PSoC® Creator™
Project Datasheet for johannesPSoC5GreenhouseAutomation

Creation Time: 05/29/2025 15:04:02
User: JOSE\jajuv
Project: johannesPSoC5GreenhouseAutomation
Tool: PSoC Creator 4.2

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone (USA): 800.858.1810

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

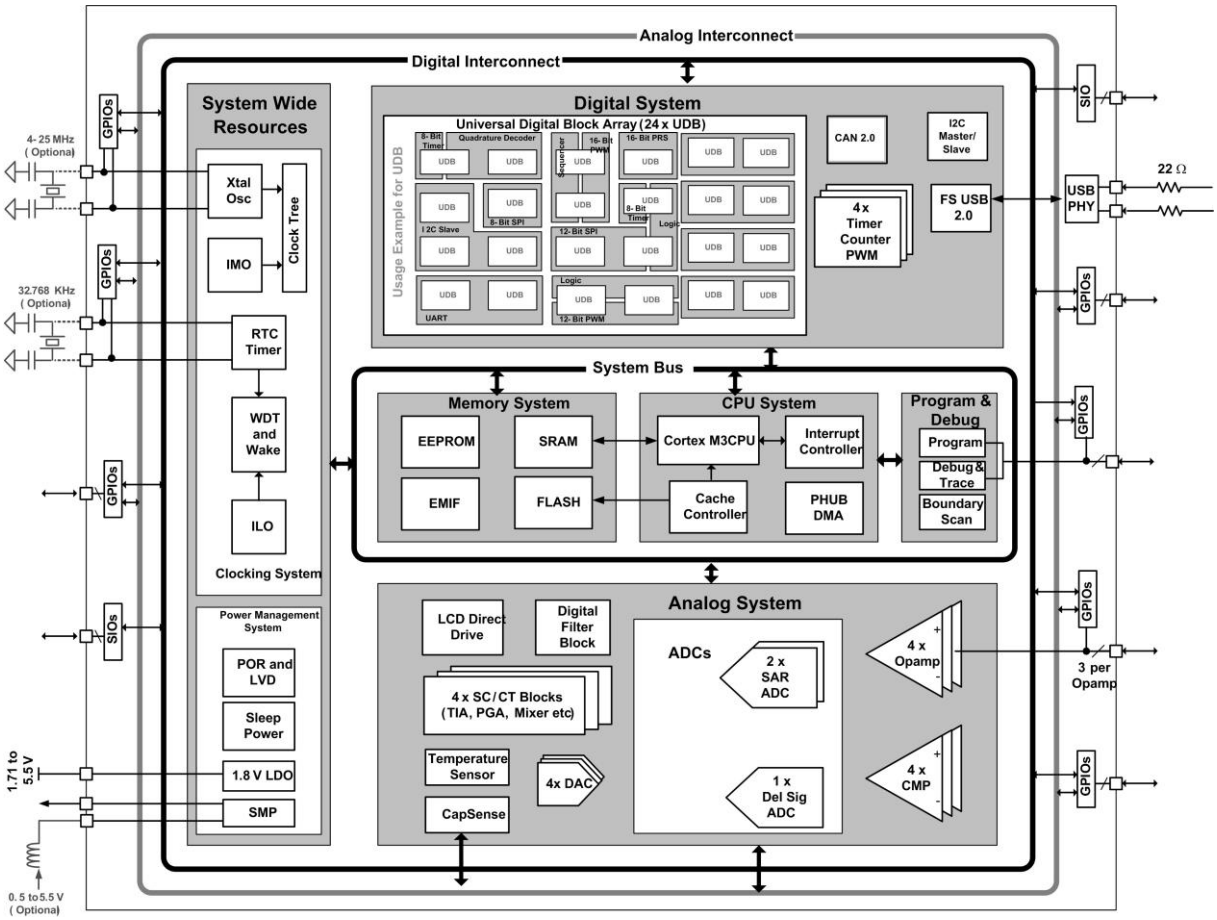


Table 1 lists the key characteristics of this device.

Resource Type	Used	Free	Max	% Used
Digital Clocks	5	3	8	62.50 %
Analog Clocks	1	3	4	25.00 %
CapSense Buffers	1	1	2	50.00 %
Digital Filter Block	0	1	1	0.00 %
Interrupts	4	28	32	12.50 %
IO	9	39	48	18.75 %
Segment LCD	0	1	1	0.00 %
CAN 2.0b	0	1	1	0.00 %
I2C	0	1	1	0.00 %
USB	0	1	1	0.00 %
DMA Channels	0	24	24	0.00 %
Timer	1	3	4	25.00 %
UDB				
Macrocells	47	145	192	24.48 %
Unique P-terms	78	306	384	20.31 %
Total P-terms	88			
Datapath Cells	10	14	24	41.67 %
Status Cells	5	19	24	20.83 %
StatusI Registers	3			
Sync Cells (x1)	1			
Routed Count7 Load/Enable	1			
Control Cells	4	20	24	16.67 %
Control Registers	2			
Count7 Cells	2			
Opamp	0	4	4	0.00 %

