**Please note that Cypress is an Infineon Technologies Company.**

The document following this cover page is marked as "Cypress" document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

**Continuity of document content**

The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

**Continuity of ordering part numbers**

Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

www.infineon.com

# Programmable System-on-Chip (PSoC®)

## General Description

PSoC® 5LP is a true programmable embedded system-on-chip, integrating configurable analog and digital peripherals, memory, and a microcontroller on a single chip. The PSoC 5LP architecture boosts performance through:

- 32-bit Arm® Cortex®-M3 core plus DMA controller at up to 80 MHz
- Ultra low power with industry's widest voltage range
- Programmable digital and analog peripherals enable custom functions
- Flexible routing of any analog or digital peripheral function to any pin

PSoC devices employ a highly configurable system-on-chip architecture for embedded control design. They integrate configurable analog and digital circuits, controlled by an on-chip microcontroller. A single PSoC device can integrate as many as 100 digital and analog peripheral functions, reducing design time, board space, power consumption, and system cost while improving system quality.

## Features

- Operating characteristics
  - Voltage range: 1.71 to 5.5 V, up to 6 power domains
  - Temperature range (ambient) –40 to 85 °C[1]
  - DC to 80-MHz operation
  - Power modes
    - Active mode 3.1 mA at 6 MHz, and 15.4 mA at 48 MHz
    - 2-µA sleep mode
    - 300-nA hibernate mode with RAM retention
  - Boost regulator from 0.5-V input up to 5-V output

- Performance
  - 32-bit Arm Cortex-M3 CPU, 32 interrupt inputs
  - 24-channel direct memory access (DMA) controller

- Memories
  - Up to 256 KB program flash, with cache and security features
  - Up to 32 KB additional flash for error correcting code (ECC)
  - Up to 64 KB RAM
  - 2 KB EEPROM

- Digital peripherals
  - Four 16-bit timer, counter, and PWM (TCPWM) blocks
  - I$^2$C, 1 Mbps bus speed
  - USB 2.0 certified Full-Speed (FS) 12 Mbps peripheral interface (TID#10840032) using internal oscillator[2]
  - 20 to 24 universal digital blocks (UDB), programmable to create any number of functions:
    - 8-, 16-, 24-, and 32-bit timers, counters, and PWMs
    - I$^2$C, UART, SPI, I2S, LIN 2.0 interfaces
    - Cyclic redundancy check (CRC)
    - Pseudo random sequence (PRS) generators
    - Quadrature decoders
    - Gate-level logic functions

- Programmable clocking
  - 3- to 74-MHz internal oscillator, 2% accuracy at 3 MHz
  - 4- to 25-MHz external crystal oscillator
  - Internal PLL clock generation up to 80 MHz
  - Low-power internal oscillator at 1, 33, and 100 kHz
  - 32.768-kHz external watch crystal oscillator
  - 12 clock dividers routable to any peripheral or I/O

- Analog peripherals
  - 12-bit SAR ADC
  - 8-bit DAC
  - Two comparators
  - CapSense® support, up to 62 sensors
  - 1.024 V ±1% internal voltage reference

- Versatile I/O system
  - 46 to 72 I/O pins – up to 62 general-purpose I/Os (GPIOs)
  - Up to eight performance I/O (SIO) pins
    - 25 mA current sink
    - Programmable input threshold and output high voltages
    - Can act as a general-purpose comparator
    - Hot swap capability and overvoltage tolerance
  - Two USBIO pins that can be used as GPIOs
  - Route any digital or analog peripheral to any GPIO
  - LCD direct drive from any GPIO, up to 46 × 16 segments
  - CapSense support from any GPIO
  - 1.2-V to 5.5-V interface voltages, up to four power domains

- Programming, debug, and trace
  - JTAG (4-wire), serial wire debug (SWD) (2-wire), single wire viewer (SWV), and Traceport (5-wire) interfaces
  - Arm debug and trace modules embedded in the CPU core
  - Bootloader programming through I$^2$C, SPI, UART, USB, and other interfaces

- Package options: 68-pin QFN, 100-pin TQFP, and 99-pin CSP

- Development support with free PSoC Creator™ tool
  - Schematic and firmware design support
  - Over 100 PSoC Components™ integrate multiple ICs and system interfaces into one PSoC. Components are free embedded ICs represented by icons. Drag and drop component icons to design systems in PSoC Creator.
  - Includes free GCC compiler, supports Keil/Arm MDK compiler
  - Supports device programming and debugging

---

**Notes**
1. The maximum storage temperature is 150 °C in compliance with JEDEC Standard JESD22-A103, High Temperature Storage Life.
2. This feature on select devices only. See Ordering Information on page 104 for details.

---

## More Information

Cypress provides a wealth of data at www.cypress.com to help you to select the right PSoC device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the knowledge base article KBA86521, How to Design with PSoC 3, PSoC 4, and PSoC 5LP. Following is an abbreviated list for PSoC 5LP:
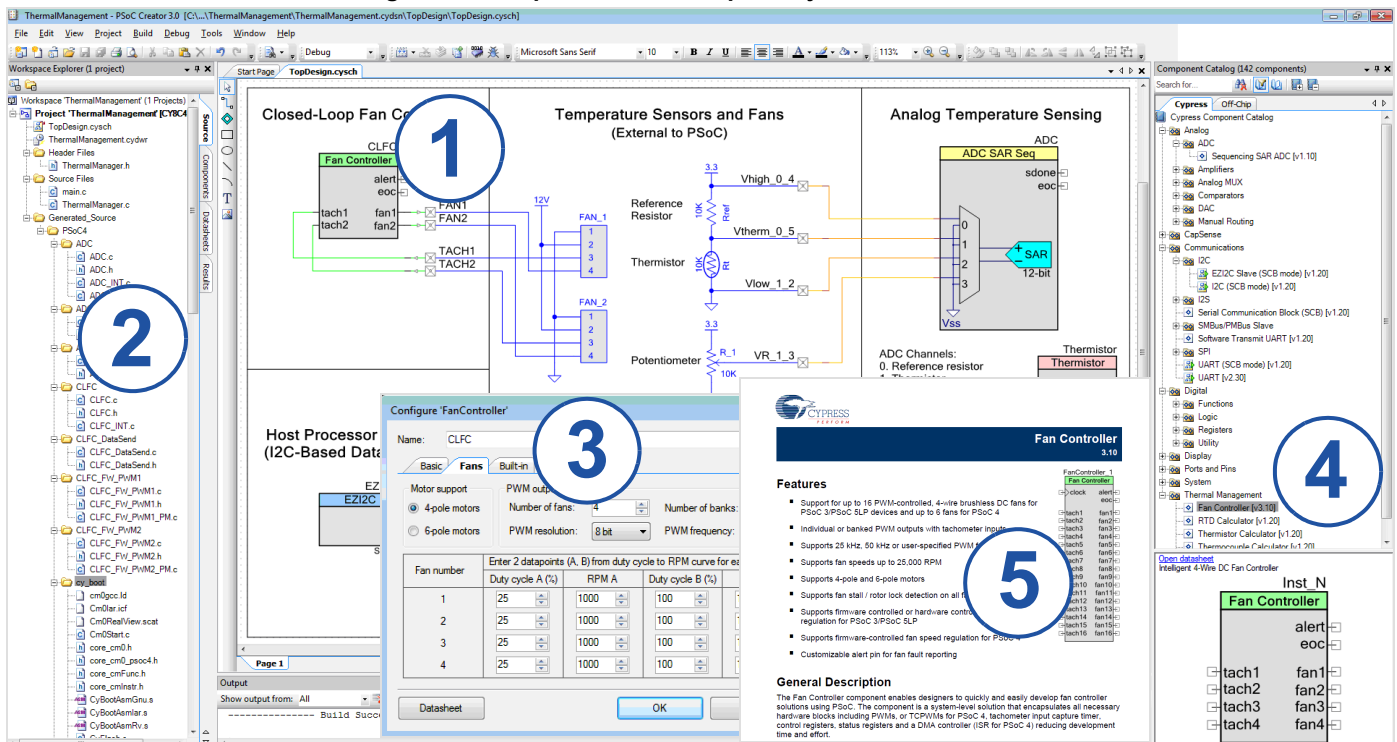
- Overview: PSoC Portfolio, PSoC Roadmap
- Product Selectors: PSoC 1, PSoC 3, PSoC 4, PSoC 5LP
  In addition, PSoC Creator includes a device selection tool.
- Application notes: Cypress offers a large number of PSoC application notes and code examples covering a broad range of topics, from basic to advanced level. Recommended application notes for getting started with PSoC 5LP are:
  □ AN77759: Getting Started With PSoC 5LP
  □ AN77835: PSoC 3 to PSoC 5LP Migration Guide
  □ AN61290: Hardware Design Considerations
  □ AN57821: Mixed Signal Circuit Board Layout
  □ AN58304: Pin Selection for Analog Designs
  □ AN81623: Digital Design Best Practices
  □ AN73854: Introduction To Bootloaders

- Development Kits:
  □ CY8CKIT-059 is a low-cost platform for prototyping, with a unique snap-away programmer and debugger on the USB connector.
  □ CY8CKIT-050 is designed for analog performance, for developing high-precision analog, low-power, and low-voltage applications.
  □ CY8CKIT-001 provides a common development platform for any one of the PSoC 1, PSoC 3, PSoC 4, or PSoC 5LP families of devices.
  □ The MiniProg3 device provides an interface for flash programming and debug.
- Technical Reference Manuals (TRM)
  □ Architecture TRM
  □ Registers TRM
- Programming Specification

## PSoC Creator

PSoC Creator is a free Windows-based Integrated Design Environment (IDE). It enables concurrent hardware and firmware design of PSoC 3, PSoC 4, and PSoC 5LP based systems. Create designs using classic, familiar schematic capture supported by over 100 pre-verified, production-ready PSoC Components; see the list of component datasheets. With PSoC Creator, you can:

1. Drag and drop component icons to build your hardware system design in the main design workspace
2. Codesign your application firmware with the PSoC hardware, using the PSoC Creator IDE C compiler
3. Configure components using the configuration tools
4. Explore the library of 100+ components
5. Review component datasheets

**Figure 1. Multiple-Sensor Example Project in PSoC Creator**

# Contents

# 1. Architectural Overview

Introducing the CY8C52LP family of ultra low power, flash Programmable System-on-Chip (PSoC) devices, part of a scalable 8-bit PSoC 3 and 32-bit PSoC 5LP platform. The CY8C52LP family provides configurable blocks of analog, digital, and interconnect circuitry around a CPU subsystem. The combination of a CPU with a flexible analog subsystem, digital subsystem, routing, and I/O enables a high level of integration in a wide variety of consumer, industrial, and medical applications.

