PSoC® Creator™
Project Datasheet for LTC

Creation Time: 06/02/2014 11:45:53
User: Tommy-Pc\Tommy
Project: LTC

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

Copyrights

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

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.

The information in this document is subject to change without notice and should not be construed as a commitment by Cypress. While reasonable precautions have been taken, Cypress assumes no responsibility for any errors that may appear in this document. No part of this document may be copied, or reproduced for commercial use, in any form or by any means without the prior written consent of Cypress.

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. 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 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 3 is a family of 8-bit devices with the following characteristics:

- An 8-bit single cycle pipelined 8051 processor, running up to 67 MHz, with a nested vectored interrupt controller (NVIC) and a high-performance DMA controller. The single cycle 8051 CPU runs ten times faster than a standard 8051 processor.
- Digital system that includes configurable Universal Digital Blocks (UDBs) and specific function peripherals, such as USB, CAN and I2C
- Analog subsystem that includes configurable switched (SC) and continuous time (CT) blocks, up to 20-bit Delta Sigma converters (ADC), SAR ADCs, 8-bit DACs that can be configured for 12-bit operation, op amps, comparators, PGAs, 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 CY8C38 family member PSoC 3 device. For details on all the systems listed above, please refer to the PSoC 3 Technical Reference Manual .

Figure 1. CY8C38 Device Family Block Diagram

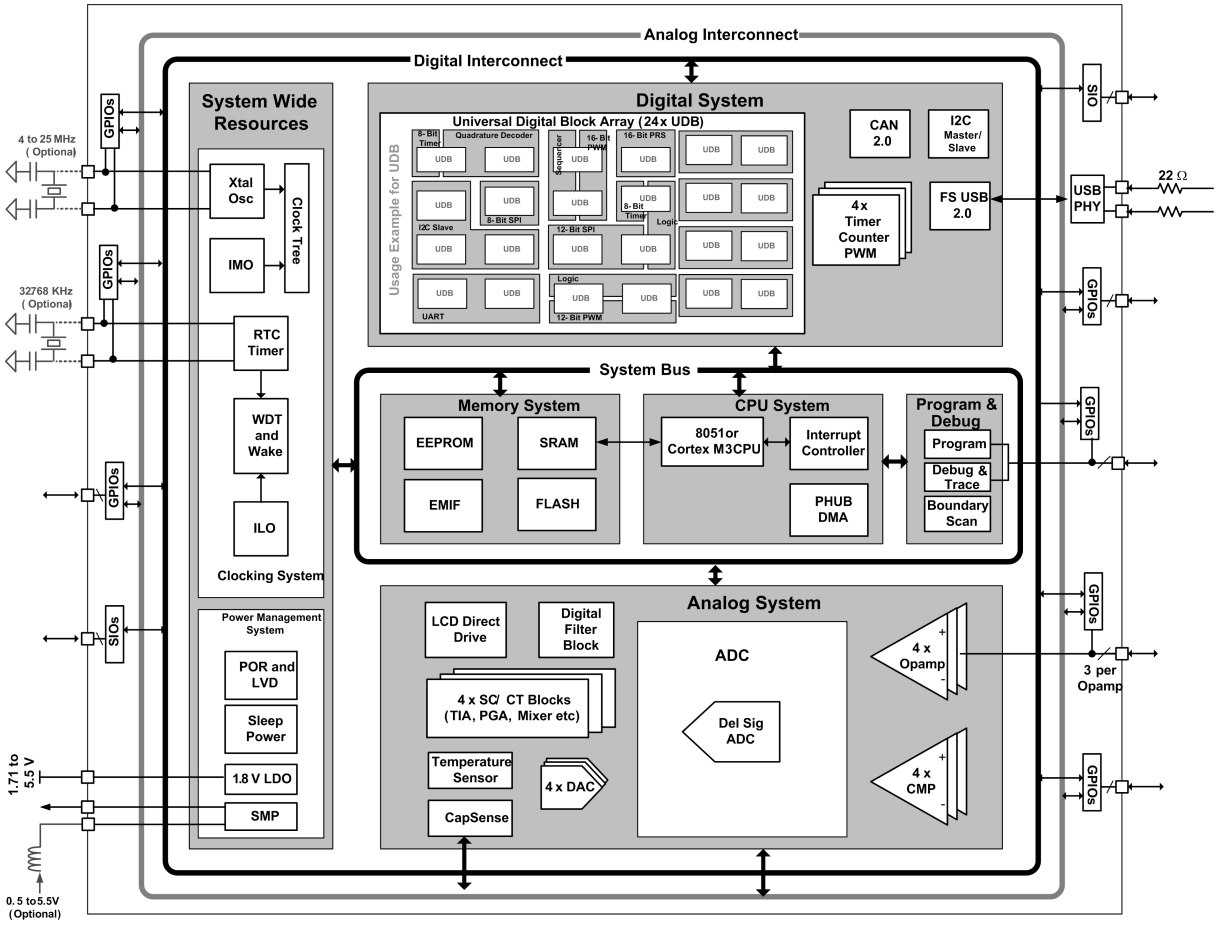


Table 1 lists the key characteristics of this device.

Table 1. Device Characteristics

Name	Value
Architecture	PSoC 3
Family	CY8C38
CPU speed (MHz)	67
Flash size (kBytes)	64
SRAM size (kBytes)	8
EEPROM size (Bytes)	2048
Trace Buffer (kBytes)	4
Vdd range (V)	1.7 to 5.5
Automotive qualified	No (Industrial Grade Only)
Temp range (Celcius)	-40 to 85
JTAG ID	0x1E028069

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

Table 2 lists the device resources that this design uses:

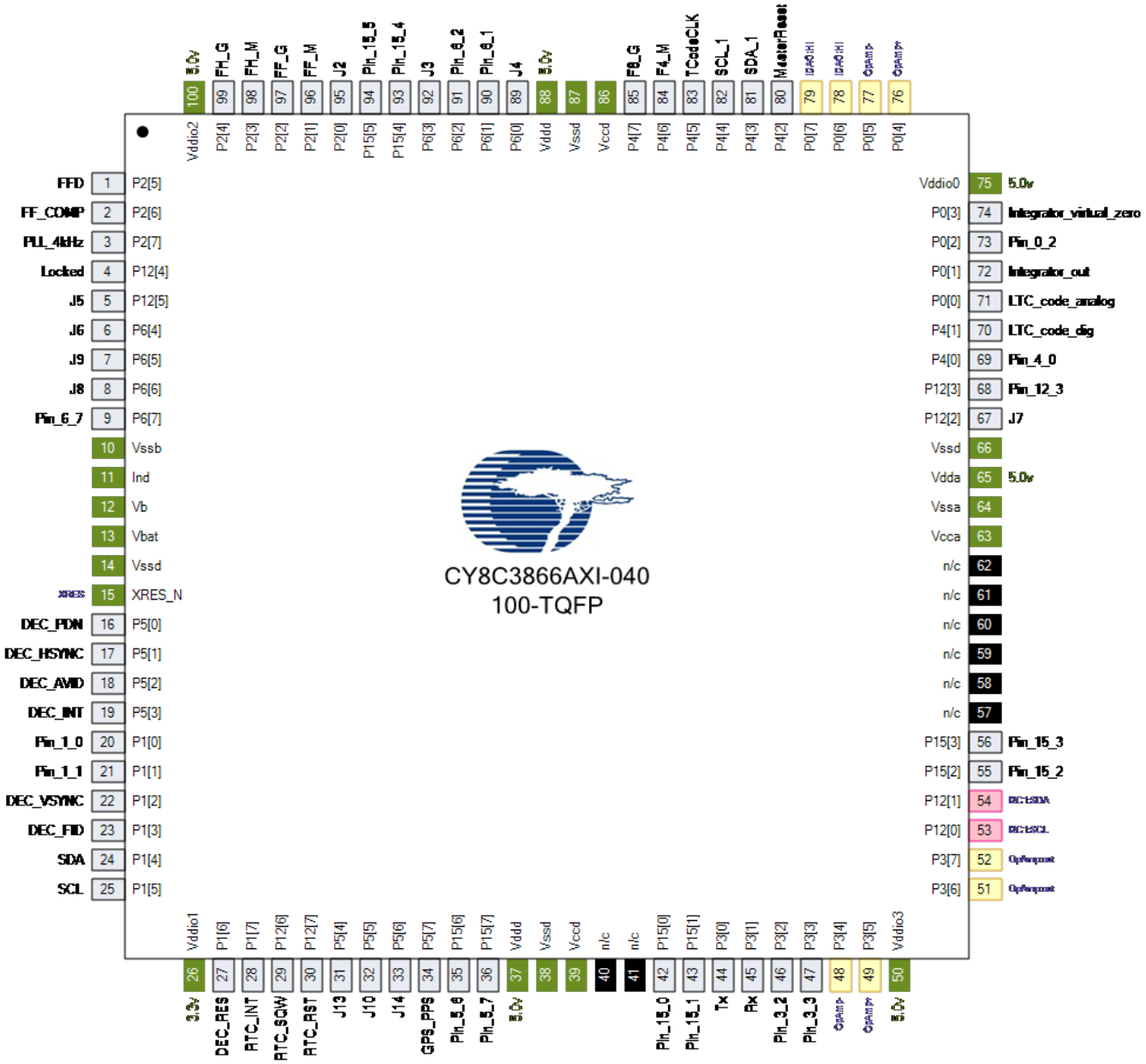
Table 2. Device Resources

Name	Resources in Use	Total Resources Available
Digital clock dividers	7 (87.5%)	8
Analog clock dividers	0 (0.0%)	4
Pins	62 (86.1%)	72
UDB Macrocells	125 (65.1%)	192
UDB Unique Pterms	238 (62.0%)	384
UDB Datapath Cells	16 (66.7%)	24
UDB Status Cells	16 (66.7%)	24
UDB Control Cells	9 (37.5%)	24
DMA Channels	0 (0.0%)	24
Interrupts	12 (37.5%)	32
DSM Fixed Blocks	0 (0.0%)	1
VIDAC Fixed Blocks	0 (0.0%)	4
SC Fixed Blocks	0 (0.0%)	4
Comparator Fixed Blocks	0 (0.0%)	4
Opamp Fixed Blocks	2 (50.0%)	4
CapSense Buffers	0 (0.0%)	2
CAN Fixed Blocks	0 (0.0%)	1
Decimator Fixed Blocks	0 (0.0%)	1
I2C Fixed Blocks	1 (100.0%)	1
Timer Fixed Blocks	1 (25.0%)	4
DFB Fixed Blocks	0 (0.0%)	1
USB Fixed Blocks	0 (0.0%)	1
LCD Fixed Blocks	0 (0.0%)	1
EMIF Fixed Blocks	0 (0.0%)	1
LPF Fixed Blocks	0 (0.0%)	2