Comparator	1	3	4	25.00 %
Delta-Sigma ADC	1	0	1	100.00 %
LPF	0	2	2	0.00 %
SAR ADC	0	2	2	0.00 %
Analog (SC/CT) Blocks	0	4	4	0.00 %

Table 1. Device Characteristics

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

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

Table 2 lists the device resources that this design uses:

Table 2. Device Resources

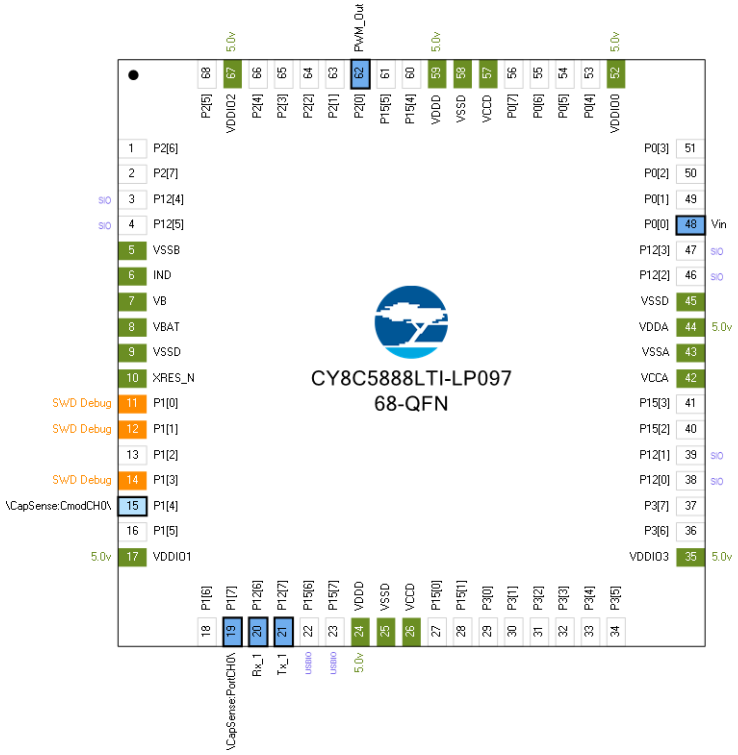
Resource Type	Used	Free	Max	% Used
DAC				
VIDAC	1	3	4	25.00 %

about the device

2 Pins

Figure 2 shows the pin layout of this device.

Figure 2. Device Pin Layout



2.1 Hardware Pins

Pin	Port	Name	Type	Drive Mode	Reset State
1	P2[6]	GPIO [unused]			HiZ Analog Unb
2	P2[7]	GPIO [unused]			HiZ Analog Unb
3	P12[4]	SIO [unused]			HiZ Analog Unb
4	P12[5]	SIO [unused]			HiZ Analog Unb
5	VSSB	VSSB	Dedicated		
6	IND	IND	Dedicated		
7	VB	VB	Dedicated		

8	VBAT	VBAT	Dedicated		
9	VSSD	VSSD	Power		
10	XRES_N	XRES_N	Dedicated		
11	P1[0]	Debug:SWD_IO	Reserved		
12	P1[1]	Debug:SWD_CK	Reserved		
13	P1[2]	GPIO [unused]			HiZ Analog Unb
14	P1[3]	Debug:SWV	Reserved		
15	P1[4]	\CapSense:CmodCH0\	Analog	HiZ analog	HiZ Analog Unb
16	P1[5]	GPIO [unused]			HiZ Analog Unb
17	VDDIO1	VDDIO1	Power		
18	P1[6]	GPIO [unused]			HiZ Analog Unb
19	P1[7]	\CapSense:PortCH0\	Analog	OD, DL	HiZ Analog Unb
20	P12[6]	Rx_1	Dgtl In	HiZ digital	HiZ Analog Unb
21	P12[7]	Tx_1	Dgtl Out	Strong drive	HiZ Analog Unb
22	P15[6]	USB IO [unused]			HiZ Analog Unb
23	P15[7]	USB IO [unused]			HiZ Analog Unb
24	VDDD	VDDD	Power		
25	VSSD	VSSD	Power		
26	VCCD	VCCD	Power		
27	P15[0]	GPIO [unused]			HiZ Analog Unb
28	P15[1]	GPIO [unused]			HiZ Analog Unb
29	P3[0]	GPIO [unused]			HiZ Analog Unb
30	P3[1]	GPIO [unused]			HiZ Analog Unb
31	P3[2]	GPIO [unused]			HiZ Analog Unb
32	P3[3]	GPIO [unused]			HiZ Analog Unb
33	P3[4]	GPIO [unused]			HiZ Analog Unb
34	P3[5]	GPIO [unused]			HiZ Analog Unb
35	VDDIO3	VDDIO3	Power		
36	P3[6]	GPIO [unused]			HiZ Analog Unb
37	P3[7]	GPIO [unused]			HiZ Analog Unb
38	P12[0]	SIO [unused]			HiZ Analog Unb