**Figure 1-1. Simplified Block Diagram**
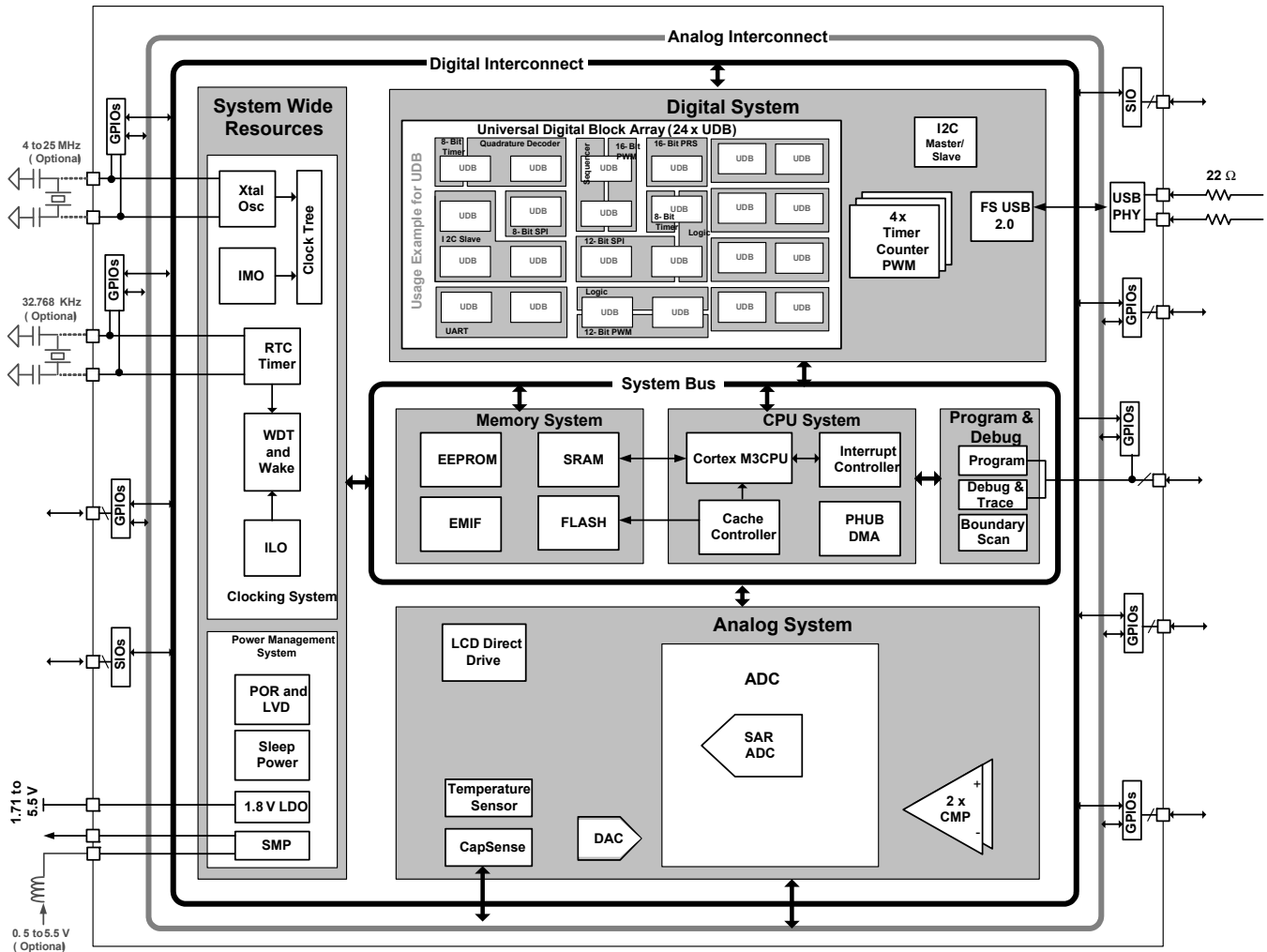


Figure 1-1 illustrates the major components of the CY8C52LP family. They are:

- Arm Cortex-M3 CPU subsystem
- Nonvolatile subsystem
- Programming, debug, and test subsystem
- Inputs and outputs
- Clocking
- Power
- Digital subsystem
- Analog subsystem

PSoC's digital subsystem provides half of its unique configurability. It connects a digital signal from any peripheral to any pin through the digital system interconnect (DSI). It also provides functional flexibility through an array of small, fast, low power UDBs. PSoC Creator provides a library of pre-built and tested standard digital peripherals (UART, SPI, LIN, PRS, CRC, timer, counter, PWM, AND, OR, and so on) that are mapped to the UDB array. You can also easily create a digital circuit using boolean primitives by means of graphical design entry. Each UDB contains programmable array logic (PAL)/programmable logic device (PLD) functionality, together with a small state machine engine to support a wide variety of peripherals.

In addition to the flexibility of the UDB array, PSoC also provides configurable digital blocks targeted at specific functions. For the CY8C52LP family these blocks can include four 16-bit timers, counters, and PWM blocks; I²C slave, master, and multimaster; Full-Speed USB.

For more details on the peripherals see the "Example Peripherals" section on page 39 of this datasheet. For information on UDBs, DSI, and other digital blocks, see the "Digital Subsystem" section on page 39 of this datasheet.

PSoC's analog subsystem is the second half of its unique configurability. All analog performance is based on a highly accurate absolute voltage reference with less than 1% error over temperature and voltage. The configurable analog subsystem includes:

- Analog muxes
- Comparators
- Voltage references
- ADC
- DAC

All GPIO pins can route analog signals into and out of the device using the internal analog bus. This allows the device to interface up to 62 discrete analog signals.

The CY8C52LP family offers a SAR ADC. Featuring 12-bit conversions at up to 1 M samples per second, it also offers low nonlinearity and offset errors and SNR better than 70 dB. It is well suited for a variety of higher speed analog applications.

A high-speed voltage or current DAC supports 8-bit output signals at an update rate of 8 Msps in IDAC and 1 Msps in VDAC. It can be routed out of any GPIO pin. You can create higher resolution voltage PWM DAC outputs using the UDB array. This can be used to create a pulse width modulated (PWM) DAC of up to 10 bits, at up to 48 kHz. The digital DACs in each UDB support PWM, PRS, or delta-sigma algorithms with programmable widths.

In addition to the ADC and DAC, the analog subsystem provides multiple comparators. See the "Analog Subsystem" section on page 49 of this datasheet for more details.

PSoC's CPU subsystem is built around a 32-bit three-stage pipelined ARM Cortex-M3 processor running at up to 80 MHz. The Cortex-M3 includes a tightly integrated nested vectored interrupt controller (NVIC) and various debug and trace modules. The overall CPU subsystem includes a DMA controller, flash cache, and RAM. The NVIC provides low latency, nested interrupts, and tail-chaining of interrupts and other features to increase the efficiency of interrupt handling. The DMA controller enables peripherals to exchange data without CPU involvement. This allows the CPU to run slower (saving power) or use those CPU cycles to improve the performance of firmware algorithms. The flash cache also reduces system power consumption by allowing less frequent flash access.

PSoC's nonvolatile subsystem consists of flash, byte-writeable EEPROM, and nonvolatile configuration options. It provides up to 256 KB of on-chip flash. The CPU can reprogram individual blocks of flash, enabling boot loaders. You can enable an ECC for high-reliability applications. A powerful and flexible protection model secures your sensitive information, allowing selective memory block locking for read and write protection. Two KB of

byte-writable EEPROM is available on-chip to store application data. Additionally, selected configuration options such as boot speed and pin drive mode are stored in nonvolatile memory. This allows settings to activate immediately after power-on reset (POR).