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]	FFD	Dgtl Out	Strong drive	HiZ Analog Unb
2	P2[6]	FF_COMP	Dgtl Out	Strong drive	HiZ Analog Unb
3	P2[7]	PLL_4kHz	Dgtl In	Res pull up	HiZ Analog Unb
4	P12[4]	Locked	Dgtl In	HiZ digital	HiZ Analog Unb
5	P12[5]	J5		Strong drive	HiZ Analog Unb
6	P6[4]	J6		Strong drive	HiZ Analog Unb
7	P6[5]	J9		Strong drive	HiZ Analog Unb
8	P6[6]	J8		Strong drive	HiZ Analog Unb
9	P6[7]	Pin_6_7		Strong drive	HiZ Analog Unb
10	Vssb	Vssb	Power		
11	Ind	Power			
12	Vb	Vb	Power		
13	Vbat	Vbat	Power		
14	Vssd	Vssd	Power		
15	XRES_N	XRES_N	Power		
16	P5[0]	DEC_PDN		Strong drive	HiZ Analog Unb
17	P5[1]	DEC_HSYNC		Res pull down	HiZ Analog Unb
18	P5[2]	DEC_AVID		Res pull down	HiZ Analog Unb
19	P5[3]	DEC_INT	Dgtl In	Res pull up	HiZ Analog Unb
20	P1[0]	Pin_1_0		Strong drive	HiZ Analog Unb
21	P1[1]	Pin_1_1		Strong drive	HiZ Analog Unb
22	P1[2]	DEC_VSYNC		Res pull down	HiZ Analog Unb
23	P1[3]	DEC_FID		Res pull down	HiZ Analog Unb
24	P1[4]	SDA	Dgtl I/O	Res pull up	HiZ Analog Unb
25	P1[5]	SCL	Dgtl I/O	Res pull up	HiZ Analog Unb
26	Vio1	Vio1	Power		
27	P1[6]	DEC_RES		Strong drive	HiZ Analog Unb
28	P1[7]	RTC_INT		Res pull up	HiZ Analog Unb
29	P12[6]	RTC_SQW		Res pull up	HiZ Analog Unb
30	P12[7]	RTC_RST		Res pull up	HiZ Analog Unb
31	P5[4]	J13		Strong drive	HiZ Analog Unb
32	P5[5]	J10		Res pull up	HiZ Analog Unb
33	P5[6]	J14		Strong drive	HiZ Analog Unb
34	P5[7]	GPS_PPS	Dgtl In	Res pull up	HiZ Analog Unb
35	P15[6]	Pin_5_6		Strong drive	HiZ Analog Unb
36	P15[7]	Pin_5_7		Strong drive	HiZ Analog Unb
37	Vddd	Vddd	Power		
38	Vssd	Vssd	Power		
39	Vccd	Vccd	Power		
42	P15[0]	Pin_15_0		Strong drive	HiZ Analog Unb
43	P15[1]	Pin_15_1		Strong drive	HiZ Analog Unb
44	P3[0]	Tx	Dgtl Out	Strong drive	HiZ Analog Unb

Pin	Port	Name	Type	Drive Mode	Reset State
45	P3[1]	Rx	Dgtl In	Res pull up	HiZ Analog Unb
46	P3[2]	Pin_3_2		Strong drive	HiZ Analog Unb
47	P3[3]	Pin_3_3		Strong drive	HiZ Analog Unb
48	P3[4]	GPIO [unused]			HiZ Analog Unb
49	P3[5]	GPIO [unused]			HiZ Analog Unb
50	Vio3	Vio3	Power		
51	P3[6]	GPIO [unused]			HiZ Analog Unb
52	P3[7]	GPIO [unused]			HiZ Analog Unb
53	P12[0]	SIO [unused]			HiZ Analog Unb
54	P12[1]	SIO [unused]			HiZ Analog Unb
55	P15[2]	Pin_15_2		Strong drive	HiZ Analog Unb
56	P15[3]	Pin_15_3		Strong drive	HiZ Analog Unb
63	Vcca	Vcca	Power		
64	Vssa	Vssa	Power		
65	Vdda	Vdda	Power		
66	Vssd	Vssd	Power		
67	P12[2]	J7		Strong drive	HiZ Analog Unb
68	P12[3]	Pin_12_3	Dgtl Out	Strong drive	HiZ Analog Unb
69	P4[0]	Pin_4_0		Strong drive	HiZ Analog Unb
70	P4[1]	LTC_code_dig	Dgtl Out	Strong drive	HiZ Analog Unb
71	P0[0]	LTC_code_analog	Analog	HiZ analog	HiZ Analog Unb
72	P0[1]	Integrator_out	Analog	HiZ analog	HiZ Analog Unb
73	P0[2]	Pin_0_2		Strong drive	HiZ Analog Unb
74	P0[3]	Integrator_virtual_zero	Analog	HiZ analog	HiZ Analog Unb
75	Vio0	Vio0	Power		
76	P0[4]	GPIO [unused]			HiZ Analog Unb
77	P0[5]	GPIO [unused]			HiZ Analog Unb
78	P0[6]	GPIO [unused]			HiZ Analog Unb
79	P0[7]	GPIO [unused]			HiZ Analog Unb
80	P4[2]	MasterReset		Res pull up	HiZ Analog Unb
81	P4[3]	SDA_1	Dgtl I/O	OD, DL	HiZ Analog Unb
82	P4[4]	SCL_1	Dgtl I/O	OD, DL	HiZ Analog Unb
83	P4[5]	TCodeCLK	Dgtl In	HiZ digital	HiZ Analog Unb
84	P4[6]	F4_M		Res pull up	HiZ Analog Unb
85	P4[7]	F8_G		Res pull up	HiZ Analog Unb
86	Vccd	Vccd	Power		
87	Vssd	Vssd	Power		
88	Vddd	Vddd	Power		
89	P6[0]	J4		Strong drive	HiZ Analog Unb
90	P6[1]	Pin_6_1		Strong drive	HiZ Analog Unb
91	P6[2]	Pin_6_2		Strong drive	HiZ Analog Unb
92	P6[3]	J3		Strong drive	HiZ Analog Unb
93	P15[4]	Pin_15_4		Strong drive	HiZ Analog Unb
94	P15[5]	Pin_15_5		Strong drive	HiZ Analog Unb
95	P2[0]	J2		Strong drive	HiZ Analog Unb
96	P2[1]	FF_M	Dgtl In	Res pull up	HiZ Analog Unb
97	P2[2]	FF_G	Dgtl In	Res pull up	HiZ Analog Unb
98	P2[3]	FH_M		Res pull up	HiZ Analog Unb
99	P2[4]	FH_G		Res pull up	HiZ Analog Unb
100	Vio2	Vio2	Power		

Abbreviations used in Table 3 have the following meanings:

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

- Dgtl In = Digital Input
- Res pull up = Resistive pull up
- HiZ digital = High impedance digital
- Res pull down = Resistive pull down
- Dgtl I/O = Digital In/Out
- HiZ analog = High impedance analog
- OD, DL = Open drain, drives low

2.2 Software Pins

Table 4 contains information about the software pins on this device in alphabetical order. (Only software-accessible pins are shown.)

Table 4. Software Pins

Name	Port	Type	Reset State
DEC_AVID	P5[2]		HiZ Analog Unb
DEC_FID	P1[3]		HiZ Analog Unb
DEC_HSYNC	P5[1]		HiZ Analog Unb
DEC_INT	P5[3]	Dgtl In	HiZ Analog Unb
DEC_PDN	P5[0]		HiZ Analog Unb
DEC_RES	P1[6]		HiZ Analog Unb
DEC_VSYNC	P1[2]		HiZ Analog Unb
F4_M	P4[6]		HiZ Analog Unb
F8_G	P4[7]		HiZ Analog Unb
FF_COMP	P2[6]	Dgtl Out	HiZ Analog Unb
FF_G	P2[2]	Dgtl In	HiZ Analog Unb
FF_M	P2[1]	Dgtl In	HiZ Analog Unb
FFD	P2[5]	Dgtl Out	HiZ Analog Unb
FH_G	P2[4]		HiZ Analog Unb
FH_M	P2[3]		HiZ Analog Unb
GPS_PPS	P5[7]	Dgtl In	HiZ Analog Unb
Integrator_out	P0[1]	Analog	HiZ Analog Unb
Integrator_virtual_zero	P0[3]	Analog	HiZ Analog Unb
J10	P5[5]		HiZ Analog Unb
J13	P5[4]		HiZ Analog Unb
J14	P5[6]		HiZ Analog Unb
J2	P2[0]		HiZ Analog Unb
J3	P6[3]		HiZ Analog Unb
J4	P6[0]		HiZ Analog Unb
J5	P12[5]		HiZ Analog Unb
J6	P6[4]		HiZ Analog Unb
J7	P12[2]		HiZ Analog Unb
J8	P6[6]		HiZ Analog Unb
J9	P6[5]		HiZ Analog Unb
Locked	P12[4]	Dgtl In	HiZ Analog Unb
LTC_code_analog	P0[0]	Analog	HiZ Analog Unb
LTC_code_dig	P4[1]	Dgtl Out	HiZ Analog Unb
MasterReset	P4[2]		HiZ Analog Unb
Pin_0_2	P0[2]		HiZ Analog Unb
Pin_1_0	P1[0]		HiZ Analog Unb
Pin_1_1	P1[1]		HiZ Analog Unb
Pin_12_3	P12[3]	Dgtl Out	HiZ Analog Unb
Pin_15_0	P15[0]		HiZ Analog Unb
Pin_15_1	P15[1]		HiZ Analog Unb
Pin_15_2	P15[2]		HiZ Analog Unb
Pin_15_3	P15[3]		HiZ Analog Unb
Pin_15_4	P15[4]		HiZ Analog Unb
Pin_15_5	P15[5]		HiZ Analog Unb
Pin_3_2	P3[2]		HiZ Analog Unb
Pin_3_3	P3[3]		HiZ Analog Unb

Name	Port	Type	Reset State
Pin_4_0	P4[0]		HiZ Analog Unb
Pin_5_6	P15[6]		HiZ Analog Unb
Pin_5_7	P15[7]		HiZ Analog Unb
Pin_6_1	P6[1]		HiZ Analog Unb
Pin_6_2	P6[2]		HiZ Analog Unb
Pin_6_7	P6[7]		HiZ Analog Unb
PLL_4kHz	P2[7]	Dgtl In	HiZ Analog Unb
Power	Ind		
RTC_INT	P1[7]		HiZ Analog Unb
RTC_RST	P12[7]		HiZ Analog Unb
RTC_SQW	P12[6]		HiZ Analog Unb
Rx	P3[1]	Dgtl In	HiZ Analog Unb
SCL	P1[5]	Dgtl I/O	HiZ Analog Unb
SCL_1	P4[4]	Dgtl I/O	HiZ Analog Unb
SDA	P1[4]	Dgtl I/O	HiZ Analog Unb
SDA_1	P4[3]	Dgtl I/O	HiZ Analog Unb
TCodeCLK	P4[5]	Dgtl In	HiZ Analog Unb
Tx	P3[0]	Dgtl Out	HiZ Analog Unb

Abbreviations used in Table 4 have the following meanings:

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

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 5. 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
Clear SRAM During Startup	True
Unused Bonded IO	Allow but warn