about the device

39	P12[1]	SIO [unused]			HiZ Analog Unb
40	P15[2]	GPIO [unused]			HiZ Analog Unb
41	P15[3]	GPIO [unused]			HiZ Analog Unb
42	VCCA	VCCA	Power		
43	VSSA	VSSA	Power		
44	VDDA	VDDA	Power		
45	VSSD	VSSD	Power		

Table 3 contains information

pins have been omitted.)

Table 3. Device Pins

pins on this device in pin order. (No connection ["n/c"]

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

Abbreviations used in Table 3 have the following meanings:

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

2.2 Hardware Ports

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

about the device

Table 4. Device Ports

Port	Pin	Name	Type	Drive Mode	Reset State
P0[0]	48	Vin	Analog	HiZ analog	HiZ Analog Unb
P0[1]	49	GPIO [unused]			HiZ Analog Unb
P0[2]	50	GPIO [unused]			HiZ Analog Unb
P0[3]	51	GPIO [unused]			HiZ Analog Unb
P0[4]	53	GPIO [unused]			HiZ Analog Unb
P0[5]	54	GPIO [unused]			HiZ Analog Unb
P0[6]	55	GPIO [unused]			HiZ Analog Unb
P0[7]	56	GPIO [unused]			HiZ Analog Unb
P1[0]	11	Debug:SWD_IO	Reserved		
P1[1]	12	Debug:SWD_CK	Reserved		
P1[2]	13	GPIO [unused]			HiZ Analog Unb
P1[3]	14	Debug:SWV	Reserved		
P1[4]	15	\CapSense:CmodCH0\	Analog	HiZ analog	HiZ Analog Unb
P1[5]	16	GPIO [unused]			HiZ Analog Unb
P1[6]	18	GPIO [unused]			HiZ Analog Unb
P1[7]	19	\CapSense:PortCH0\	Analog	OD, DL	HiZ Analog Unb
P12[0]	38	SIO [unused]			HiZ Analog Unb
P12[1]	39	SIO [unused]			HiZ Analog Unb
P12[2]	46	SIO [unused]			HiZ Analog Unb
P12[3]	47	SIO [unused]			HiZ Analog Unb
P12[4]	3	SIO [unused]			HiZ Analog Unb
P12[5]	4	SIO [unused]			HiZ Analog Unb
P12[6]	20	Rx_1	Dgtl In	HiZ digital	HiZ Analog Unb
P12[7]	21	Tx_1	Dgtl Out	Strong drive	HiZ Analog Unb
P15[0]	27	GPIO [unused]			HiZ Analog Unb
P15[1]	28	GPIO [unused]			HiZ Analog Unb
P15[2]	40	GPIO [unused]			HiZ Analog Unb
P15[3]	41	GPIO [unused]			HiZ Analog Unb

P15[4]	60	GPIO [unused]			HiZ Analog Unb
P15[5]	61	GPIO [unused]			HiZ Analog Unb
P15[6]	22	USB IO [unused]			HiZ Analog Unb
P15[7]	23	USB IO [unused]			HiZ Analog Unb
P2[0]	62	PWM_Out	Dgtl Out	Strong drive	HiZ Analog Unb
P2[1]	63	GPIO [unused]			HiZ Analog Unb
P2[2]	64	GPIO [unused]			HiZ Analog Unb
P2[3]	65	GPIO [unused]			HiZ Analog Unb
P2[4]	66	GPIO [unused]			HiZ Analog Unb
P2[5]	68	GPIO [unused]			HiZ Analog Unb
P2[6]	1	GPIO [unused]			HiZ Analog Unb
P2[7]	2	GPIO [unused]			HiZ Analog Unb
P3[0]	29	GPIO [unused]			HiZ Analog Unb
P3[1]	30	GPIO [unused]			HiZ Analog Unb
P3[2]	31	GPIO [unused]			HiZ Analog Unb
P3[3]	32	GPIO [unused]			HiZ Analog Unb
P3[4]	33	GPIO [unused]			HiZ Analog Unb
Port	**Pin**	**Name**	**Type**	**Drive Mode**	**Reset State**
P3[5]	34	GPIO [unused]			HiZ Analog Unb
P3[6]	36	GPIO [unused]			HiZ Analog Unb
P3[7]	37	GPIO [unused]			HiZ Analog Unb