The three types of PSoC I/O are extremely flexible. All I/Os have many drive modes that are set at POR. PSoC also provides up to four I/O voltage domains through the VDDIO pins. Every GPIO has analog I/O, LCD drive, CapSense, flexible interrupt generation, slew rate control, and digital I/O capability. The SIOs on PSoC allow $V_{OH}$ to be set independently of VDDIO when used as outputs. When SIOs are in input mode they are high impedance. This is true even when the device is not powered or when the pin voltage goes above the supply voltage. This makes the SIO ideally suited for use on an I²C bus where the PSoC may not be powered when other devices on the bus are. The SIO pins also have high current sink capability for applications such as LED drives. The programmable input threshold feature of the SIO can be used to make the SIO function as a general purpose analog comparator. For devices with Full-Speed USB, the USB physical interface is also provided (USBIO). When not using USB these pins may also be used for limited digital functionality and device programming. All the features of the PSoC I/Os are covered in detail in the "I/O System and Routing" section on page 32 of this datasheet.

The PSoC device incorporates flexible internal clock generators, designed for high stability and factory trimmed for high accuracy. The internal main oscillator (IMO) is the master clock base for the system and has 2% accuracy at 3 MHz. The IMO can be configured to run from 3 MHz up to 74 MHz. Multiple clock derivatives can be generated from the main clock frequency to meet application needs. The device provides a PLL to generate system clock frequencies up to 80 MHz from the IMO, external crystal, or external reference clock. It also contains a separate, very low-power internal low-speed oscillator (ILO) for the sleep and watchdog timers. A 32.768-kHz external watch crystal is also supported for use in RTC applications. The clocks, together with programmable clock dividers, provide the flexibility to integrate most timing requirements.

The CY8C52LP family supports a wide supply operating range from 1.71 to 5.5 V. This allows operation from regulated supplies such as 1.8 ± 5%, 2.5 V ±10%, 3.3 V ± 10%, or 5.0 V ± 10%, or directly from a wide range of battery types. In addition, it provides an integrated high-efficiency synchronous boost converter that can power the device from supply voltages as low as 0.5 V. This enables the device to be powered directly from a single battery. In addition, you can use the boost converter to generate other voltages required by the device, such as a 3.3 V supply for LCD glass drive. The boost's output is available on the $V_{BOOST}$ pin, allowing other devices in the application to be powered from the PSoC.

PSoC supports a wide range of low-power modes. These include a 300 nA hibernate mode with RAM retention and a 2-µA sleep mode with RTC. In the second mode the optional 32.768-kHz watch crystal runs continuously and maintains an accurate RTC.

Power to all major functional blocks, including the programmable digital and analog peripherals, can be controlled independently by firmware. This allows low-power background processing when some peripherals are not in use. This, in turn, provides a total device current of only 3.1 mA when the CPU is running at 6 MHz.

The details of the PSoC power modes are covered in the "Power System" section on page 26 of this datasheet.

PSoC uses JTAG (4-wire) or SWD (2-wire) interfaces for programming, debug, and test. Using these standard interfaces you can debug or program the PSoC with a variety of hardware solutions from Cypress or third party vendors. The Cortex-M3 debug and trace modules include FPB, DWT, ETM, and ITM. These modules have many features to help solve difficult debug and trace problems. Details of the programming, test, and debugging interfaces are discussed in the "Programming, Debug Interfaces, Resources" section on page 55 of this datasheet.

## 2. Pinouts

Each VDDIO pin powers a specific set of I/O pins. (The USBIOs are powered from VDDD.) Using the VDDIO pins, a single PSoC can support multiple voltage levels, reducing the need for off-chip level shifters. The black lines drawn on the pinout diagrams in Figure 2-3 and Figure 2-4, as well as Table 2-1, show the pins that are powered by each VDDIO.

Each VDDIO may source up to 100 mA total to its associated I/O pins, as shown in Figure 2-1.

**Figure 2-1. VDDIO Current Limit**
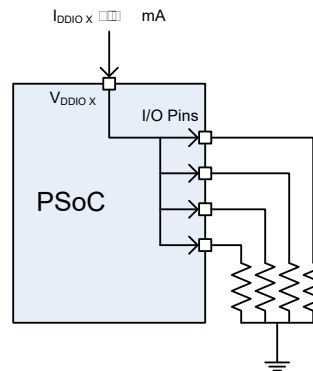


Conversely, for the 100-pin and 68-pin devices, the set of I/O pins associated with any VDDIO may sink up to 100 mA total, as shown in Figure 2-2.

**Figure 2-2. I/O Pins Current Limit**
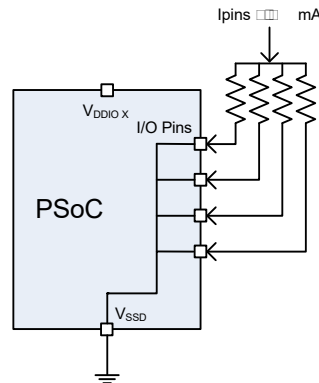
**Figure 2-3. 68-pin QFN Part Pinout**[3]
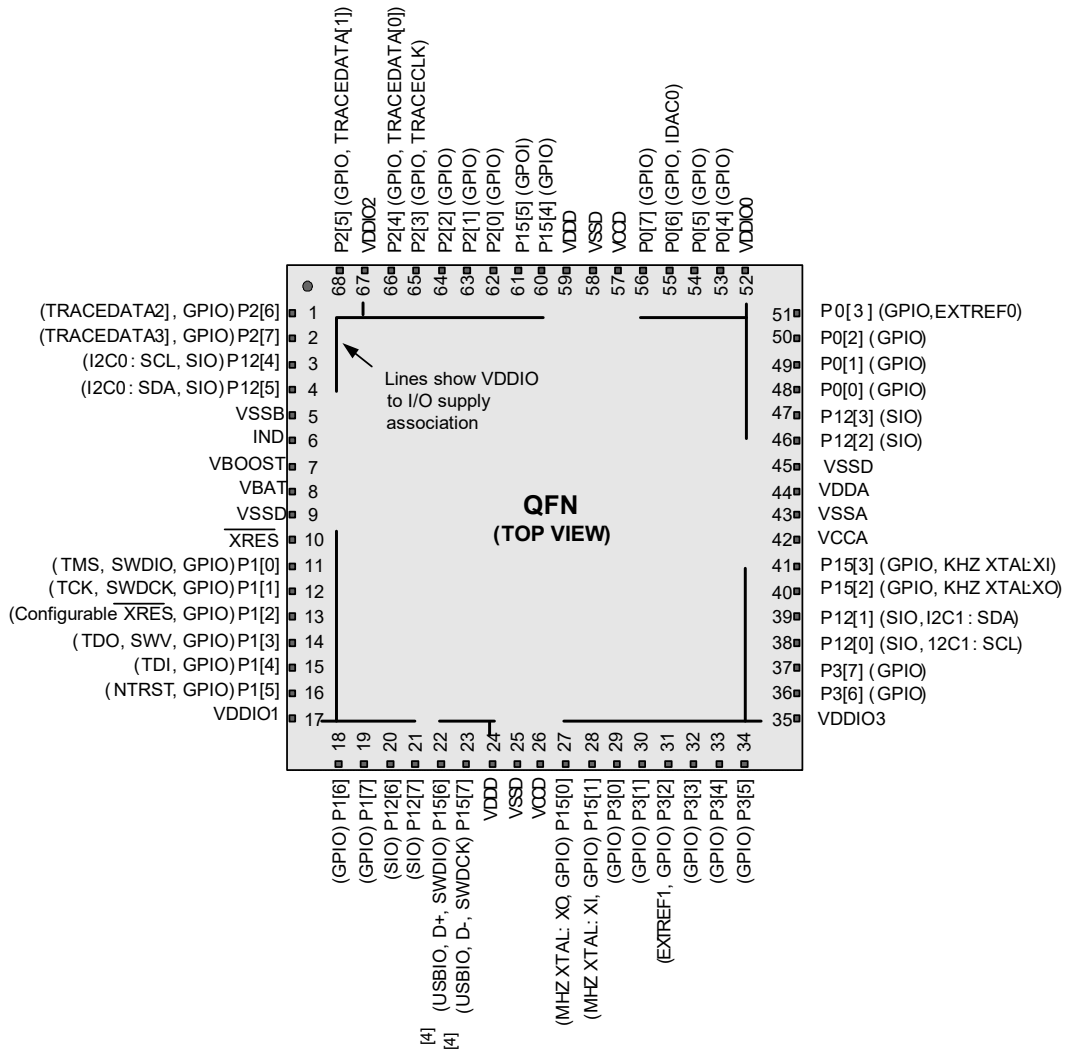


**Notes**
3. The center pad on the QFN package should be connected to digital ground (VSSD) for best mechanical, thermal, and electrical performance. If not connected to ground, it should be electrically floated and not connected to any other signal. For more information, see AN72845, Design Guidelines for QFN Devices.
4. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.