3.2 System Debug Settings

Table 6. System Debug Settings

Name	Value
Debug Select	GPIO
Enable Device Protection	False
Use Optional XRES	False

3.3 System Operating Conditions

Table 7. System Operating Conditions

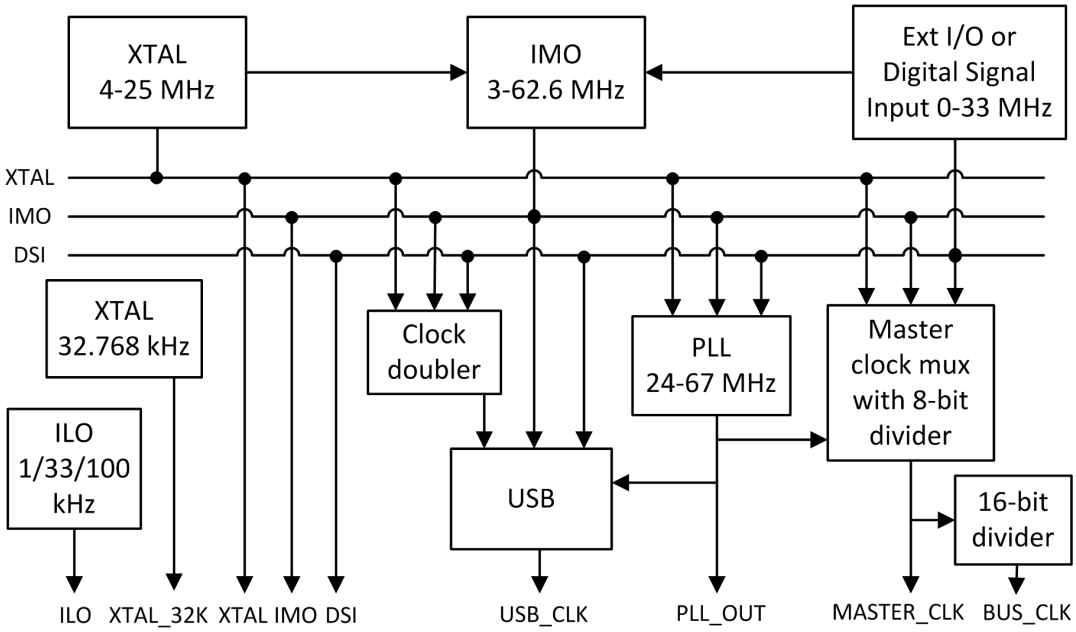
Name	Value
Vddd (V)	5.0
Vdda (V)	5.0
Variable Vdda	False
Vddio0 (V)	5.0
Vddio1 (V)	3.3
Vddio2 (V)	5.0
Vddio3 (V)	5.0
Temperature Range	0C - 85/125C

4 Clocks

The clock system includes these clock resources:

- Four internal clock sources increase system integration:
  - 3 to 62.6 MHz Internal Main Oscillator (IMO) ±1% at 3 MHz
  - 1 kHz, 33 kHz, 100 kHz Internal Low Speed Oscillator (ILO) outputs
  - 12 to 67 MHz clock doubler output, sourced from IMO, MHz External Crystal Oscillator (MHzECO), and Digital System Interconnect (DSI)
  - 24 to 67 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 8 lists the system clocks used in this design.

Table 8. System Clocks

Name	Domain	Source	Desired Freq (MHz)	Nominal Freq (MHz)	Accuracy (%)	Start at Reset	Enabled
MASTER_CLK	DIGITAL	PLL_OUT	0	42	±0	True	True
Digital Signal	DIGITAL	OCXO_27MHz	27	27	±0	False	True
XTAL 32kHz	DIGITAL		0.0328	0	±0	False	False
XTAL	DIGITAL		25	0	±0	False	False
ILO	DIGITAL		0	0.001	-50,+100	True	True
PLL_OUT	DIGITAL	Digital Signal	42	42	±0	True	True
IMO	DIGITAL		3	3	±1	True	True
BUS_CLK	DIGITAL	MASTER_CLK	0	42	±0	True	True
USB_CLK	DIGITAL	IMO	48	0	±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

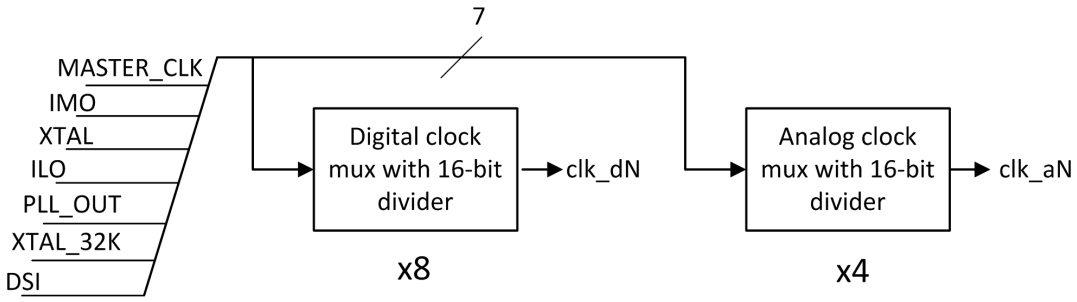


Table 9 lists the local clocks used in this design.

Table 9. Local Clocks

Name	Domain	Source	Desired Freq (MHz)	Nominal Freq (MHz)	Accuracy (%)	Start at Reset	Enabled
Clock_3	DIGITAL	BUS_CLK	0	42	±0	True	True
I2C_2_IntClock	DIGITAL	MASTER_CLK	1.6	1.6154	±0	True	True
Clock_6	DIGITAL	BUS_CLK	0	42	±0	True	True
Clock_5	DIGITAL	ILO	0	0	-50,+100	True	True
UART_1_In-tClock	DIGITAL	MASTER_CLK	0.9216	0.913	±0	True	True
Clock_2	DIGITAL	BUS_CLK	0	42	±0	True	True
Clock_4	DIGITAL	MASTER_CLK	42	42	±0	True	True
timer_clock	DIGITAL	IMO	0.001	0.001	±1	True	True
Clock_7	DIGITAL	BUS_CLK	0	42	±0	True	True
Clock_1	DIGITAL	MASTER_CLK	42	42	±0	True	True

Name	Domain	Source	Desired Freq (MHz)	Nominal Freq (MHz)	Accuracy (%)	Start at Reset	Enabled
Clock_11	DIGITAL	BUS_CLK	0	42	±0	True	True
Clock_9	DIGITAL	MASTER_CLK	1	1	±0	True	True
Clock_8	DIGITAL	BUS_CLK	0	42	±0	True	True
Clock_10	DIGITAL	BUS_CLK	0	42	±0	True	True

For more information on clocking resources, please refer to:

- Clocking System chapter in the PSoC 3 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 10. Interrupts

Name	Priority	Vector
I2C_2_I2C_IRQ	7	0
isr_Frame	7	3
isr_GPS_PPS	7	4
isr_Locked	7	5
isr_PhaseTiming	7	6
isr_SPIM_1	7	7
isr_Timer_1	7	17
isr_UART_1	7	8
isr_VITC	7	9
MainI2C_isr	7	15
SPIM_1_TxInternalInterrupt	7	1
UART_1_TXInternalInterrupt	7	2

For more information on interrupts, please refer to:

- Interrupt Controller chapter in the PSoC 3 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 3 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 11 lists the Flash protection settings for your design.

Table 11. Flash Protection Settings

Start Address	End Address	Protection Level
0x0	0xFFFF	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 - External read protect (Factory upgrade)
- R - External write protect (Field upgrade)
- W - Full Protection

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

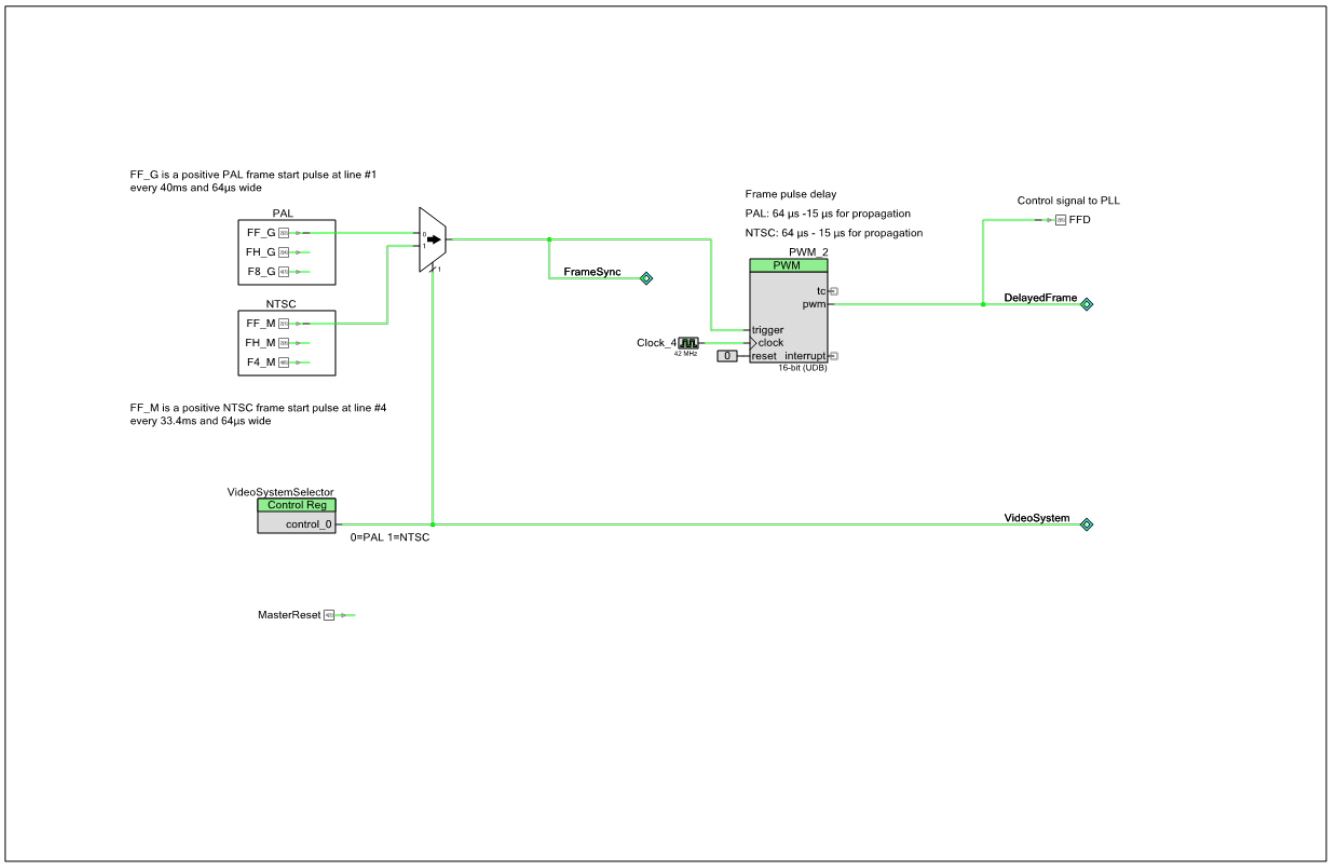
- Flash Protection chapter in the PSoC 3 Technical Reference Manual
- Flash and EEPROM chapter in the System Reference Guide
  - CyFlash API routines
  - CyWrite API routines