Abbreviations used in Table 4 have the following meanings:

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

about the device

2.3 Software Pins

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

Table 5. Software Pins

Name	Port	Type	Reset State
\CapSense:CmodCH0\	P1[4]	Analog	HiZ Analog Unb
\CapSense:PortCH0\	P1[7]	Analog	HiZ Analog Unb
Debug:SWD_CK	P1[1]	Reserved	
Debug:SWD_IO	P1[0]	Reserved	
Debug:SWV	P1[3]	Reserved	
GPIO [unused]	P0[2]		HiZ Analog Unb
GPIO [unused]	P0[3]		HiZ Analog Unb
GPIO [unused]	P0[1]		HiZ Analog Unb
GPIO [unused]	P15[3]		HiZ Analog Unb
GPIO [unused]	P3[6]		HiZ Analog Unb
GPIO [unused]	P3[5]		HiZ Analog Unb
GPIO [unused]	P3[4]		HiZ Analog Unb
GPIO [unused]	P15[2]		HiZ Analog Unb
GPIO [unused]	P3[3]		HiZ Analog Unb
GPIO [unused]	P3[7]		HiZ Analog Unb
GPIO [unused]	P0[4]		HiZ Analog Unb
GPIO [unused]	P2[2]		HiZ Analog Unb
GPIO [unused]	P2[1]		HiZ Analog Unb
GPIO [unused]	P2[3]		HiZ Analog Unb
GPIO [unused]	P2[5]		HiZ Analog Unb
GPIO [unused]	P2[4]		HiZ Analog Unb
GPIO [unused]	P0[6]		HiZ Analog Unb
GPIO [unused]	P0[5]		HiZ Analog Unb
GPIO [unused]	P0[7]		HiZ Analog Unb

GPIO [unused]	P15[5]		HiZ Analog Unb
GPIO [unused]	P15[4]		HiZ Analog Unb
GPIO [unused]	P1[6]		HiZ Analog Unb
GPIO [unused]	P2[6]		HiZ Analog Unb
GPIO [unused]	P1[5]		HiZ Analog Unb
GPIO [unused]	P2[7]		HiZ Analog Unb
GPIO [unused]	P1[2]		HiZ Analog Unb
GPIO [unused]	P15[0]		HiZ Analog Unb
GPIO [unused]	P3[1]		HiZ Analog Unb
GPIO [unused]	P3[0]		HiZ Analog Unb
GPIO [unused]	P15[1]		HiZ Analog Unb
GPIO [unused]	P3[2]		HiZ Analog Unb
PWM_Out	P2[0]	Dgtl Out	HiZ Analog Unb
Rx_1	P12[6]	Dgtl In	HiZ Analog Unb
SIO [unused]	P12[5]		HiZ Analog Unb
SIO [unused]	P12[2]		HiZ Analog Unb
SIO [unused]	P12[1]		HiZ Analog Unb
SIO [unused]	P12[3]		HiZ Analog Unb
SIO [unused]	P12[4]		HiZ Analog Unb
SIO [unused]	P12[0]		HiZ Analog Unb
Tx_1	P12[7]	Dgtl Out	HiZ Analog Unb
Name	**Port**	**Type**	**Reset State**
USB IO [unused]	P15[7]		HiZ Analog Unb
USB IO [unused]	P15[6]		HiZ Analog Unb
Vin	P0[0]	Analog	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	False
Enable Fast IMO During Startup	True
Unused Bonded IO	Allow but warn
Heap Size (bytes)	0x0800
Stack Size (bytes)	0x0800
Include CMSIS Core Peripheral Library Files	True

3.2 System Debug Settings

Table 7. System Debug Settings

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

3.3 System Operating Conditions

Table 8. System Operating Conditions

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

4 Clocks

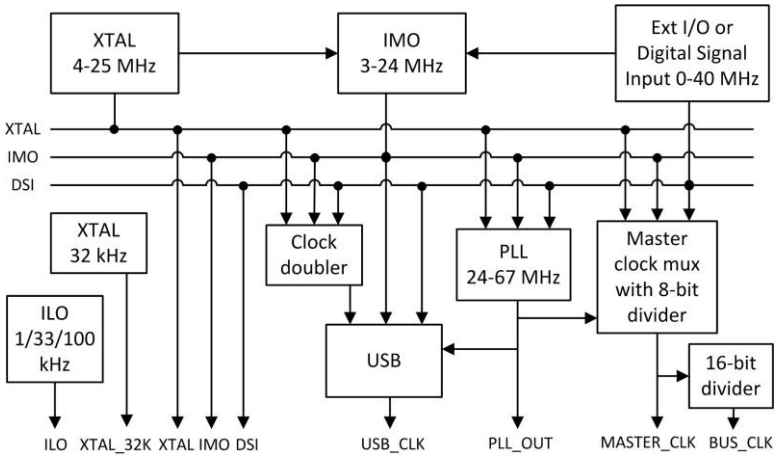
4 Clocks

The clock system includes these clock resources:

- Four internal clock sources increase system integration:

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

Figure 3. System Clock Configuration



4.1 System Clocks

Table 9 lists the system clocks used in this design.