**Figure 2-4. 100-pin TQFP Part Pinout**
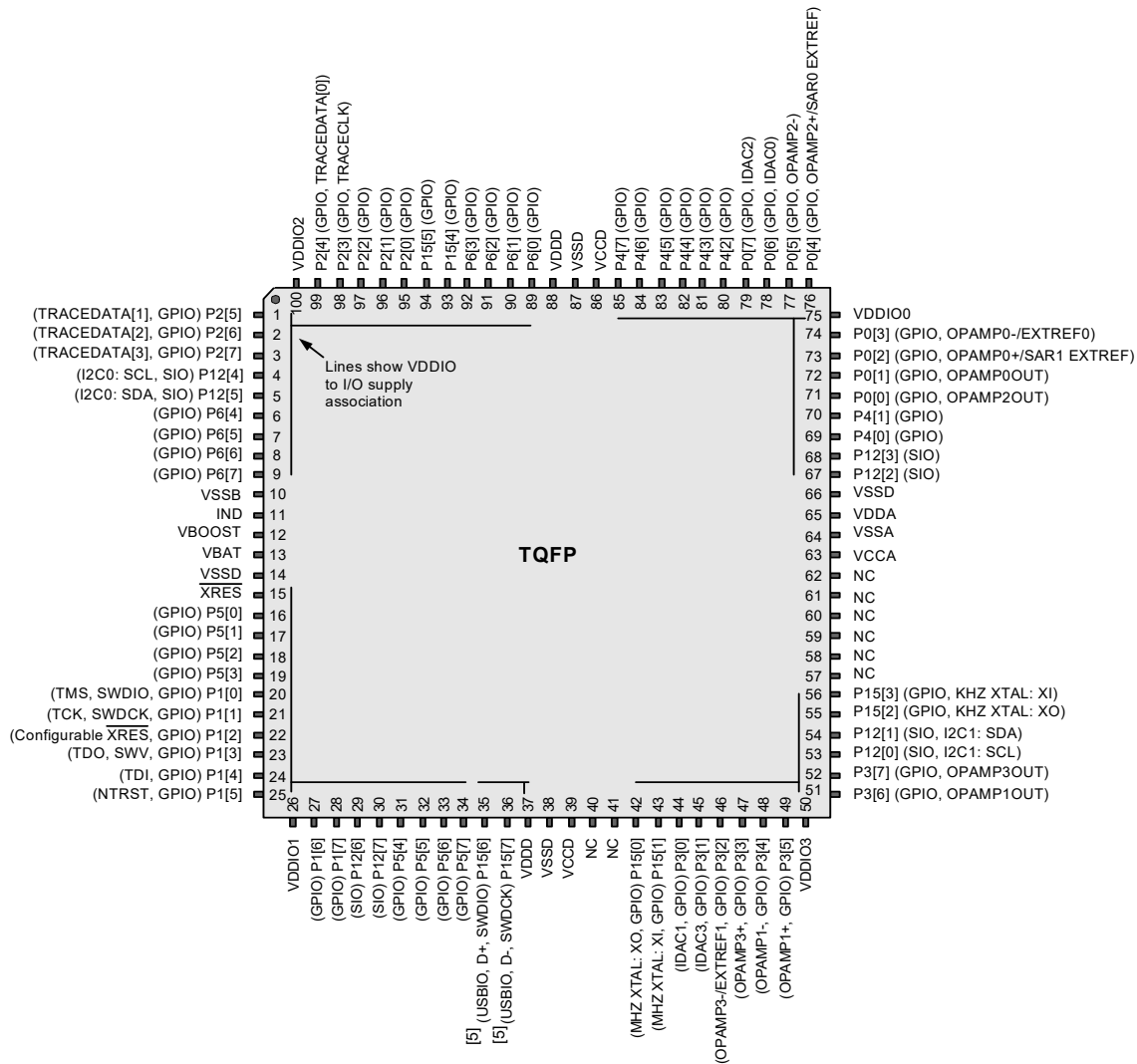


**Table 2-1. V_DDIO and Port Pin Associations**

| VDDIO | Port Pins |
|---|---|
| VDDIO0 | P0[7:0], P4[7:0], P12[3:2] |
| VDDIO1 | P1[7:0], P5[7:0], P12[7:6] |
| VDDIO2 | P2[7:0], P6[7:0], P12[5:4], P15[5:4] |
| VDDIO3 | P3[7:0], P12[1:0], P15[3:0] |
| VDDD | P15[7:6] (USB D+, D-) |

**Note**
5.  Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.

Table 2-2 shows the pinout for the 99-pin CSP package. Since there are four $V_{DDIO}$ pins, the set of I/O pins associated with any $V_{DDIO}$ may sink up to 100 mA total, same as for the 100-pin and 68-pin devices.

**Table 2-2. CSP Pinout**

| Ball | Name | Ball | Name | Ball | Name | Ball | Name |
|------|------|------|------|------|------|------|------|
| E5 | P2[5] | L2 | VIO1 | B2 | P3[6] | C8 | VIO0 |
| G6 | P2[6] | K2 | P1[6] | B3 | P3[7] | D7 | P0[4] |
| G5 | P2[7] | C9 | P4[2] | C3 | P12[0] | E7 | P0[5] |
| H6 | P12[4] | E8 | P4[3] | C4 | P12[1] | B9 | P0[6] |
| K7 | P12[5] | K1 | P1[7] | E3 | P15[2] | D8 | P0[7] |
| L8 | P6[4] | H2 | P12[6] | E4 | P15[3] | D9 | P4[4] |
| J6 | P6[5] | F4 | P12[7] | A1 | NC | F8 | P4[5] |
| H5 | P6[6] | J1 | P5[4] | A9 | NC | F7 | P4[6] |
| J5 | P6[7] | H1 | P5[5] | L1 | NC | E6 | P4[7] |
| L7 | VSSB | F3 | P5[6] | L9 | NC | E9 | VCCD |
| K6 | Ind | G1 | P5[7] | A3 | VCCA | F9 | VSSD |
| L6 | VBOOST | G2 | P15[6][6] | A4 | VSSA | G9 | VDDD |
| K5 | VBAT | F2 | P15[7][6] | B7 | VSSA | H9 | P6[0] |
| L5 | VSSD | E2 | VDDD | B8 | VSSA | G8 | P6[1] |
| L4 | XRES | F1 | VSSD | C7 | VSSA | H8 | P6[2] |
| J4 | P5[0] | E1 | VCCD | A5 | VDDA | J9 | P6[3] |
| K4 | P5[1] | D1 | P15[0] | A6 | VSSD | G7 | P15[4] |
| K3 | P5[2] | D2 | P15[1] | B5 | P12[2] | F6 | P15[5] |
| L3 | P5[3] | C1 | P3[0] | A7 | P12[3] | F5 | P2[0] |
| H4 | P1[0] | C2 | P3[1] | C5 | P4[0] | J7 | P2[1] |
| J3 | P1[1] | D3 | P3[2] | D5 | P4[1] | J8 | P2[2] |
| H3 | P1[2] | D4 | P3[3] | B6 | P0[0] | K9 | P2[3] |
| J2 | P1[3] | B4 | P3[4] | C6 | P0[1] | H7 | P2[4] |
| G4 | P1[4] | A2 | P3[5] | A8 | P0[2] | K8 | VIO2 |
| G3 | P1[5] | B1 | VIO3 | D6 | P0[3] | | |

Figure 2-5 and Figure 2-6 show an example schematic and an example PCB layout, for the 100-pin TQFP part, for optimal analog performance on a 2-layer board.

■ The two pins labeled VDDD must be connected together.

■ The two pins labeled VCCD must be connected together, with capacitance added, as shown in Figure 2-5 and Power System on page 26. The trace between the two VCCD pins should be as short as possible.

■ The two pins labeled VSSD must be connected together.

For information on circuit board layout issues for mixed signals, refer to the application note AN57821 - Mixed Signal Circuit Board Layout Considerations for PSoC® 3 and PSoC 5.

**Note**
6. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.

**Figure 2-5. Example Schematic for 100-pin TQFP Part with Power Connections**
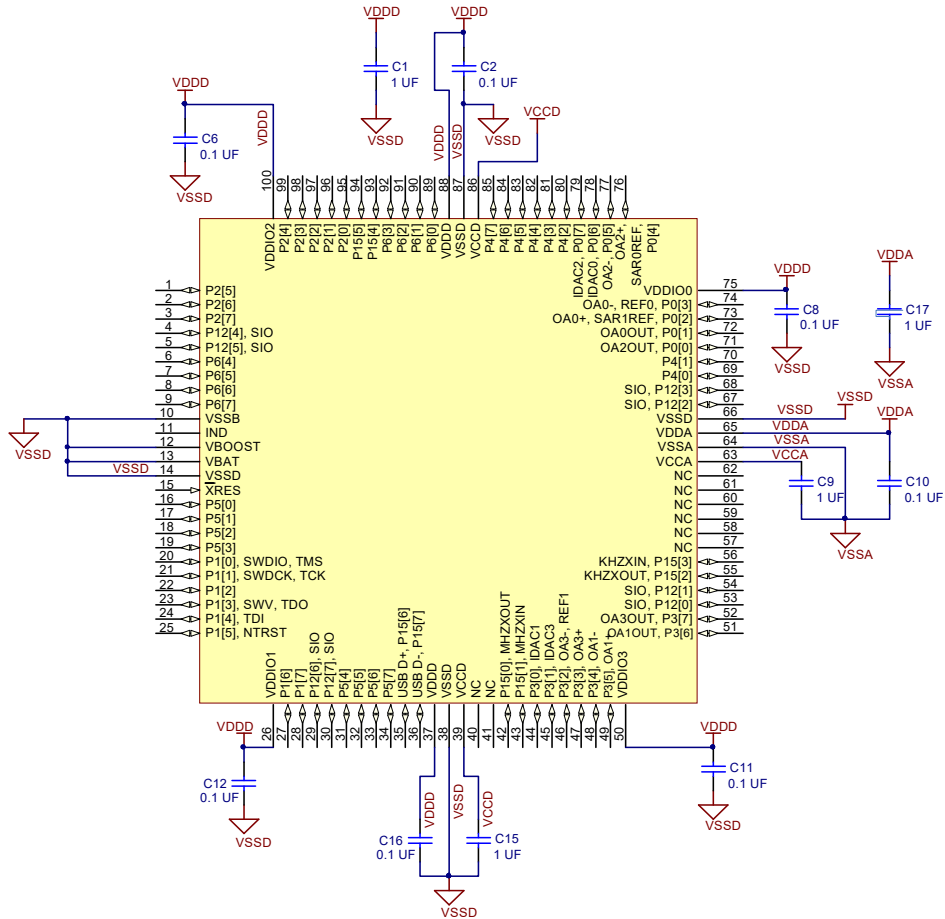


**Note** The two VCCD pins must be connected together with as short a trace as possible. A trace under the device is recommended, as shown in Figure 2-6.