7 Design Contents

This design's schematic content consists of the following 12 schematic sheets:

7.1 Schematic Sheet: Frames

Figure 5. Schematic Sheet: Frames

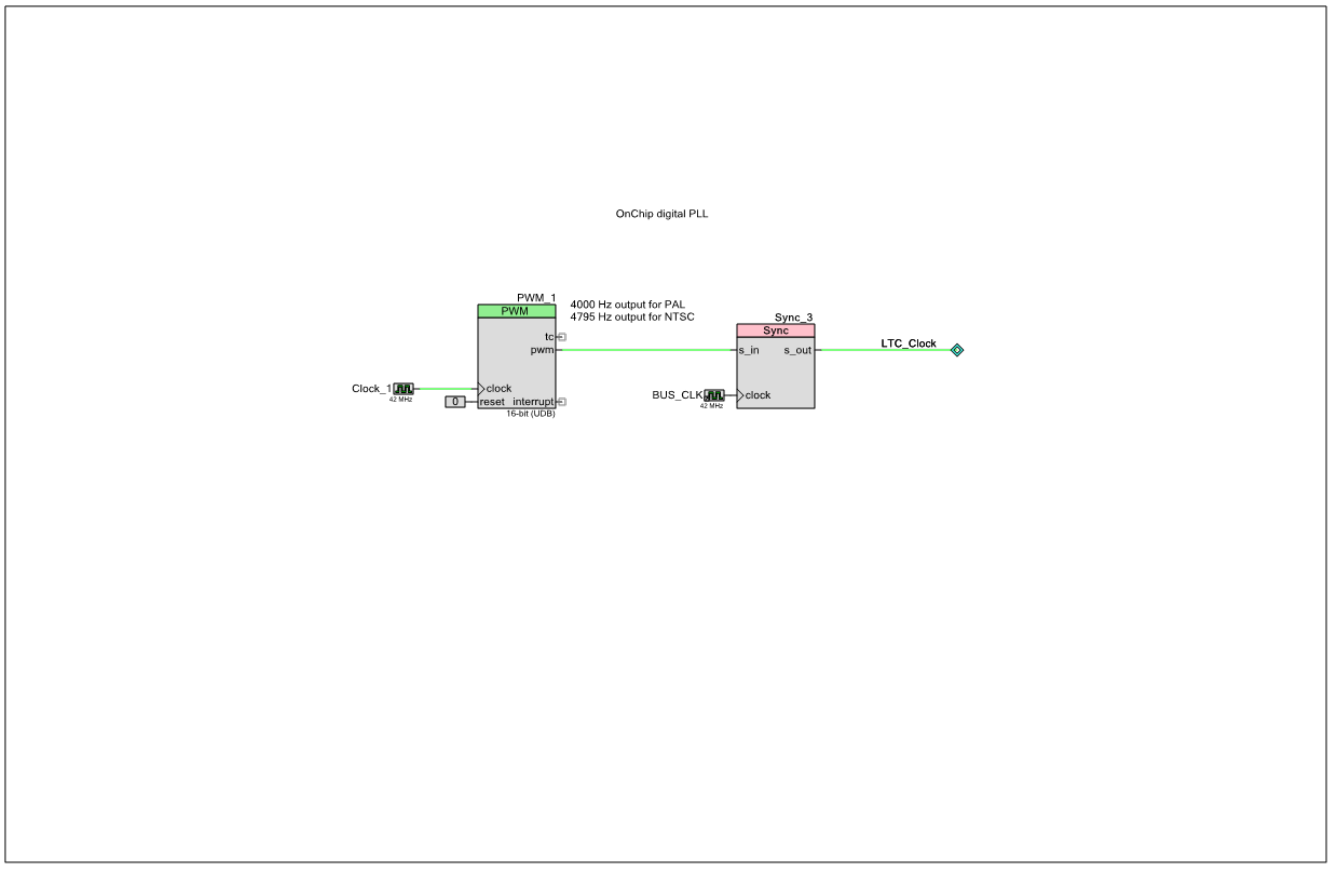


This schematic sheet contains the following component instances:

- Instance PWM_2 (type: PWM_v3_0)
- Instance VideoSystemSelector (type: CyControlReg_v1_70)

7.2 Schematic Sheet: PLL

Figure 6. Schematic Sheet: PLL

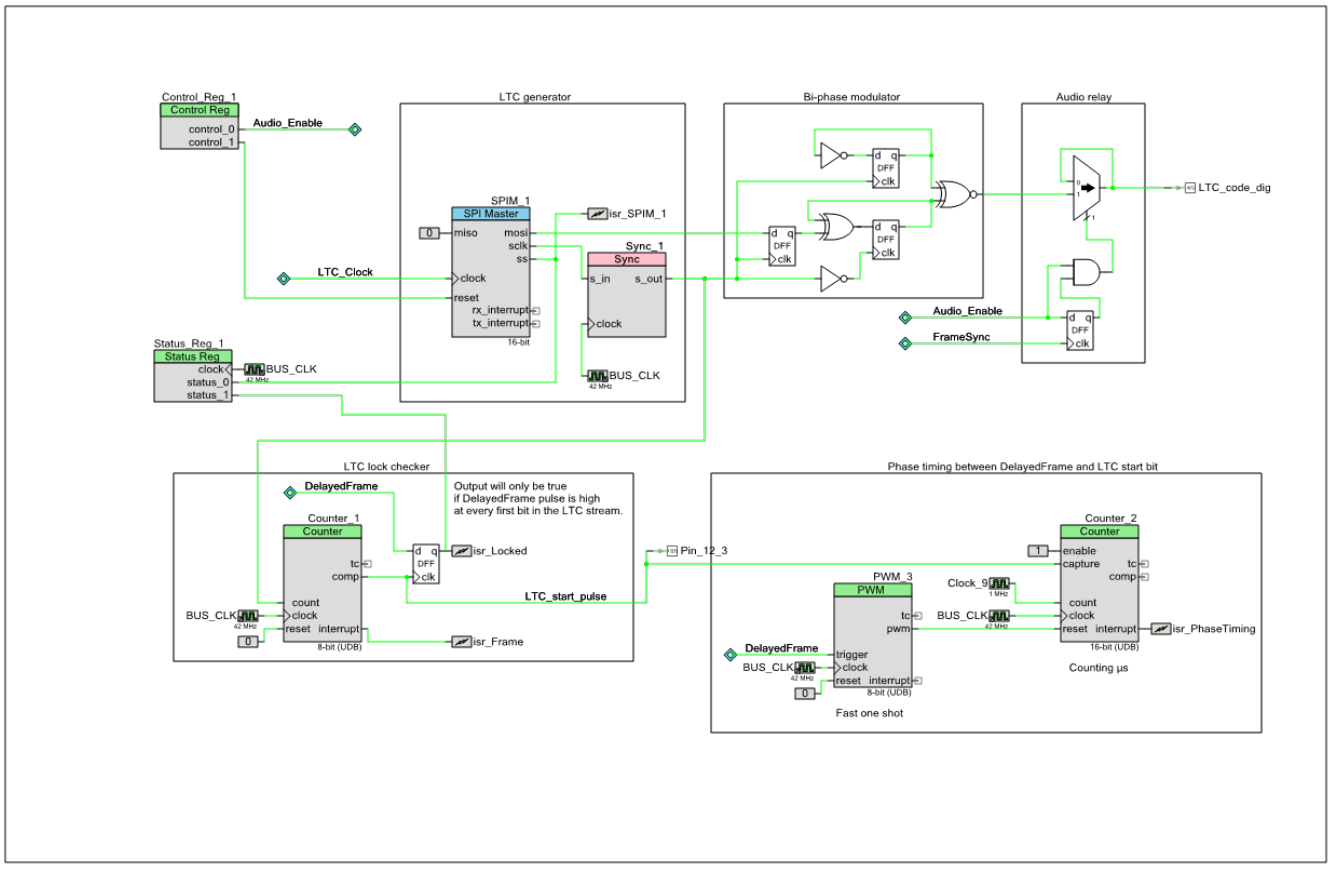


This schematic sheet contains the following component instances:

- Instance PWM_1 (type: PWM_v3_0)

7.3 Schematic Sheet: LTC digital

Figure 7. Schematic Sheet: LTC digital

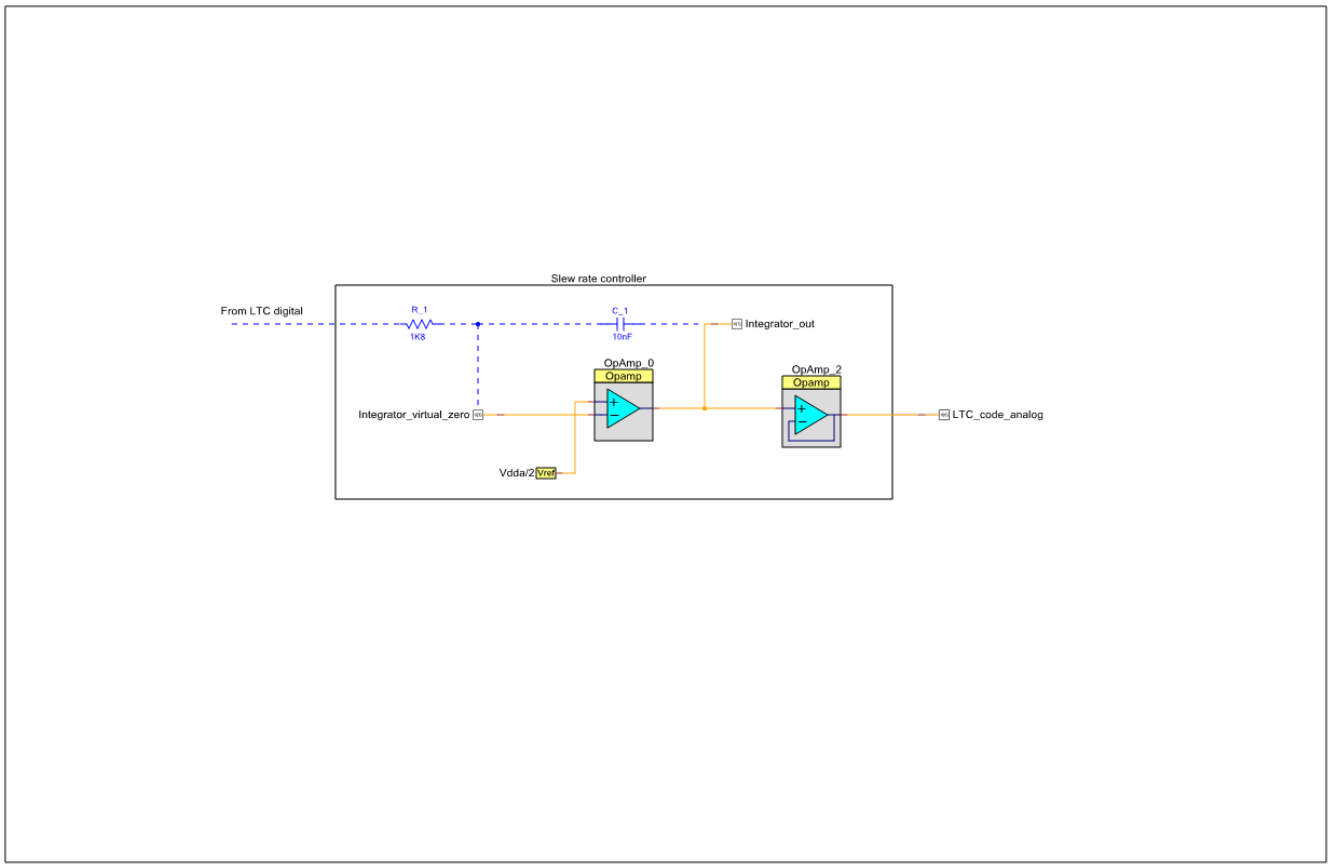


This schematic sheet contains the following component instances:

- Instance Control_Reg_1 (type: CyControlReg_v1_70)
- Instance Counter_1 (type: Counter_v2_40)
- Instance Counter_2 (type: Counter_v2_40)
- Instance PWM_3 (type: PWM_v3_0)
- Instance SPIM_1 (type: SPI_Master_v2_40)
- Instance Status_Reg_1 (type: CyStatusReg_v1_80)

7.4 Schematic Sheet: LTC analog

Figure 8. Schematic Sheet: LTC analog



This schematic sheet contains the following component instances:

- Instance OpAmp_0 (type: OpAmp_v1_90)
- Instance OpAmp_2 (type: OpAmp_v1_90)

7.5 Schematic Sheet: RTC & VITC

Figure 9. Schematic Sheet: RTC & VITC

This schematic sheet contains the following component instances:

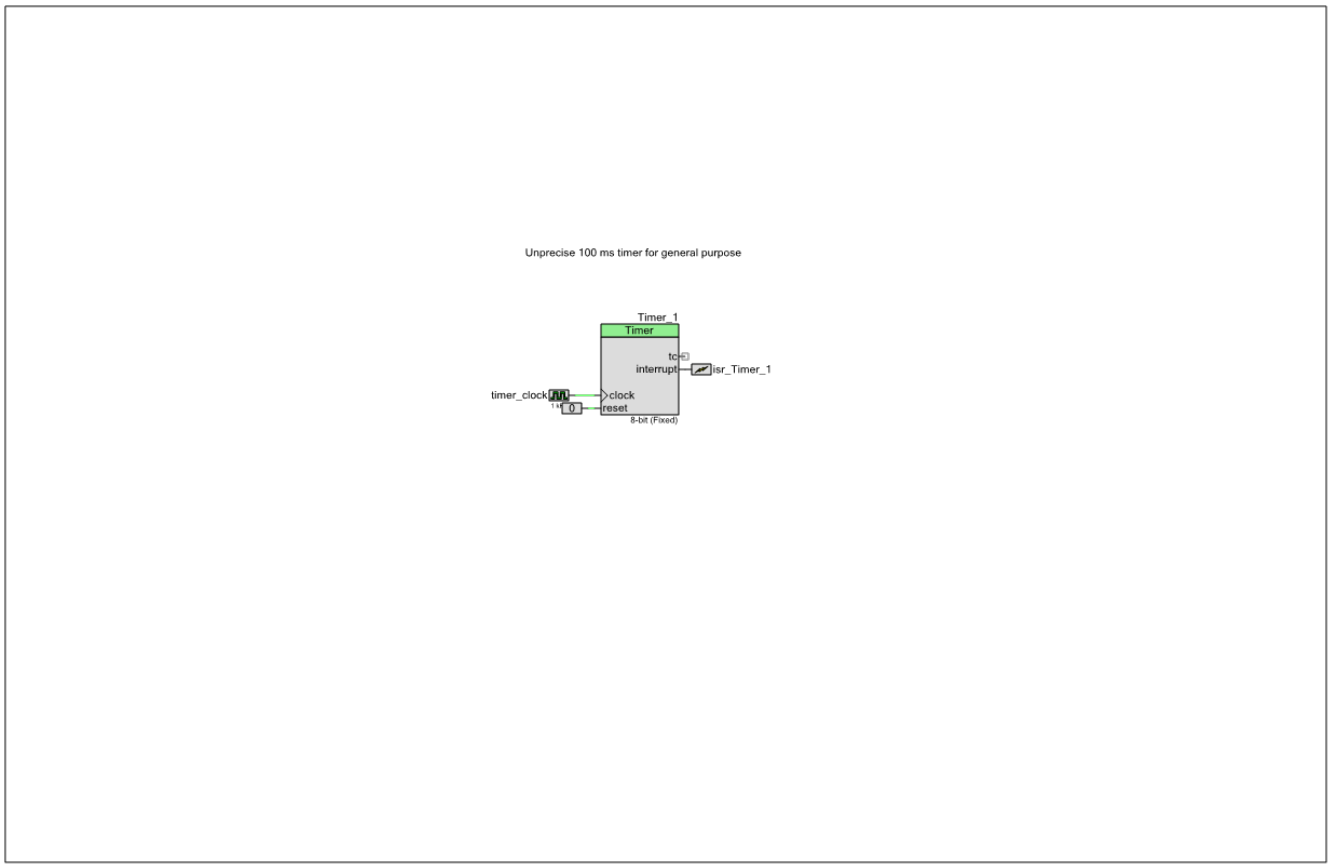
- Instance I2C_2 (type: I2C_v3_30)

7.6 Schematic Sheet: GPS

Figure 10. Schematic Sheet: GPS

7.7 Schematic Sheet: Timers

Figure 11. Schematic Sheet: Timers

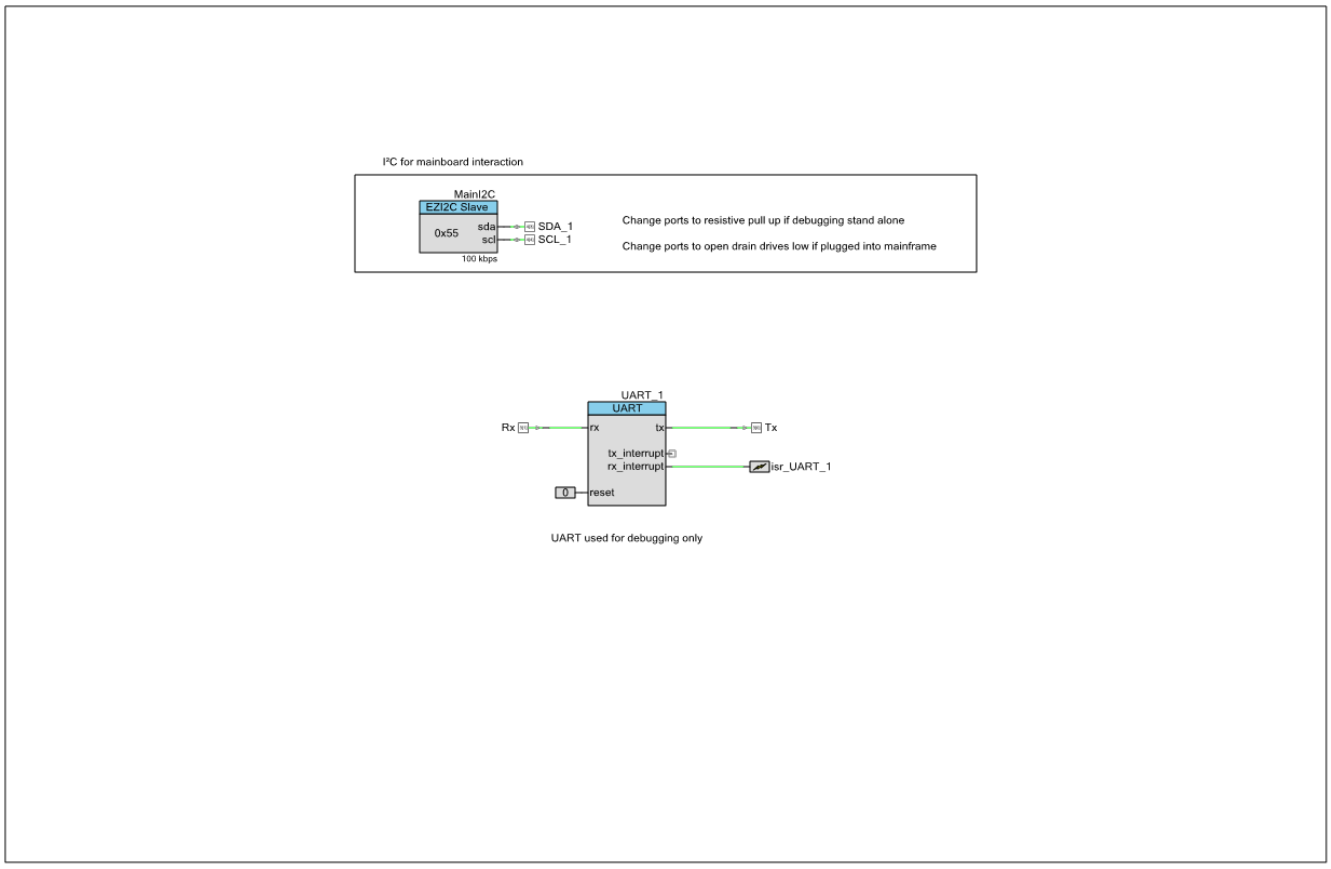


This schematic sheet contains the following component instances:

- Instance Timer_1 (type: Timer_v2_50)