Table 9. System Clocks

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

4.2 Local and Design Wide Clocks

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

Figure 4. Local and Design Wide Clock Configuration

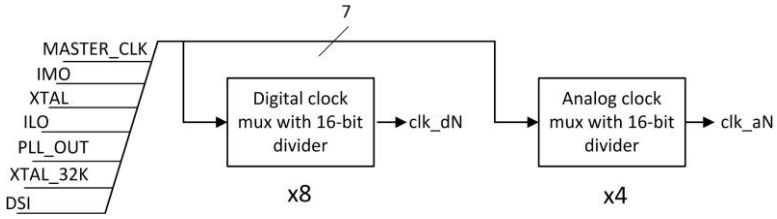


Table 10 lists the design wide clocks used in this design.

Table 10. Design Wide Clocks

Name	Domain	Source	Desired Freq	Nominal Freq	Accuracy (%)	Start at Reset	Enabled
Clock_1	DIGITAL	MASTER_CLK	? MHz	1 kHz	±1	True	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
CapSense_Clock_tmp	DIGITAL	BUS_CLK	? MHz	24 MHz	±1	True	True
ADC_DelSig_1_Ext_CP_Clk	DIGITAL	MASTER_CLK	? MHz	24 MHz	±1	True	True
CapSense_IntClock	DIGITAL	MASTER_CLK	12 MHz	12 MHz	±1	True	True

Name	Domain	Source	Desired Freq	Nominal Freq	Accuracy (%)	Start at Reset	Enabled
Clock_2	DIGITAL	MASTER_CLK	1 MHz	1 MHz	±1	True	True
UART_1_-IntClock	DIGITAL	MASTER_CLK	460.8 kHz	461.538 kHz	±1	True	True
ADC_DelSig_1_theACLK	ANALOG	MASTER_CLK	128 kHz	127.66 kHz	±1	True	True
timer_clock	DIGITAL	Clock_1	? MHz	1 kHz	±1	True	True

For more information on clocking resources, please refer to:

- Clocking System chapter in the PSoC 5LP Technical Reference Manual
- Clocking chapter in the System Reference Guide ○ 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	Intr Num	Vector	Priority
PWM_ctrl	0	0	7
CapSense_IsrCH0	1	1	7
isr_Timer_1	2	2	1
ADC_DelSig_1_IRQ	29	29	0

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:

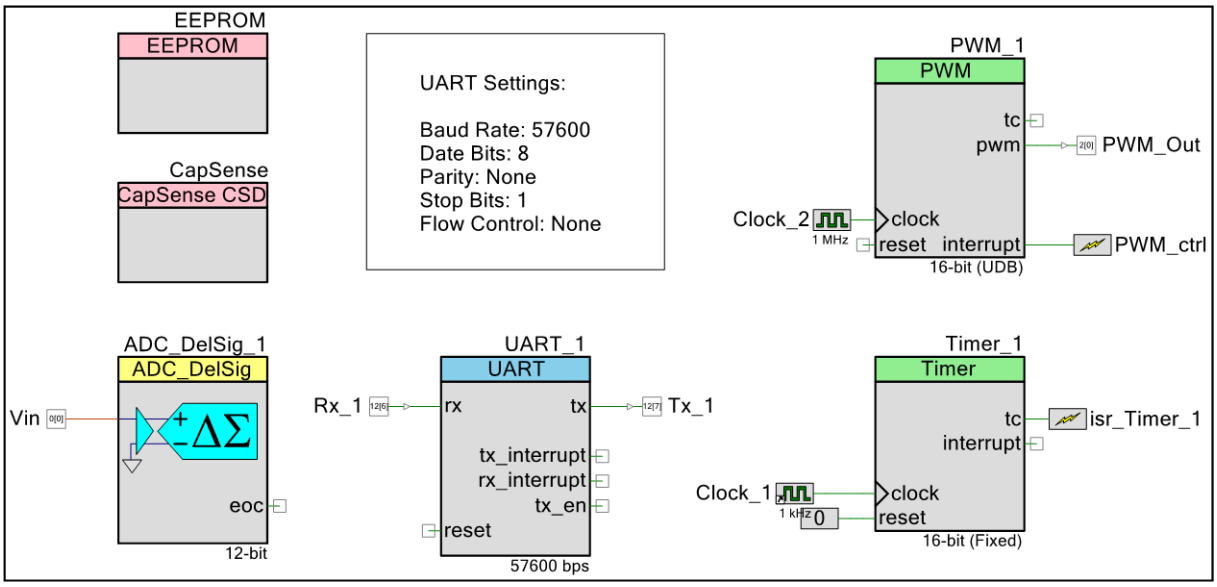
- Flash Protection chapter in the PSoC 5LP Technical Reference Manual
- Flash and EEPROM chapter in the System Reference Guide ○ CyWrite API routines ○ CyFlash API routines