For more information on pad layout, refer to http://www.cypress.com/cad-resources/psoc-5lp-cad-libraries.

**Figure 2-6. Example PCB Layout for 100-pin TQFP Part for Optimal Analog Performance**

# 3. Pin Descriptions

**IDAC0.** Low resistance output pin for high IDAC.
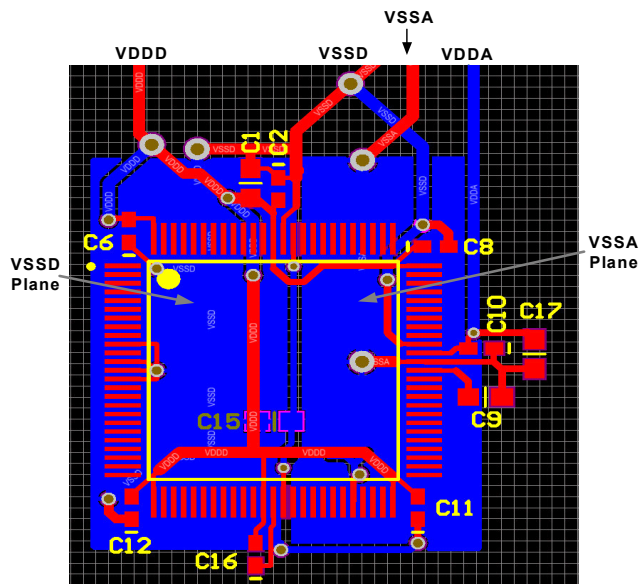
**Extref0, Extref1.** External reference input to the analog system.

**SAR0 EXTREF, SAR1 EXTREF.** External references for SAR ADCs

**GPIO.** General purpose I/O pin provides interfaces to the CPU, digital peripherals, analog peripherals, interrupts, LCD segment drive, and CapSense[7].

**I2C0: SCL, I2C1: SCL.** $I^2C$ SCL line providing wake from sleep on an address match. Any I/O pin can be used for $I^2C$ SCL if wake from sleep is not required.

**I2C0: SDA, I2C1: SDA.** $I^2C$ SDA line providing wake from sleep on an address match. Any I/O pin can be used for $I^2C$ SDA if wake from sleep is not required.

**Ind.** Inductor connection to boost pump.

**kHz XTAL: Xo, kHz XTAL: Xi.** 32.768-kHz crystal oscillator pin.

**MHz XTAL: Xo, MHz XTAL: Xi.** 4 to 25-MHz crystal oscillator pin.

**nTRST.** Optional JTAG Test Reset programming and debug port connection to reset the JTAG connection.

**SIO.** Special I/O provides interfaces to the CPU, digital peripherals and interrupts with a programmable high threshold voltage, analog comparator, high sink current, and high impedance state when the device is unpowered.

**SWDCK.** Serial wire debug clock programming and debug port connection.

**SWDIO.** Serial wire debug Input and output programming and debug port connection.

**Note**
7. GPIOs with opamp outputs are not recommended for use with CapSense.

**TCK.** JTAG test clock programming and debug port connection.

**TDI.** JTAG test data In programming and debug port connection.

**TDO.** JTAG test data out programming and debug port connection.

**TMS.** JTAG test mode select programming and debug port connection.

**TRACECLK.** Cortex-M3 TRACEPORT connection, clocks TRACEDATA pins.

**TRACEDATA[3:0].** Cortex-M3 TRACEPORT connections, output data.

**SWV.** Single wire viewer output.

**USBIO, D+.** Provides D+ connection directly to a USB 2.0 bus. May be used as a digital I/O pin; it is powered from VDDD instead of from a $V_{DDIO}$. Pins are Do Not Use (DNU) on devices without USB.

**USBIO, D-.** Provides D- connection directly to a USB 2.0 bus. May be used as a digital I/O pin; it is powered from $V_{DDD}$ instead of from a $V_{DDIO}$. Pins are Do Not Use (DNU) on devices without USB.

**VBOOST.** Power sense connection to boost pump.

**VBAT.** Battery supply to boost pump.

**VCCA. Output of the analog core regulator or the input to the analog core.** Requires a 1uF capacitor to VSSA. The regulator output is not designed to drive external circuits. **Note that if you use the device with an external core regulator (externally regulated mode), the voltage applied to this pin must not exceed the allowable range of 1.71 V to 1.89 V.** When using the internal core regulator, (internally regulated mode, the default), do not tie any power to this pin. For details see "Power System" section on page 26.

**VCCD. Output of the digital core regulator or the input to the digital core.** The two VCCD pins must be shorted together, with the trace between them as short as possible, and a 1uF capacitor to VSSD. The regulator output is not designed to drive external circuits. **Note that if you use the device with an external core regulator (externally regulated mode), the voltage applied to this pin must not exceed the allowable range of 1.71 V to 1.89 V.** When using the internal core regulator (internally regulated mode, the default), do not tie any power to this pin. For details see "Power System" section on page 26.

**VDDA.** Supply for all analog peripherals and analog core regulator. VDDA must be the highest voltage present on the device. All other supply pins must be less than or equal to VDDA.

**VDDD.** Supply for all digital peripherals and digital core regulator. VDDD must be less than or equal to VDDA.

**VSSA.** Ground for all analog peripherals.

**VSSB.** Ground connection for boost pump.

**VSSD.** Ground for all digital logic and I/O pins.

**VDDIO0, VDDIO1, VDDIO2, VDDIO3.** Supply for I/O pins. Each VDDIO must be tied to a valid operating voltage (1.71 V to 5.5 V), and must be less than or equal to VDDA.

**XRES.** External reset pin. Active low with internal pull-up.

# 4. CPU

## 4.1 ARM Cortex-M3 CPU

The CY8C52LP family of devices has an ARM Cortex-M3 CPU core. The Cortex-M3 is a low power 32-bit three-stage pipelined Harvard architecture CPU that delivers 1.25 DMIPS/MHz. It is intended for deeply embedded applications that require fast interrupt handling features.

**Figure 4-1. ARM Cortex-M3 Block Diagram**
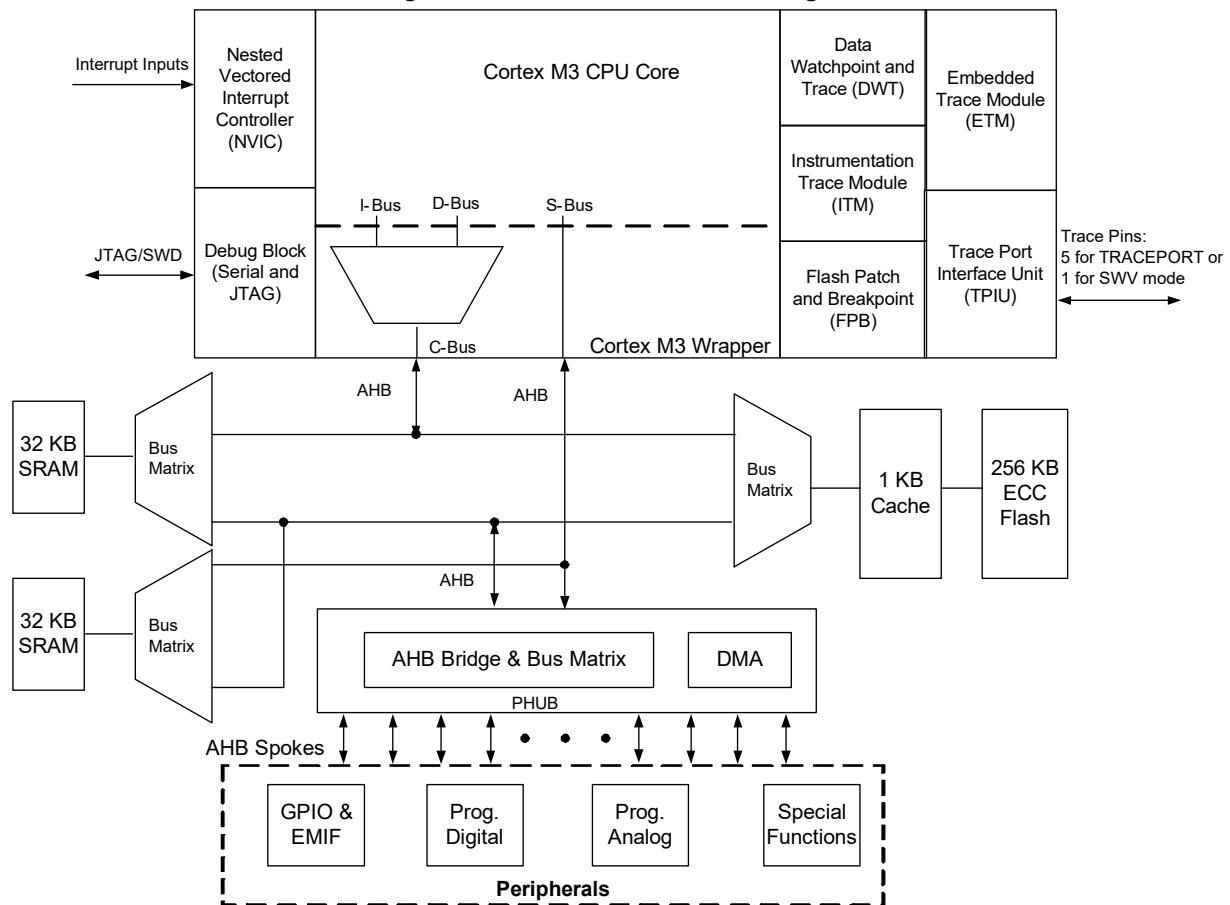


The Cortex-M3 CPU subsystem includes these features:

- ARM Cortex-M3 CPU

- Programmable nested vectored interrupt controller (NVIC), tightly integrated with the CPU core

- Full-featured debug and trace modules, tightly integrated with the CPU core

- Up to 256 KB of flash memory, 2 KB of EEPROM, and 64 KB of SRAM

- Cache controller

- Peripheral HUB (PHUB)

- DMA controller

- External memory interface (EMIF)

### 4.1.1 Cortex-M3 Features

The Cortex-M3 CPU features include:

- 4-GB address space. Predefined address regions for code, data, and peripherals. Multiple buses for efficient and simultaneous accesses of instructions, data, and peripherals.