7.8 Schematic Sheet: I²C & UART

Figure 12. Schematic Sheet: I²C & UART

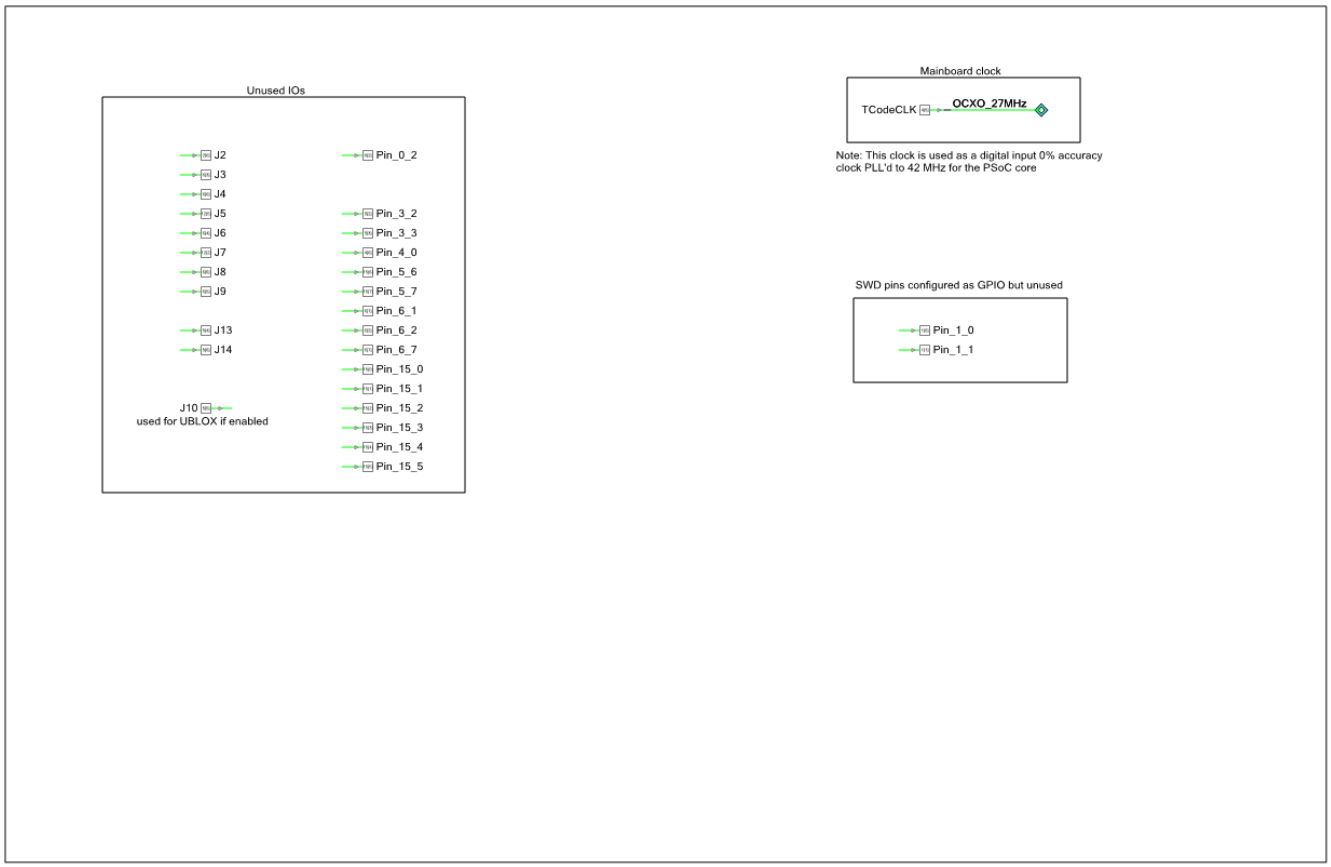


This schematic sheet contains the following component instances:

- Instance MainI2C (type: EZI2C_v1_90)
- Instance UART_1 (type: UART_v2_30)

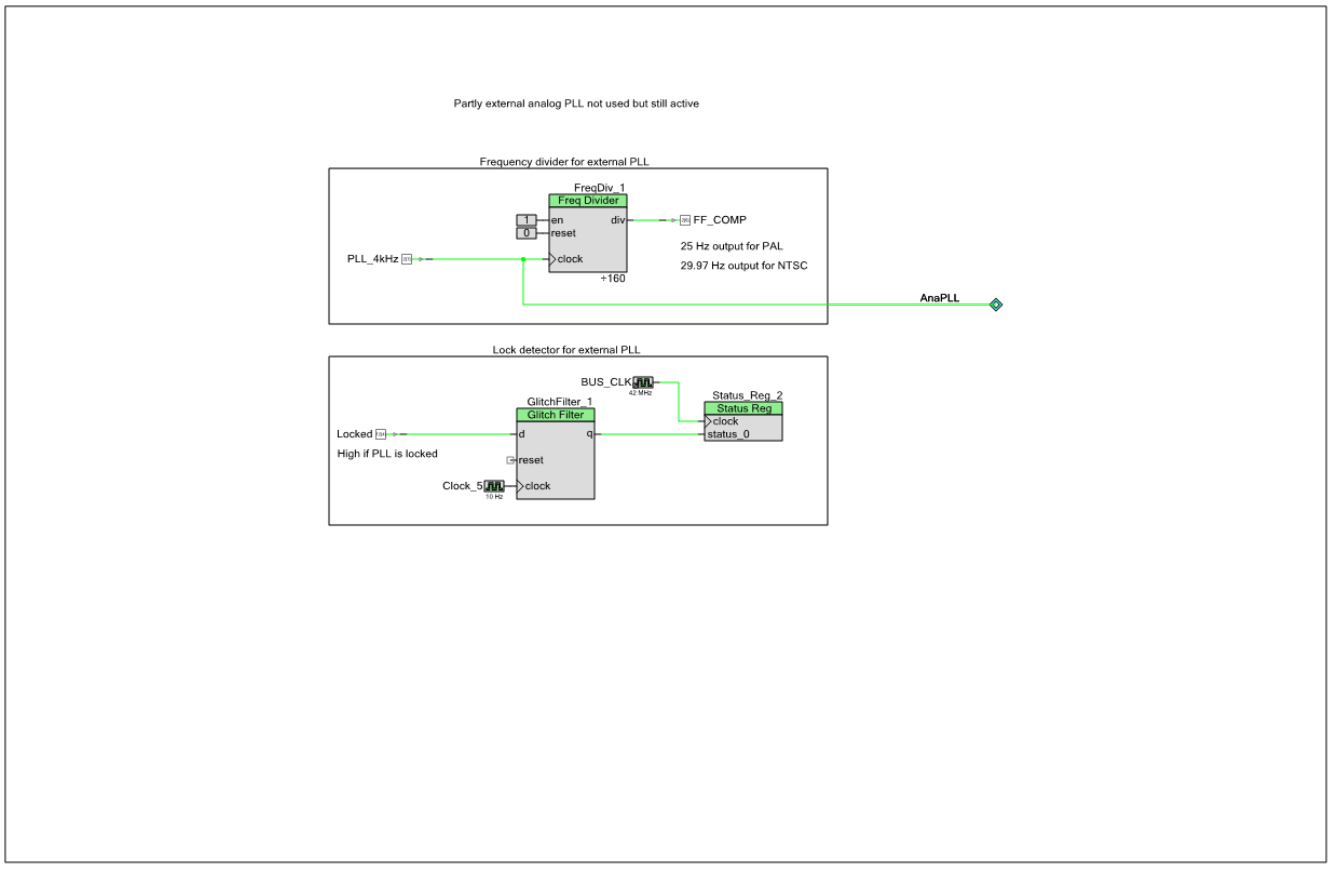
7.9 Schematic Sheet: IOs

Figure 13. Schematic Sheet: IOs



7.10 Schematic Sheet: Analog PLL

Figure 14. Schematic Sheet: Analog PLL

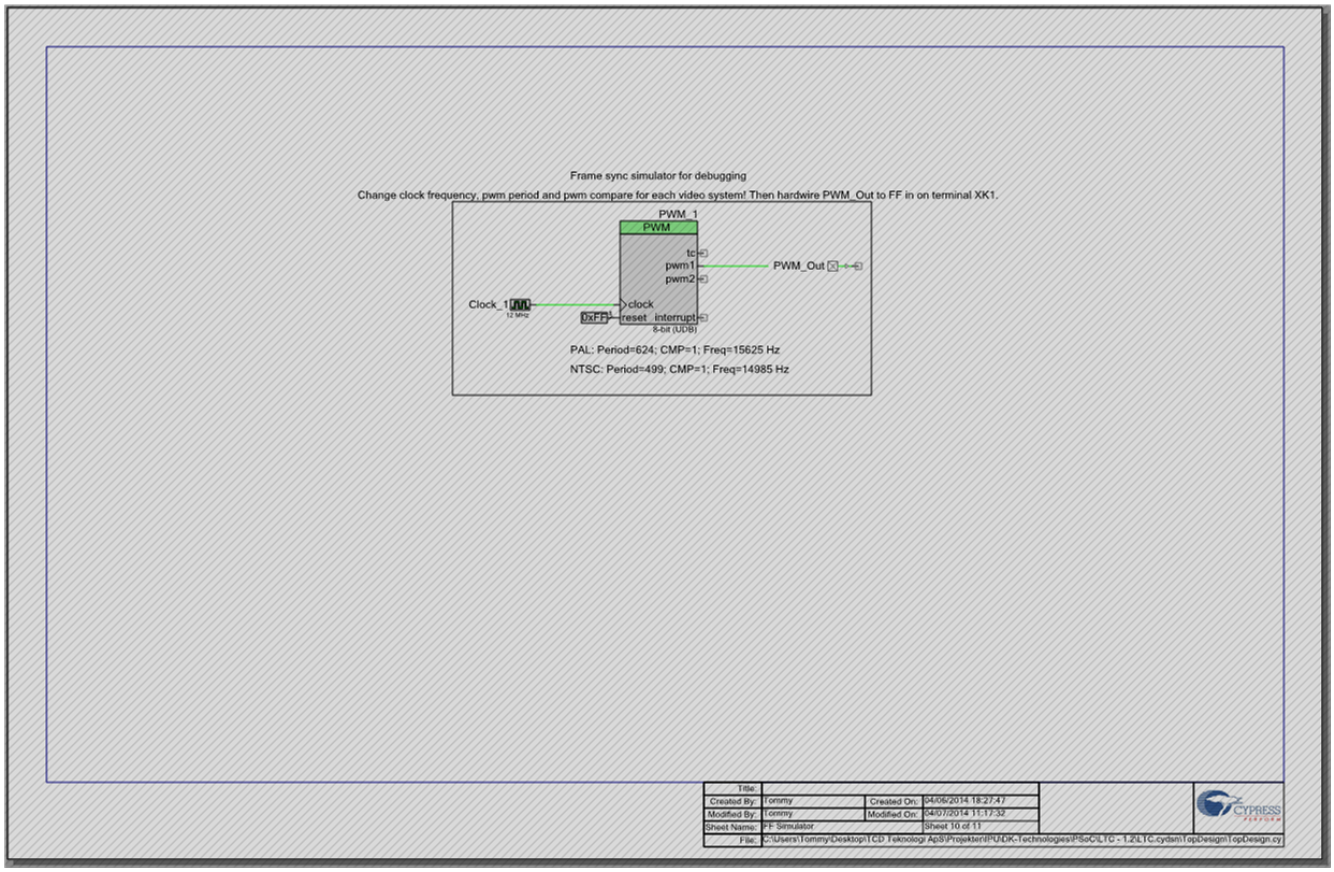


This schematic sheet contains the following component instances:

- Instance FreqDiv_1 (type: FreqDiv_v1_0)
- Instance GlitchFilter_1 (type: GlitchFilter_v2_0)
- Instance Status_Reg_2 (type: CyStatusReg_v1_80)