7 Design Contents

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

7.1 Schematic Sheet: Page 1

Figure 5. Schematic Sheet: Page 1



This schematic sheet contains the following component instances:

- Instance ADC_DelSig_1 (type: ADC_DelSig_v3_30)
- Instance CapSense (type: CapSense_CSD_v3_50)
- Instance EEPROM (type: EEPROM_v3_0)
- Instance PWM_1 (type: PWM_v3_30)
- Instance Timer_1 (type: Timer_v2_80)

- Instance UART_1 (type: UART_v2_50)

8 Components

8.1 Component type: ADC_DelSig [v3.30]

8.1.1 Instance ADC_DelSig_1

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

Table 14. Component Parameters for ADC_DelSig_1

Parameter Name	Value	Description
ADC_Alignment	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Alignment_Config2	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Alignment_Config3	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Alignment_Config4	Right	This parameter determines how the result is aligned in the 24 bit result word.
ADC_Charge_Pump_Clock	false	Low power charge pump clock selection
ADC_Clock	Internal	Parameter for selecting the ADC clock type.
ADC_Input_Mode	Single	Differential or Single ended input mode
ADC_Input_Range	Vssa to Vdda	Choose input operating mode that best supports the range of the signals being measured.
ADC_Input_Range_Config2	-Input +/- Vref	Choose input operating mode that best supports the range of the signals being measured.
ADC_Input_Range_Config3	-Input +/- Vref	Choose input operating mode that best supports the range of the signals being measured.
ADC_Input_Range_Config4	-Input +/- Vref	Choose input operating mode that best supports the range of the signals being measured.
ADC_Power	Medium Power	Sets power level of ADC.
ADC_Reference	Internal Vdda/4	Selects voltage reference source and configuration.
ADC_Reference_Config2	Internal 1.024 Volts	Selects voltage reference source and configuration.
ADC_Reference_Config3	Internal 1.024 Volts	Selects voltage reference source and configuration.
ADC_Reference_Config4	Internal 1.024 Volts	Selects voltage reference source and configuration.
ADC_Resolution	12	ADC Resolution in bits

ADC_Resolution_Config2	16	ADC Resolution in bits
ADC_Resolution_Config3	16	ADC Resolution in bits
ADC_Resolution_Config4	16	ADC Resolution in bits

Parameter Name	Value	Description
Clock_Frequency	64000	Determines the ADC clock frequency.
Comment_Config1	Default Config	Parameter which holds the user comment for the config1.
Comment_Config2	Second Config	Parameter which holds the user comment for the config2.
Comment_Config3	Third Config	Parameter which holds the user comment for the config3.
Comment_Config4	Fourth Config	Parameter which holds the user comment for the config4.
Config1_Name	CFG1	This parameter is used to create constants in the header file for config 1.
Config2_Name	CFG2	This parameter is used to create constants in the header file for config 2.
Config3_Name	CFG3	This parameter is used to create constants in the header file for config 3.
Config4_Name	CFG4	This parameter is used to create constants in the header file for config 4.
Configs	1	Number of active configurations
Conversion_Mode	2 - Continuous	ADC conversion mode
Conversion_Mode_Config2	2 - Continuous	ADC conversion mode
Conversion_Mode_Config3	2 - Continuous	ADC conversion mode
Conversion_Mode_Config4	2 - Continuous	ADC conversion mode
Enable_Vref_Vss	false	Determines whether or not to connect ADC's reference Vssa to AGL[6].
EnableModulatorInput	false	When this parameter is enabled, the modulator input terminal will be enabled on the symbol.
Input_Buffer_Gain	1	Gain of input amplifier
Input_Buffer_Gain_Config2	1	Gain of input amplifier
Input_Buffer_Gain_Config3	1	Gain of input amplifier
Input_Buffer_Gain_Config4	1	Gain of input amplifier
Input_Buffer_Mode	Rail to Rail	Buffer Mode type selection
Input_Buffer_Mode_Config2	Rail to Rail	Buffer Mode type selection
Input_Buffer_Mode_Config3	Rail to Rail	Buffer Mode type selection
Input_Buffer_Mode_Config4	Rail to Rail	Buffer Mode type selection
Ref_Voltage	1.25	Set reference voltage
Ref_Voltage_Config2	1.024	Set reference voltage
Ref_Voltage_Config3	1.024	Set reference voltage
Ref_Voltage_Config4	1.024	Set reference voltage
rm_int	false	Removes internal interrupt (IRQ)