- The Thumb®-2 instruction set, which offers ARM-level performance at Thumb-level code density. This includes 16-bit and 32-bit instructions. Advanced instructions include:
  - Bit-field control
  - Hardware multiply and divide
  - Saturation
  - If-Then
  - Wait for events and interrupts
  - Exclusive access and barrier
  - Special register access

  The Cortex-M3 does not support ARM instructions.

- Bit-band support for the SRAM region. Atomic bit-level write and read operations for SRAM addresses.

- Unaligned data storage and access. Contiguous storage of data of different byte lengths.

- Operation at two privilege levels (privileged and user) and in two modes (thread and handler). Some instructions can only be executed at the privileged level. There are also two stack pointers: Main (MSP) and Process (PSP). These features support a multitasking operating system running one or more user-level processes.

- Extensive interrupt and system exception support.

### 4.1.2 Cortex-M3 Operating Modes

The Cortex-M3 operates at either the privileged level or the user level, and in either the thread mode or the handler mode. Because the handler mode is only enabled at the privileged level, there are actually only three states, as shown in Table 4-1.

**Table 4-1. Operational Level**

| Condition | Privileged | User |
|---|---|---|
| Running an exception | Handler mode | Not used |
| Running main program | Thread mode | Thread mode |

At the user level, access to certain instructions, special registers, configuration registers, and debugging components is blocked. Attempts to access them cause a fault exception. At the privileged level, access to all instructions and registers is allowed. The processor runs in the handler mode (always at the privileged level) when handling an exception, and in the thread mode when not.

### 4.1.3 CPU Registers

The Cortex-M3 CPU registers are listed in Table 4-2. Registers R0-R15 are all 32 bits wide.

**Table 4-2. Cortex M3 CPU Registers**

| Register | Description |
|---|---|
| R0-R12 | General purpose registers R0-R12 have no special architecturally defined uses. Most instructions that specify a general purpose register specify R0-R12.<br>■ Low Registers: Registers R0-R7 are accessible by all instructions that specify a general purpose register.<br>■ High Registers: Registers R8-R12 are accessible by all 32-bit instructions that specify a general purpose register; they are not accessible by all 16-bit instructions. |
| R13 | R13 is the stack pointer register. It is a banked register that switches between two 32-bit stack pointers: the main stack pointer (MSP) and the process stack pointer (PSP). The PSP is used only when the CPU operates at the user level in thread mode. The MSP is used in all other privilege levels and modes. Bits[0:1] of the SP are ignored and considered to be 0, so the SP is always aligned to a word (4 byte) boundary. |
| R14 | R14 is the link register (LR). The LR stores the return address when a subroutine is called. |

**Table 4-2. Cortex M3 CPU Registers** (continued)

| Register | Description |
|---|---|
| R15 | R15 is the program counter (PC). Bit 0 of the PC is ignored and considered to be 0, so instructions are always aligned to a half word (2 byte) boundary. |
| xPSR | The program status registers are divided into three status registers, which are accessed either together or separately:<br>■ Application program status register (APSR) holds program execution status bits such as zero, carry, negative, in bits[27:31].<br>■ Interrupt program status register (IPSR) holds the current exception number in bits[0:8].<br>■ Execution program status register (EPSR) holds control bits for interrupt continuable and IF-THEN instructions in bits[10:15] and [25:26]. Bit 24 is always set to 1 to indicate Thumb mode. Trying to clear it causes a fault exception. |
| PRIMASK | A 1-bit interrupt mask register. When set, it allows only the nonmaskable interrupt (NMI) and hard fault exception. All other exceptions and interrupts are masked. |
| FAULTMASK | A 1-bit interrupt mask register. When set, it allows only the NMI. All other exceptions and interrupts are masked. |
| BASEPRI | A register of up to nine bits that define the masking priority level. When set, it disables all interrupts of the same or higher priority value. If set to 0 then the masking function is disabled. |
| CONTROL | A 2-bit register for controlling the operating mode.<br>Bit 0: 0 = privileged level in thread mode, 1 = user level in thread mode.<br>Bit 1: 0 = default stack (MSP) is used, 1 = alternate stack is used. If in thread mode or user level then the alternate stack is the PSP. There is no alternate stack for handler mode; the bit must be 0 while in handler mode. |

## 4.2 Cache Controller

The CY8C52LP family has a 1 KB, 4-way set-associative cache between the CPU and the flash memory. This improves instruction execution rate and reduces system power consumption by requiring less frequent flash access.

## 4.3 DMA and PHUB

The PHUB and the DMA controller are responsible for data transfer between the CPU and peripherals, and also data transfers between peripherals. The PHUB and DMA also control device configuration during boot. The PHUB consists of:

- A central hub that includes the DMA controller, arbiter, and router

- Multiple spokes that radiate outward from the hub to most peripherals

There are two PHUB masters: the CPU and the DMA controller. Both masters may initiate transactions on the bus. The DMA channels can handle peripheral communication without CPU intervention. The arbiter in the central hub determines which DMA channel is the highest priority if there are multiple requests.

### 4.3.1 PHUB Features

■ CPU and DMA controller are both bus masters to the PHUB

■ Eight multi-layer AHB bus parallel access paths (spokes) for peripheral access

■ Simultaneous CPU and DMA access to peripherals located on different spokes

■ Simultaneous DMA source and destination burst transactions on different spokes

■ Supports 8-, 16-, 24-, and 32-bit addressing and data

**Table 4-3. PHUB Spokes and Peripherals**

| PHUB Spokes | Peripherals |
|---|---|
| 0 | SRAM |
| 1 | IOs, PICU, EMIF |
| 2 | PHUB local configuration, Power manager, Clocks, IC, SWV, EEPROM, Flash programming interface |
| 3 | Analog interface and trim, Decimator |
| 4 | USB, I²C, Timers, Counters, and PWMs |
| 5 | Reserved |
| 6 | UDBs group 1 |
| 7 | UDBs group 2 |

### 4.3.2 DMA Features

■ 24 DMA channels

■ Each channel has one or more transaction descriptors (TDs) to configure channel behavior. Up to 128 total TDs can be defined

■ TDs can be dynamically updated

■ Eight levels of priority per channel

■ Any digitally routable signal, the CPU, or another DMA channel, can trigger a transaction

■ Each channel can generate up to two interrupts per transfer

■ Transactions can be stalled or canceled

■ Supports transaction size of infinite or 1 to 64 k bytes

■ Large transactions may be broken into smaller bursts of 1 to 127 bytes

■ TDs may be nested and/or chained for complex transactions

### 4.3.3 Priority Levels

The CPU always has higher priority than the DMA controller when their accesses require the same bus resources. Due to the system architecture, the CPU can never starve the DMA. DMA channels of higher priority (lower priority number) may interrupt current DMA transfers. In the case of an interrupt, the current transfer is allowed to complete its current transaction. To ensure latency limits when multiple DMA accesses are requested simultaneously, a fairness algorithm guarantees an interleaved minimum percentage of bus bandwidth for priority levels 2 through 7. Priority levels 0 and 1 do not take part in the fairness algorithm and may use 100% of the bus bandwidth. If a tie occurs on two DMA requests of the same priority level, a simple round robin method is used to evenly share the allocated bandwidth. The round robin allocation can be disabled for each DMA channel, allowing it to always be at the head of the line. Priority levels 2 to 7 are guaranteed the minimum bus bandwidth shown in Table 4-4 after the CPU and DMA priority levels 0 and 1 have satisfied their requirements.

**Table 4-4. Priority Levels**

| Priority Level | % Bus Bandwidth |
|---|---|
| 0 | 100.0 |
| 1 | 100.0 |
| 2 | 50.0 |
| 3 | 25.0 |
| 4 | 12.5 |
| 5 | 6.2 |
| 6 | 3.1 |
| 7 | 1.5 |

When the fairness algorithm is disabled, DMA access is granted based solely on the priority level; no bus bandwidth guarantees are made.