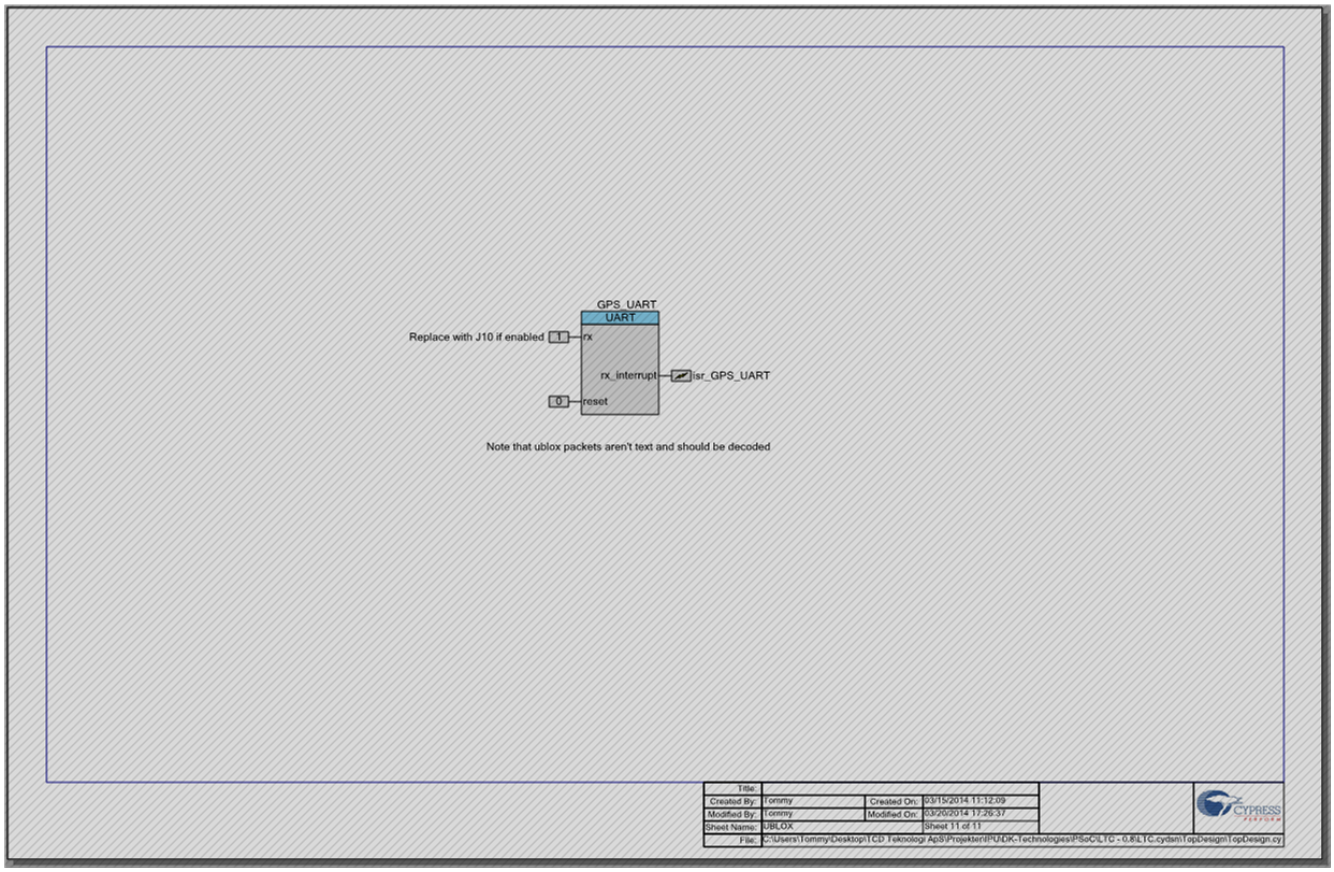
7.11 Schematic Sheet: FF Simulator

Figure 15. Schematic Sheet: FF Simulator



7.12 Schematic Sheet: UBLOX

Figure 16. Schematic Sheet: UBLOX



8 Components

8.1 Component type: Counter [v2.40]

8.1.1 Instance Counter_1

Description: 8, 16, 24 or 32-bit Counter
Instance type: Counter [v2.40]
Datasheet: online component datasheet for Counter

Table 12. Component Parameters for Counter_1

Parameter Name	Value	Description
CaptureMode	None	Defines the functionality of the capture input. Default is None which does not have a capture input pin
ClockMode	Down Counter	Defines the operation of the counter. \nBasic: Count is incremented on the rising edge of the clock input. \n Clock_And_Direction: Clock is incremented or decremented on the rising edge of the clock input based on the direction of the input. \nClock_And_UpCnt_DwnCnt: Clock is an oversampling clock. On the rising edge of UpCnt, the counter is incremented and on the rising edge of DwnCnt, the counter is decremented.
CompareMode	Equal To	Specifies the compare output mode.
CompareStatusEdgeSense	true	Specifies whether rising edge sense for interrupt generation with the Compare output will be used. May be disabled to reduce resource usage.
CompareValue	80	Defines the compare value. Valid vales are from 0 to the period value.
EnableMode	Software Only	Choose which enable controls the enable of the counter. This can be either through software with the control register, through hardware with the input pin or a combination of both where both must be active for the counter to be enabled.
FixedFunction	false	Defines whether Fixed Function Block usage is required.
InterruptOnCapture	false	Enables the counter status register to produce an interrupt output signal on a capture event.
InterruptOnCompare	true	Enables the counter status register to produce an interrupt output signal on compare true.

Parameter Name	Value	Description
InterruptOnOverUnderFlow	false	Enables the counter status register to produce an interrupt output signal on over flow or under flow.
InterruptOnTC	false	Enables the counter status register to produce an interrupt output signal on terminal count.
Period	80	Defines the counter period value in clock counts from 1 to 2^Width-1.
ReloadOnCapture	false	Reloads the counter value to a set value on a capture input event.
ReloadOnCompare	false	Reloads the counter value to a set value on a compare equal event.
ReloadOnOverUnder	true	Reloads the counter value to a set value when overflow or underflow is detected.
ReloadOnReset	true	Reloads the counter value to a set value when reset input is high.
Resolution	8	Defines the width of the counter. It can be 8, 16, 24 or 32 (24 or 32 cannot use Fixed Function block).
RunMode	Continuous	Define the hardware operation to run continuously or run till a terminal count.
UseInterrupt	true	Allows for complete optimization of resource usage down to removing the status register if not required by the user.

8.1.2 Instance Counter_2

Description: 8, 16, 24 or 32-bit Counter
Instance type: Counter [v2.40]
Datasheet: online component datasheet for Counter

Table 13. Component Parameters for Counter_2

Parameter Name	Value	Description
CaptureMode	Rising Edge	Defines the functionality of the capture input. Default is None which does not have a capture input pin

Parameter Name	Value	Description
ClockMode	Up Counter	Defines the operation of the counter. \nBasic: Count is incremented on the rising edge of the clock input. \n Clock_And_Direction: Clock is incremented or decremented on the rising edge of the clock input based on the direction of the input. \nClock_And_UpCnt_DwnCnt: Clock is an oversampling clock. On the rising edge of UpCnt, the counter is incremented and on the rising edge of DwnCnt, the counter is decremented.
CompareMode	Less Than	Specifies the compare output mode.
CompareStatusEdgeSense	true	Specifies whether rising edge sense for interrupt generation with the Compare output will be used. May be disabled to reduce resource usage.
CompareValue	32768	Defines the compare value. Valid vales are from 0 to the period value.
EnableMode	Hardware Only	Choose which enable controls the enable of the counter. This can be either through software with the control register, through hardware with the input pin or a combination of both where both must be active for the counter to be enabled.
FixedFunction	false	Defines whether Fixed Function Block usage is required.
InterruptOnCapture	true	Enables the counter status register to produce an interrupt output signal on a capture event.
InterruptOnCompare	false	Enables the counter status register to produce an interrupt output signal on compare true.
InterruptOnOverUnderFlow	false	Enables the counter status register to produce an interrupt output signal on over flow or under flow.
InterruptOnTC	false	Enables the counter status register to produce an interrupt output signal on terminal count.
Period	65535	Defines the counter period value in clock counts from 1 to 2^Width-1.
ReloadOnCapture	false	Reloads the counter value to a set value on a capture input event.
ReloadOnCompare	false	Reloads the counter value to a set value on a compare equal event.

Parameter Name	Value	Description
ReloadOnOverUnder	false	Reloads the counter value to a set value when overflow or underflow is detected.
ReloadOnReset	true	Reloads the counter value to a set value when reset input is high.
Resolution	16	Defines the width of the counter. It can be 8, 16, 24 or 32 (24 or 32 cannot use Fixed Function block).
RunMode	Continuous	Define the hardware operation to run continuously or run till a terminal count.
UseInterrupt	true	Allows for complete optimization of resource usage down to removing the status register if not required by the user.

8.2 Component type: CyControlReg [v1.70]

8.2.1 Instance Control_Reg_1

Description: The Control Register allows the firmware to set values for to use for digital signals.
Instance type: CyControlReg [v1.70]
Datasheet: online component datasheet for CyControlReg

Table 14. Component Parameters for Control_Reg_1

Parameter Name	Value	Description
Bit0Mode	DirectMode	Defines bit 0 mode
Bit1Mode	DirectMode	Defines bit 1 mode
Bit2Mode	DirectMode	Defines bit 2 mode
Bit3Mode	DirectMode	Defines bit 3 mode
Bit4Mode	DirectMode	Defines bit 4 mode
Bit5Mode	DirectMode	Defines bit 5 mode
Bit6Mode	DirectMode	Defines bit 6 mode
Bit7Mode	DirectMode	Defines bit 7 mode
BitValue	0	Defines bit value
BusDisplay	false	Displays the output terminals as bus
ExternalReset	false	Shows the reset terminal
NumOutputs	2	Defines the number of outputs needed (1-8)

8.2.2 Instance VideoSystemSelector

Description: The Control Register allows the firmware to set values for to use for digital signals.
Instance type: CyControlReg [v1.70]
Datasheet: online component datasheet for CyControlReg

Table 15. Component Parameters for VideoSystemSelector

Parameter Name	Value	Description
Bit0Mode	DirectMode	Defines bit 0 mode
Bit1Mode	DirectMode	Defines bit 1 mode

Parameter Name	Value	Description
Bit2Mode	DirectMode	Defines bit 2 mode
Bit3Mode	DirectMode	Defines bit 3 mode
Bit4Mode	DirectMode	Defines bit 4 mode
Bit5Mode	DirectMode	Defines bit 5 mode
Bit6Mode	DirectMode	Defines bit 6 mode
Bit7Mode	DirectMode	Defines bit 7 mode
BitValue	0	Defines bit value
BusDisplay	false	Displays the output terminals as bus
ExternalReset	false	Shows the reset terminal
NumOutputs	1	Defines the number of outputs needed (1-8)

8.3 Component type: CyStatusReg [v1.80]

8.3.1 Instance Status_Reg_1

Description: The Status Register allows the firmware to read values from digital signals.
Instance type: CyStatusReg [v1.80]
Datasheet: online component datasheet for CyStatusReg