Sample_Rate	4000	Sample Rate in Hz
Sample_Rate_Config2	10000	Sample Rate in Hz
Sample_Rate_Config3	10000	Sample Rate in Hz
Sample_Rate_Config4	10000	Sample Rate in Hz
Start_of_Conversion	Software	Continuous conversions or hardware controlled
User Comments		Instance-specific comments.

8

8.2 Component type: CapSense_CSD [v3.50]

8.2.1 Instance CapSense

Description: The CapSense CSD component provides capacitive buttons, sliders, touch pads and proximity sensors
Instance type: CapSense_CSD [v3.50]
Datasheet: online component datasheet for CapSense_CSD

Table 15. Component Parameters for CapSense

Parameter Name	Value	Description
AnalogSwitchDivider	11	Defines value of analog switch divider for all capsense system.
ClockSource	Internal	Defines the clock source of component.
ConnectInactiveSensors	Ground	Defines the sensor inactive state.
CurrentSource	IDAC Sourcing	Defines IDAC configurations. The CSD IDAC Source - set IDAC polarity to Source, for CSD IDAC Sink - to Sink and for CSD Rb - Remove IDAC. The CSA scanning method do not use this parameter.
EnableTuneHelper	false	Allows generation of tuner APIs.
EzI2CInstanceName	EZI2C	Default instance name of Tuner communication component.
GuardSensorEnable	false	Enables Guard sensor. This type of sensor typically required for water proof applications.
IdacRange	32 uA	Defines Idac Range for all sensors.
Implementation_CH0	UDB mult	Selects implementation type of channel 0.
Implementation_CH1	UDB mult	Selects implementation type of channel 1.
IntClockFrequency	12	Selects internal clock frequency.
LowBaselineReset	5	Defines the number samples with raw counts less than baseline needed to make baseline snap down to the raw count level.

MultipleAnalogSwitchDivider	true	Defines the Analog Switch Divider usage. If true each scan slot will use dedicated Analog Switch Divider value, in other case – all sensors will use only one Analog Switch Divider value.
NegativeNoiseThreshold	20	Defines the negative difference between the rawcount and baseline levels for baseline resting to the rawcaount level.
NumberOfChannels	1	Defines number of channels.
PrescalerOptions	UDB	Defines prescaler implementation type.
Parameter Name	**Value**	**Description**
PrsOptions	Enabled 16 bits, full speed	Defines PRS as None, PRS8, PRS16 and PRS16 with time multiplexed.
RawDataFilterType	None	Defines filter applied to raw data values.
RbNumber_CH0	0	Defines total value of bleed resistor for channel 0.
RbNumber_CH1	0	Defines total value of bleed resistor for channel 1.
ScanSpeed	Normal	Defines ScanSpeed value for all capsense system.
SensorAutoReset	false	Enabling auto reset causes baseline to always update regardless of whether the difference counts are above or below the noise threshold. When auto reset is disabled, Baseline only updates when difference counts are within the plus/minus noise threshold (the noise threshold is mirrored).
SensorNumber_CH0	1	Total Sensors Count for channel 0.
SensorNumber_CH1	0	Total Sensors Count for channel 1.
ShieldEnable	false	Defines visibity for shield output.
TunerProperties		Contains additional parameters required for tuner
TuningMethod	None	Defines tuning method for all capsense system.
User Comments		Instance-specific comments.
VrefOptions	1.024V	Selects reference source for all capsense system. The 1.2V does not supported yet.
VrefValue	64	Defines value of data register for VDAC reference source. This value is proportional to VDAC range.

WaterProofingEnabled	false	Enables special capsense system settings to use in water proof designs.
WidgetResolution	8	Defines Signal resolution as uint8 or uint16. Valid values are 8 and 16.

8.3 Component type: EEPROM [v3.0]

8.3.1 Instance EEPROM

Parameter Name	Value	Description

Description:	User Comments		Instance-specific comments.

Provides an API to Erase and Write EEPROM. Instance type: EEPROM [v3.0]
Datasheet: **online component datasheet for EEPROM**

Table 16. Component Parameters for EEPROM
8

8.4 Component type: PWM [v3.30]

8.4.1 Instance PWM_1

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

Table 17. 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	Less	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	1499	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	true	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	19999	Defines the PWM period value
Parameter Name	**Value**	**Description**
PWMMode	One Output	Defines the overall mode of the PWM
Resolution	16	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
User Comments		Instance-specific comments.