### 4.3.4 Transaction Modes Supported

The flexible configuration of each DMA channel and the ability to chain multiple channels allow the creation of both simple and complex use cases. General use cases include, but are not limited to:

#### 4.3.4.1 Simple DMA

In a simple DMA case, a single TD transfers data between a source and sink (peripherals or memory location). The basic timing diagrams of DMA read and write cycles are shown in Figure 4-2. For more description on other transfer modes, refer to the Technical Reference Manual.

**Figure 4-2. DMA Timing Diagram.**



Basic DMA Read Transfer without wait states



Basic DMA Write Transfer without wait states
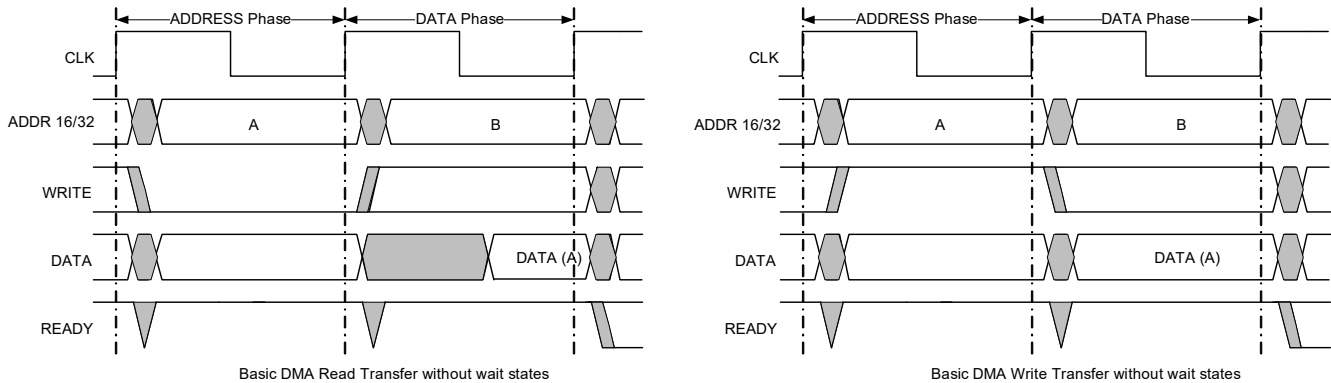
### 4.3.4.2 Auto Repeat DMA

Auto repeat DMA is typically used when a static pattern is repetitively read from system memory and written to a peripheral. This is done with a single TD that chains to itself.

### 4.3.4.3 Ping Pong DMA

A ping pong DMA case uses double buffering to allow one buffer to be filled by one client while another client is consuming the data previously received in the other buffer. In its simplest form, this is done by chaining two TDs together so that each TD calls the opposite TD when complete.

### 4.3.4.4 Circular DMA

Circular DMA is similar to ping pong DMA except it contains more than two buffers. In this case there are multiple TDs; after the last TD is complete it chains back to the first TD.

### 4.3.4.5 Indexed DMA

In an indexed DMA case, an external master requires access to locations on the system bus as if those locations were shared memory. As an example, a peripheral may be configured as an SPI or I²C slave where an address is received by the external master. That address becomes an index or offset into the internal system bus memory space. This is accomplished with an initial "address fetch" TD that reads the target address location from the peripheral and writes that value into a subsequent TD in the chain. This modifies the TD chain on the fly. When the "address fetch" TD completes it moves on to the next TD, which has the new address information embedded in it. This TD then carries out the data transfer with the address location required by the external master.

### 4.3.4.6 Scatter Gather DMA

In the case of scatter gather DMA, there are multiple noncontiguous sources or destinations that are required to effectively carry out an overall DMA transaction. For example, a packet may need to be transmitted off of the device and the packet elements, including the header, payload, and trailer, exist in various noncontiguous locations in memory. Scatter gather DMA allows the segments to be concatenated together by using multiple TDs in a chain. The chain gathers the data from the multiple locations. A similar concept applies for the reception of data onto the device. Certain parts of the received data may need to be scattered to various locations in memory for software processing convenience. Each TD in the chain specifies the location for each discrete element in the chain.

### 4.3.4.7 Packet Queuing DMA

Packet queuing DMA is similar to scatter gather DMA but specifically refers to packet protocols. With these protocols, there may be separate configuration, data, and status phases associated with sending or receiving a packet.

For instance, to transmit a packet, a memory mapped configuration register can be written inside a peripheral, specifying the overall length of the ensuing data phase. The CPU can set up this configuration information anywhere in system memory and copy it with a simple TD to the peripheral. After the configuration phase, a data phase TD (or a series of data phase TDs) can begin (potentially using scatter gather). When the data phase TD(s) finish, a status phase TD can be invoked that reads some memory mapped status information from the peripheral and copies it to a location in system memory specified by the CPU for later inspection. Multiple sets of configuration, data, and status phase "subchains" can be strung together to create larger chains that transmit multiple packets in this way. A similar concept exists in the opposite direction to receive the packets.

### 4.3.4.8 Nested DMA

One TD may modify another TD, as the TD configuration space is memory mapped similar to any other peripheral. For example, a first TD loads a second TD's configuration and then calls the second TD. The second TD moves data as required by the application. When complete, the second TD calls the first TD, which again updates the second TD's configuration. This process repeats as often as necessary.

## 4.4 Interrupt Controller

The Cortex-M3 NVIC supports 16 system exceptions and 32 interrupts from peripherals, as shown in Table 4-5.

**Table 4-5. Cortex-M3 Exceptions and Interrupts**

| Exception Number | Exception Type | Priority | Exception Table Address Offset | Function |
|---|---|---|---|---|
| | | | 0x00 | Starting value of R13 / MSP |
| 1 | Reset | –3 (highest) | 0x04 | Reset |
| 2 | NMI | –2 | 0x08 | Non maskable interrupt |
| 3 | Hard fault | –1 | 0x0C | All classes of fault, when the corresponding fault handler cannot be activated because it is currently disabled or masked |
| 4 | MemManage | Programmable | 0x10 | Memory management fault, for example, instruction fetch from a nonexecutable region |
| 5 | Bus fault | Programmable | 0x14 | Error response received from the bus system; caused by an instruction prefetch abort or data access error |
| 6 | Usage fault | Programmable | 0x18 | Typically caused by invalid instructions or trying to switch to ARM mode |
| 7–10 | – | – | 0x1C–0x28 | Reserved |
| 11 | SVC | Programmable | 0x2C | System service call via SVC instruction |
| 12 | Debug monitor | Programmable | 0x30 | Debug monitor |
| 13 | – | – | 0x34 | Reserved |
| 14 | PendSV | Programmable | 0x38 | Deferred request for system service |
| 15 | SYSTICK | Programmable | 0x3C | System tick timer |
| 16–47 | IRQ | Programmable | 0x40–0x3FC | Peripheral interrupt request #0 – #31 |

Bit 0 of each exception vector indicates whether the exception is executed using ARM or Thumb instructions. Because the Cortex-M3 only supports Thumb instructions, this bit must always be 1. The Cortex-M3 non maskable interrupt (NMI) input can be routed to any pin, via the DSI, or disconnected from all pins. See "DSI Routing Interface Description" section on page 44.

The Nested Vectored Interrupt Controller (NVIC) handles interrupts from the peripherals, and passes the interrupt vectors to the CPU. It is closely integrated with the CPU for low latency interrupt handling. Features include:

■ 32 interrupts. Multiple sources for each interrupt.

■ Eight priority levels, with dynamic priority control.

■ Priority grouping. This allows selection of preempting and non preempting interrupt levels.

■ Support for tail-chaining, and late arrival, of interrupts. This enables back-to-back interrupt processing without the overhead of state saving and restoration between interrupts.

■ Processor state automatically saved on interrupt entry, and restored on interrupt exit, with no instruction overhead.

If the same priority level is assigned to two or more interrupts, the interrupt with the lower vector number is executed first. Each interrupt vector may choose from three interrupt sources: Fixed Function, DMA, and UDB. The fixed function interrupts are direct connections to the most common interrupt sources and provide the lowest resource cost connection. The DMA interrupt sources provide direct connections to the two DMA interrupt sources provided per DMA channel. The third interrupt source for vectors is from the UDB digital routing array. This allows any digital signal available to the UDB array to be used as an interrupt source. All interrupt sources may be routed to any interrupt vector using the UDB interrupt source connections.