Table 16. Component Parameters for Status_Reg_1

Parameter Name	Value	Description
Bit0Mode	Transparent	Bit Mode for Bit 0 of the Status Register
Bit1Mode	Transparent	Bit Mode for Bit 1 of the Status Register
Bit2Mode	Sticky (Clear on Read)	Bit Mode for Bit 2 of the Status Register
Bit3Mode	Transparent	Bit Mode for Bit 3 of the Status Register
Bit4Mode	Transparent	Bit Mode for Bit 4 of the Status Register
Bit5Mode	Transparent	Bit Mode for Bit 5 of the Status Register
Bit6Mode	Transparent	Bit Mode for Bit 6 of the Status Register
Bit7Mode	Transparent	Bit Mode for Bit 7 of the Status Register
BusDisplay	false	Displays the input terminals as bus
Interrupt	false	Shows the interrupt terminal
MaskValue	0	Defines the value of the interrupt mask
NumInputs	2	Defines the number of status inputs (1-8)

8.3.2 Instance Status_Reg_2

Description: The Status Register allows the firmware to read values from digital signals.
Instance type: CyStatusReg [v1.80]
Datasheet: online component datasheet for CyStatusReg

Table 17. Component Parameters for Status_Reg_2

Parameter Name	Value	Description
Bit0Mode	Transparent	Bit Mode for Bit 0 of the Status Register
Bit1Mode	Transparent	Bit Mode for Bit 1 of the Status Register
Bit2Mode	Transparent	Bit Mode for Bit 2 of the Status Register
Bit3Mode	Transparent	Bit Mode for Bit 3 of the Status Register
Bit4Mode	Transparent	Bit Mode for Bit 4 of the Status Register
Bit5Mode	Transparent	Bit Mode for Bit 5 of the Status Register
Bit6Mode	Transparent	Bit Mode for Bit 6 of the Status Register
Bit7Mode	Transparent	Bit Mode for Bit 7 of the Status Register
BusDisplay	false	Displays the input terminals as bus
Interrupt	false	Shows the interrupt terminal
MaskValue	0	Defines the value of the interrupt mask
NumInputs	1	Defines the number of status inputs (1-8)

8.4 Component type: EZI2C [v1.90]

8.4.1 Instance MainI2C

Description: Easy to use I2C slave.
Instance type: EZI2C [v1.90]
Datasheet: online component datasheet for EZI2C

Table 18. Component Parameters for MainI2C

Parameter Name	Value	Description
BusSpeed_kHz	100	Data rate in kbps. Standard settings are 50, 100, 400 or 1000. The value must be between 50 and 1000.
EnableWakeup	false	Selects wakeup during low power state.
Hex1	true	Indicates whether primary slave address was input in hexadecimal or decimal
Hex2	false	Indicates whether secondary slave address was input in hexadecimal or decimal
I2C_Address1	85	Primary I2C slave address, always valid. (Range 0 to 127)
I2C_Address2	9	Second slave address. Only valid when two I2C Addresses are selected. (Range 0 to 127)
I2C_Addresses	1	Select between 1 or 2 slave addresses.

Parameter Name	Value	Description
I2cBusPort	Any	Select "Any" if no wakeup feature is supposed to be used. Select I2C0/I2C1 if wakeup feature is supposed to be available and bus pins are connected to the corresponding ports.
Sub_Address_Size	8 Bits	Selects either 8 or 16 bit sub-address decoding.

8.5 Component type: FreqDiv [v1.0]

8.5.1 Instance FreqDiv_1

Description: Frequency Divider
Instance type: FreqDiv [v1.0]
Datasheet: online component datasheet for FreqDiv

Table 19. Component Parameters for FreqDiv_1

Parameter Name	Value	Description
Divider	160	The divider used to generate the div output from the clock input.
HighPulseTime	0	Number of clock cycles each clock period that the div output is high. 0 indicates 50% duty cycle.

8.6 Component type: GlitchFilter [v2.0]

8.6.1 Instance GlitchFilter_1

Description: Removes unwanted pulses from a digital signal
Instance type: GlitchFilter [v2.0]
Datasheet: online component datasheet for GlitchFilter

Table 20. Component Parameters for GlitchFilter_1

Parameter Name	Value	Description
BypassFilter	Logic Zero	Specifies the logic level to be directly propagated to the output.
GlitchLength	10	Defines the number of samples for which input has to be stable before being propagated to the output.
SignalWidth	1	Determines the bus width of d and q terminals.

8.7 Component type: I2C [v3.30]

8.7.1 Instance I2C_2

Description: Standard I2C communication interface
Instance type: I2C [v3.30]
Datasheet: online component datasheet for I2C

Table 21. Component Parameters for I2C_2

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	false	Indicates that address has been input in hexadecimal format.
I2C_Mode	Master	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	8	7-bits I2C slave address.
SlaveClockMinusTolerance	5	Internal component clock negative tolerance value in Slave mode.

Parameter Name	Value	Description
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	true	Determines either internal or external clock source for I2C UDB.
UdbSlaveFixedPlacementEnable	false	Enables fixed placement for I2C UDB. Only available in slave mode.

8.8 Component type: OpAmp [v1.90]

8.8.1 Instance OpAmp_0

Description: Opamp
Instance type: OpAmp [v1.90]
Datasheet: online component datasheet for OpAmp

Table 22. Component Parameters for OpAmp_0

Parameter Name	Value	Description
Mode	OpAmp	Selects between uncommitted op-amp or follower mode.
Power	High Power	Selects the device power level.

8.8.2 Instance OpAmp_2

Description: Opamp
Instance type: OpAmp [v1.90]
Datasheet: online component datasheet for OpAmp

Table 23. Component Parameters for OpAmp_2

Parameter Name	Value	Description
Mode	Follower	Selects between uncommitted op-amp or follower mode.
Power	High Power	Selects the device power level.

8.9 Component type: PWM [v3.0]

8.9.1 Instance PWM_1

Description: 8 or 16-bit Pulse Width Modulator
Instance type: PWM [v3.0]
Datasheet: online component datasheet for PWM

Table 24. Component Parameters for PWM_1

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
CompareType1	Greater	Sets the compare value comparison type setting for the compare 1 output
CompareType2	Less	Sets the compare value comparison type setting for the compare 2 output
CompareValue1	5000	Compares Output 1 to value
CompareValue2	63	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	10499	Defines the PWM period value
PWMMode	One Output	Defines the overall mode of the PWM
Resolution	16	Defines the bit width of the PWM (8 or 16 bits)

Parameter Name	Value	Description
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

8.9.2 Instance PWM_2

Description: 8 or 16-bit Pulse Width Modulator
Instance type: PWM [v3.0]
Datasheet: online component datasheet for PWM

Table 25. Component Parameters for PWM_2

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
CompareType1	Less or Equal	Sets the compare value comparison type setting for the compare 1 output
CompareType2	Less or Equal	Sets the compare value comparison type setting for the compare 2 output
CompareValue1	210	Compares Output 1 to value
CompareValue2	2141	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

Parameter Name	Value	Description
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	2267	Defines the PWM period value
PWMMode	One Output	Defines the overall mode of the PWM
Resolution	16	Defines the bit width of the PWM (8 or 16 bits)
RunMode	One Shot with Multi Trigger	Defines the run mode options to be either continuous or one shot
TriggerMode	Rising Edge	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

8.9.3 Instance PWM_3

Description: 8 or 16-bit Pulse Width Modulator
Instance type: PWM [v3.0]
Datasheet: online component datasheet for PWM

Table 26. Component Parameters for PWM_3

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
CompareType1	Less or Equal	Sets the compare value comparison type setting for the compare 1 output
CompareType2	Less	Sets the compare value comparison type setting for the compare 2 output
CompareValue1	9	Compares Output 1 to value
CompareValue2	63	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.

Parameter Name	Value	Description
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	10	Defines the PWM period value
PWMMode	One Output	Defines the overall mode of the PWM
Resolution	8	Defines the bit width of the PWM (8 or 16 bits)
RunMode	One Shot with Multi Trigger	Defines the run mode options to be either continuous or one shot
TriggerMode	Rising Edge	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

8.10 Component type: SPI_Master [v2.40]

8.10.1 Instance SPIM_1

Description: Serial Peripheral Interface Master
Instance type: SPI_Master [v2.40]
Datasheet: online component datasheet for SPI_Master

Table 27. Component Parameters for SPIM_1

Parameter Name	Value	Description
BidirectMode	false	Bidirectional mode setting
ClockInternal	false	Allow use of the internal clock and desired bit rate or an external clock source
DesiredBitRate	1000000	Desired Bit Rate in bps
HighSpeedMode	false	Enables using of the High Speed Mode
InterruptOnByteComplete	false	Set Initial Interrupt Source to Enable Interrupt on Byte Transfer Complete
InterruptOnRXFull	false	Set Initial Interrupt Source to Enable Interrupt on RX FIFO Full

Parameter Name	Value	Description
InterruptOnRXNotEmpty	false	Set Initial Interrupt Source to Enable Interrupt on RX FIFO Not Empty
InterruptOnRXOverrun	false	Set Initial Interrupt Source to Enable Interrupt on RX FIFO Overrun
InterruptOnSPIDone	false	Set Initial Interrupt Source to Enable Interrupt on SPI Done
InterruptOnSPIIdle	false	Set Initial Interrupt Source to Enable Interrupt on SPI Idle
InterruptOnTXEmpty	false	Set Initial Interrupt Source to Enable Interrupt on TX FIFO Empty
InterruptOnTXNotFull	true	Set Initial Interrupt Source to Enable Interrupt on TX FIFO Not Full
Mode	CPHA = 0, CPOL = 0	SPI mode defines the Clock Phase and Clock Polarity desired
NumberOfDataBits	16	Set the Number of Data bits 3-16
RxBufferSize	4	Defines the amount of RAM Set asside for the RX Buffer
ShiftDir	LSB First	Set the Shift Out Direction
TxBufferSize	10	Defines the amount of RAM Set asside for the TX Buffer
UseRxInternalInterrupt	false	Defines whether Rx internal interrupt is used or not
UseTxInternalInterrupt	true	Defines whether Tx internal interrupt is used or not

8.11 Component type: Timer [v2.50]

8.11.1 Instance Timer_1

Description: 8, 16, 24 or 32-bit Timer
Instance type: Timer [v2.50]
Datasheet: online component datasheet for Timer

Table 28. Component Parameters for Timer_1

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.

Parameter Name	Value	Description
CaptureCounterEnabled	false	Enables the capture counter to count capture events (up to 127) before a capture is triggered.
CaptureMode	None	This parameter defines the capture input signal requirements to trigger a valid capture event
EnableMode	Software 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	true	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	false	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	1	Number of captures allowed until the counter is cleared or disabled.
Period	99	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	Continuous	Defines the hardware to run continuously, run until a terminal count is reached or run until an interrupt event is triggered.
TriggerMode	None	Defines the required trigger input signal to cause a valid trigger enable of the timer

8.12 Component type: UART [v2.30]

8.12.1 Instance UART_1

Description: Universal Asynchronous Receiver Transmitter
Instance type: UART [v2.30]
Datasheet: online component datasheet for UART

Table 29. Component Parameters for UART_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	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	true	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	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	true	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.

Parameter Name	Value	Description
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	50	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.

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 3 register map is covered in the PSoC 3 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 3 Technical Reference Manual
  - Low Power Modes chapter in the PSoC 3 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 3 Technical Reference Manual
  - Cache chapter in the System Reference Guide