Resolution	8	Defines the bit width of the PWM (8 or 16 bits)
RunMode	Continuous	Defines the run mode options to be either continuous or one shot
TriggerMode	None	Determines the mode of starting the PWM, i.e. triggering the PWM counter to start
UseInterrupt	true	Enables the placement and usage of the status register

8.3 Component type: UART [v2.50]

8.3.1 Instance UART_MODX_0

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

Table 16. Component Parameters for UART_MODX_0

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	57600	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	false	Enables the external TX enable signal output.
InternalClock	false	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
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

Parameter Name	Value	Description
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.

8.3.2 Instance UART_MODX_1

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

Table 17. Component Parameters for UART_MODX_1

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	57600	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.

Parameter Name	Value	Description
FlowControl	None	Enable the flow control signals.
HalfDuplexEn	false	Enables half duplex mode on the RX Half of the UART module.
HwTXEnSignal	false	Enables the external TX enable signal output.
InternalClock	false	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
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

Parameter Name	Value	Description
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.

8.3.3 Instance UART_MODX_2

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

Table 18. Component Parameters for UART_MODX_2

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	57600	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	false	Enables the external TX enable signal output.
InternalClock	false	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.

Parameter Name	Value	Description
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
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.

8.3.4 Instance UART_STIM_0

Description: Universal Asynchronous Receiver Transmitter
Instance type: UART [v2.50]

Datasheet: [online component datasheet for UART]

Table 19. Component Parameters for UART_STIM_0

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	57600	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	false	Enables the external TX enable signal output.
InternalClock	false	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
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

Parameter Name	Value	Description
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.

8.3.5 Instance UART_STIM_1

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

Table 20. Component Parameters for UART_STIM_1

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	57600	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.

Parameter Name	Value	Description
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	false	Enables the external TX enable signal output.
InternalClock	false	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
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

Parameter Name	Value	Description
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.

8.4 Component type: USBFS [v3.0]

8.4.1 Instance USBUART_1

Description: USB 2.0 Full Speed Device Framework
Instance type: USBFS [v3.0]
Datasheet: online component datasheet for USBFS

Table 21. Component Parameters for USBUART_1

Parameter Name	Value	Description
EnableBatteryChargDetect	false	This parameter allows to detect a charging supported USB host port using the API function USBFS_DetectPortType().
EnableCDCApi	true	Enables additional high level API's that allow the CDC device to be used similar to a UART device.
EnableMidiApi	true	Enables additional high level MIDI API's.
endpointMA	MA_Static	Endpoint memory allocation
endpointMM	EP_Manual	Endpoint memory management
epDMAautoOptimization	false	This parameter enables resource optimization for DMA with Automatic Memory Management mode. Set this parameter value to true only when a single IN endpoint is present in the device. Enabling this parameter in a multi IN endpoint device configuration causes undesired effects.
extern_cls	false	This parameter allows for user or other component to implement his own handler for Class requests. USBFS_-DispatchClassRqst() function should be implemented if this parameter enabled.
extern_vbus	false	This parameter enables external VBUSDET input.

Parameter Name	Value	Description
extern_vnd	false	This parameter allows for user or other component to implement his own handler for Vendor specific requests. USBFS_HandleVendorRqst() function should be implemented if this parameter enabled.
extJackCount	0	Max number of External MIDI IN Jack or OUT Jack descriptors
Gen16bitEpAccessApi	true	This parameter defines whether to generate APIs for the 16-bits endpoint access.
HandleMscRequests	true	This parameter is used to enable handling MSC requests and generate MSC APIs.
isrGroupArbiter	High	This parameter defines the interrupt group of the Arbiter Interrupt.
isrGroupBusReset	Low	This parameter defines the interrupt group of the Bus Reset Interrupt.
isrGroupEp0	Medium	This parameter defines the interrupt group of the Control Endpoint Interrupt (EP0).
isrGroupEp1	Medium	This parameter defines the interrupt group of the Data Endpoint 1 Interrupt.
isrGroupEp2	Medium	This parameter defines the interrupt group of the Data Endpoint 2 Interrupt.
isrGroupEp3	Medium	This parameter defines the interrupt group of the Data Endpoint 3 Interrupt.
isrGroupEp4	Medium	This parameter defines the interrupt group of the Data Endpoint 4 Interrupt.
isrGroupEp5	Medium	This parameter defines the interrupt group of the Data Endpoint 5 Interrupt.
isrGroupEp6	Medium	This parameter defines the interrupt group of the Data Endpoint 6 Interrupt.
isrGroupEp7	Medium	This parameter defines the interrupt group of the Data Endpoint 7 Interrupt.
isrGroupEp8	Medium	This parameter defines the interrupt group of the Data Endpoint 8 Interrupt.
isrGroupLpm	High	This parameter defines the interrupt group of the LPM Interrupt.
isrGroupSof	Low	This parameter defines the interrupt group of the Start of Frame Interrupt.
max_interfaces_num	2	Defines maximum interfaces number

Parameter Name	Value	Description
Mode	false	Specifies whether the implementation will create API for interfacing to UART component(s) for a corresponding set of external MIDI connections.
mon_vbus	false	The mon_vbus parameter adds a single VBUS monitor pin to the design. This pin must be connected to VBUS and must be assigned in the pin editor.
MscDescriptors		Mass Storage Class Descriptors
MscLogicalUnitsNum	1	This parameter allows to specify the number of logical units that should be supported by the Mass Storage device.
out_sof	false	The out_sof parameter enables Start-of-Frame output.
Pid	F232	Product ID
powerpad_vbus	false	This parameter enables VBUS power pad
ProdactName		This string is displayed by the Operating System when it is installing the mass storage device as the Product Name.
ProdactRevision		This string is displayed by the Operating System when it is installing the mass storage device as the Product Revision.
rm_lpm_int	true	Removes LPM ISR
VendorName		This string is displayed by the Operating System when it is installing the mass storage device as the Vendor Name.
Vid	04B4	Vendor ID

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