8.5 Component type: Timer [v2.80]

8.5.1 Instance Timer_1

Description: 8, 16, 24 or 32-bit Timer
Instance type: Timer [v2.80]
Datasheet: online component datasheet for Timer

Table 18. 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.
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.

8

Parameter Name	Value	Description
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	998	Defines the timer period (This is also the reload value when terminal count is reached)
Resolution	16	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
User Comments		Instance-specific comments.

8.6 Component type: UART [v2.50]

8.6.1 Instance UART_1

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

Table 19. 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	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	true	Enables the external TX enable signal output.
InternalClock	true	Enables the internal clock. This parameter removes the clock input pin.
InterruptOnTXComplete	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX complete' event.
InterruptOnTXFifoEmpty	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX FIFO empty' event.
InterruptOnTXFifoFull	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX FIFO full' event.
InterruptOnTXFifoNotFull	false	This is an Interrupt mask used to enable/disable the interrupt on 'TX FIFO not full' event.
IntOnAddressDetect	false	Enables the interrupt on hardware address detected event by default
IntOnAddressMatch	false	Enables the interrupt on hardware address match detected event by default
IntOnBreak	false	Enables the interrupt on break signal detected event by default
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

8

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.
User Comments		Instance-specific comments.

9 Other Resources

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

- Standard Types and Defines chapter in the System Reference Guide ○ Software base types ○ Hardware register types ○ Compiler defines ○ Cypress API return codes ○ Interrupt types and macros
- Registers ○ The full PSoC 5LP register map is covered in the PSoC 5LP Registers Technical Reference Manual ○ Register Access chapter in the System Reference Guide
  - CY_GET API routines
  - CY_SET API routines
- System Functions chapter in the System Reference Guide ○ General API routines ○ CyDelay API routines ○ CyVd Voltage Detect API routines
- Power Management ○ Power Supply and Monitoring chapter in the PSoC 5LP Technical Reference Manual ○ Low Power Modes chapter in the PSoC 5LP Technical Reference Manual ○ Power Management chapter in the System Reference Guide
  - CyPm API routines
- Watchdog Timer chapter in the System Reference Guide ○ CyWdt API routines
- Cache Management ○ Cache Controller chapter in the PSoC 5LP Technical Reference Manual ○ Cache chapter in the System Reference Guide
  - CyFlushCache() API routine

10 Hobbyist Build & Setup Guide

This chapter provides a step-by-step guide for hobbyists to build, program, and start using the PSoC5 Greenhouse Automation Device.

10.1 Materials Required

- PSoC5LP Development Kit (CY8C5888LTI-LP097)
- USB A-to-mini B cable
- Twisted pair wires
- Analog temperature sensor (compatible with ADC input, e.g., LM35)
- Servo
- Serial Terminal (e.g. PuTTY)
- Windows PC with PSoC Creator
- Breadboard and jumper wires
- 5V regulated power supply

10.2 Hardware Assembly

1. **Connect the twisted pair wires:**
    - Connect one end on P1[7] and the other on GND
2. **Connect the Temperature Sensor (e.g., LM35):**
    - GND and VDD as appropriate
    - P0[0] for input
3. **UART Pins:**
    - Tx (P12[7]) to USB-UART converter Rx
    - Rx (P12[6]) to USB-UART converter Tx
4. **PWM Output (P2[0])** for servo control.

10.3 Programming the Device

1. Install **PSoC Creator**
2. Open the johannesPSoC5GreenhouseAutomation project.
3. Compile the project (Build → Build Project).
4. Connect the PSoC5
5. Go to Debug → Program

10.4 Verifying Operation

1. Open your serial terminal emulator (Putty).
2. Set baud rate to **57600**, 8 data bits, no parity, 1 stop bit.
3. Connect to the COM port of the device. (See from “Device Manager” -> Ports (COM & LPT) -> KitProg USB-UART (Here is the COM port))
4. Open the connection, type ? and press **Enter** to check if the device responds with: Greenhouse controller #1

11 Command Interface Guide

This chapter outlines the serial command interface to interact with the greenhouse controller.

11.1 Supported Commands

Command	Description	Example / Output
?	Prints device name	Greenhouse controller #1
T hh:mm	Sets the current time (24h format)	T 14:30
T?	Gets the current time	14:30
D dd/mm/yyyy	Sets the current date	D 29/05/2025
D?	Gets the current date	29/05/2025
A	Reads all saved data in JSON format	See below
C	Clears the saved data memory	Outputs “Memory cleared” if successful

11.1 Example Output for A

```
[
  {
    "date": "29/05/2025",
    "Time": "14:30",
    "temperature": 25.0,
    "soil_moisture": 70
  }
]
```