**Table 4-6.  Interrupt Vector Table**

| Interrupt # | Cortex-M3 Exception # | Fixed Function | DMA | UDB |
|---|---|---|---|---|
| 0 | 16 | Low voltage detect (LVD) | phub_termout0[0] | udb_intr[0] |
| 1 | 17 | Cache/ECC | phub_termout0[1] | udb_intr[1] |
| 2 | 18 | Reserved | phub_termout0[2] | udb_intr[2] |
| 3 | 19 | Sleep (Pwr Mgr) | phub_termout0[3] | udb_intr[3] |
| 4 | 20 | PICU[0] | phub_termout0[4] | udb_intr[4] |
| 5 | 21 | PICU[1] | phub_termout0[5] | udb_intr[5] |
| 6 | 22 | PICU[2] | phub_termout0[6] | udb_intr[6] |
| 7 | 23 | PICU[3] | phub_termout0[7] | udb_intr[7] |
| 8 | 24 | PICU[4] | phub_termout0[8] | udb_intr[8] |
| 9 | 25 | PICU[5] | phub_termout0[9] | udb_intr[9] |
| 10 | 26 | PICU[6] | phub_termout0[10] | udb_intr[10] |
| 11 | 27 | PICU[12] | phub_termout0[11] | udb_intr[11] |
| 12 | 28 | PICU[15] | phub_termout0[12] | udb_intr[12] |
| 13 | 29 | Comparators Combined | phub_termout0[13] | udb_intr[13] |
| 14 | 30 | Reserved | phub_termout0[14] | udb_intr[14] |
| 15 | 31 | I$^2$C | phub_termout0[15] | udb_intr[15] |
| 16 | 32 | Reserved | phub_termout1[0] | udb_intr[16] |
| 17 | 33 | Timer/Counter0 | phub_termout1[1] | udb_intr[17] |
| 18 | 34 | Timer/Counter1 | phub_termout1[2] | udb_intr[18] |
| 19 | 35 | Timer/Counter2 | phub_termout1[3] | udb_intr[19] |
| 20 | 36 | Timer/Counter3 | phub_termout1[4] | udb_intr[20] |
| 21 | 37 | USB SOF Int | phub_termout1[5] | udb_intr[21] |
| 22 | 38 | USB Arb Int | phub_termout1[6] | udb_intr[22] |
| 23 | 39 | USB Bus Int | phub_termout1[7] | udb_intr[23] |
| 24 | 40 | USB Endpoint[0] | phub_termout1[8] | udb_intr[24] |
| 25 | 41 | USB Endpoint Data | phub_termout1[9] | udb_intr[25] |
| 26 | 42 | Reserved | phub_termout1[10] | udb_intr[26] |
| 27 | 43 | LCD | phub_termout1[11] | udb_intr[27] |
| 28 | 44 | Reserved | phub_termout1[12] | udb_intr[28] |
| 29 | 45 | Decimator Int | phub_termout1[13] | udb_intr[29] |
| 30 | 46 | phub_err_int | phub_termout1[14] | udb_intr[30] |
| 31 | 47 | eeprom_fault_int | phub_termout1[15] | udb_intr[31] |

# 5. Memory

## 5.1 Static RAM

CY8C52LP Static RAM (SRAM) is used for temporary data storage. Code can be executed at full speed from the portion of SRAM that is located in the code space. This process is slower from SRAM above 0x20000000. The device provides up to 64 KB of SRAM. The CPU or the DMA controller can access all of SRAM. The SRAM can be accessed simultaneously by the Cortex-M3 CPU and the DMA controller if accessing different 32-KB blocks.

## 5.2 Flash Program Memory

Flash memory in PSoC devices provides nonvolatile storage for user firmware, user configuration data, bulk data storage, and optional ECC data. The main flash memory area contains up to 256 KB of user program space.

Up to an additional 32 KB of flash space is available for Error Correcting Codes (ECC). If ECC is not used this space can store device configuration data and bulk user data. User code may not be run out of the ECC flash memory section. ECC can correct one bit error and detect two bit errors per 8 bytes of firmware memory; an interrupt can be generated when an error is detected. The flash output is 9 bytes wide with 8 bytes of data and 1 byte of ECC data.

The CPU or DMA controller read both user code and bulk data located in flash through the cache controller. This provides higher CPU performance. If ECC is enabled, the cache controller also performs error checking and correction.

Flash programming is performed through a special interface and preempts code execution out of flash. Code execution may be done out of SRAM during flash programming.

The flash programming interface performs flash erasing, programming and setting code protection levels. flash in-system serial programming (ISSP), typically used for production programming, is possible through both the SWD and JTAG interfaces. In-system programming, typically used for bootloaders, is also possible using serial interfaces such as I²C, USB, UART, and SPI, or any communications protocol.

## 5.3 Flash Security

All PSoC devices include a flexible flash protection model that prevents access and visibility to on-chip flash memory. This prevents duplication or reverse engineering of proprietary code. Flash memory is organized in blocks, where each block contains 256 bytes of program or data and 32 bytes of ECC or configuration data.

The device offers the ability to assign one of four protection levels to each row of flash. Table 5-1 lists the protection modes available. Flash protection levels can only be changed by performing a complete flash erase. The Full Protection and Field Upgrade settings disable external access (through a debugging tool such as PSoC Creator, for example). If your application requires code update through a boot loader, then use the Field Upgrade setting. Use the Unprotected setting only when no security is needed in your application. The PSoC device also offers an advanced security feature called Device Security which permanently disables all test, programming, and debug ports, protecting your application from external access (see the

"Device Security" section on page 58). For more information on how to take full advantage of the security features in PSoC, see the PSoC 5 TRM.

**Table 5-1. Flash Protection**

| Protection Setting | Allowed | Not Allowed |
|---|---|---|
| Unprotected | External read and write + internal read and write | – |
| Factory Upgrade | External write + internal read and write | External read |
| Field Upgrade | Internal read and write | External read and write |
| Full Protection | Internal read | External read and write + internal write |

**Disclaimer**

Note the following details of the flash code protection features on Cypress devices.

Cypress products meet the specifications contained in their particular Cypress datasheets. Cypress believes that its family of 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.

## 5.4 EEPROM

PSoC EEPROM memory is a byte addressable nonvolatile memory. The CY8C52LP has 2 KB of EEPROM memory to store user data. Reads from EEPROM are random access at the byte level. Reads are done directly; writes are done by sending write commands to an EEPROM programming interface. CPU code execution can continue from flash during EEPROM writes. EEPROM is erasable and writeable at the row level. The EEPROM is divided into 128 rows of 16 bytes each. The factory default values of all EEPROM bytes are 0.

Because the EEPROM is mapped to the Cortex-M3 Peripheral region, the CPU cannot execute out of EEPROM. There is no ECC hardware associated with EEPROM. If ECC is required it must be handled in firmware.

It can take as much as 20 milliseconds to write to EEPROM or flash. During this time the device should not be reset, or unexpected changes may be made to portions of EEPROM or flash. Reset sources (see Section 6.3.1) include XRES pin, software reset, and watchdog; care should be taken to make sure that these are not inadvertently activated. In addition, the low voltage detect circuits should be configured to generate an interrupt instead of a reset.

## 5.5 Nonvolatile Latches (NVLs)

PSoC has a 4-byte array of nonvolatile latches (NVLs) that are used to configure the device at reset. The NVL register map is shown in Table 5-3.

**Table 5-2.  Device Configuration NVL Register Map**

| Register Address | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0x00 | PRT3RDM[1:0] | | PRT2RDM[1:0] | | PRT1RDM[1:0] | | PRT0RDM[1:0] | |
| 0x01 | PRT12RDM[1:0] | | PRT6RDM[1:0] | | PRT5RDM[1:0] | | PRT4RDM[1:0] | |
| 0x02 | XRESMEN | DBGEN | | | | | PRT15RDM[1:0] | |
| 0x03 | DIG_PHS_DLY[3:0] | | | | ECCEN | DPS[1:0] | | CFGSPEED |

The details for individual fields and their factory default settings are shown in Table 5-3:.

**Table 5-3.  Fields and Factory Default Settings**

| Field | Description | Settings |
|---|---|---|
| PRTxRDM[1:0] | Controls reset drive mode of the corresponding IO port. See "Reset Configuration" on page 38. All pins of the port are set to the same mode. | 00b (default) - high impedance analog<br>01b - high impedance digital<br>10b - resistive pull up<br>11b - resistive pull down |
| XRESMEN | Controls whether pin P1[2] is used as a GPIO or as an external reset. P1[2] is generally used as a GPIO, and not as an external reset. | 0 (default) - GPIO<br>1 - external reset |
| DBGEN | Debug Enable allows access to the debug system, for third-party programmers. | 0 - access disabled<br>1 (default) - access enabled |
| CFGSPEED | Controls the speed of the IMO-based clock during the device boot process, for faster boot or low-power operation. | 0 (default) - 12-MHz IMO<br>1 - 48-MHz IMO |
| DPS[1:0] | Controls the usage of various P1 pins as a debug port. See "Programming, Debug Interfaces, Resources" on page 55. | 00b - 5-wire JTAG<br>01b (default) - 4-wire JTAG<br>10b - SWD<br>11b - debug ports disabled |
| ECCEN | Controls whether ECC flash is used for ECC or for general configuration and data storage. See "Flash Program Memory" on page 19. | 0 - ECC disabled<br>1 (default) - ECC enabled |
| DIG_PHS_DLY[3:0] | Selects the digital clock phase delay. | See the TRM for details. |

Although PSoC Creator provides support for modifying the device configuration NVLs, the number of NVL erase/write cycles is limited – see "Nonvolatile Latches (NVL)" on page 94.

## 5.6 External Memory Interface

CY8C52LP provides an External Memory Interface (EMIF) for connecting to external memory devices. The connection allows read and write accesses to external memories. The EMIF operates in conjunction with UDBs, I/O ports, and other hardware to generate external memory address and control signals. At 33 MHz, each memory access cycle takes four bus clock cycles.

Figure 5-1 is the EMIF block diagram. The EMIF supports synchronous and asynchronous memories. The CY8C52LP only supports one type of external memory device at a time.

External memory is located in the Cortex-M3 external RAM space; it can use up to 24 address bits. See Table 5-4 on page 22 and Memory Map on page 22. The memory can be 8 or 16 bits wide.

Cortex-M3 instructions can be fetched from external memory if it is 16-bit. Other limitations apply; for details, see application note AN89610, PSoC® 4 and PSoC 5LP Arm Cortex Code Optimization. There is no provision for code security in external memory. If code must be kept secure, then it should be placed in internal flash. See Flash Security on page 19 and Device Security on page 58.

**Figure 5-1. EMIF Block Diagram**
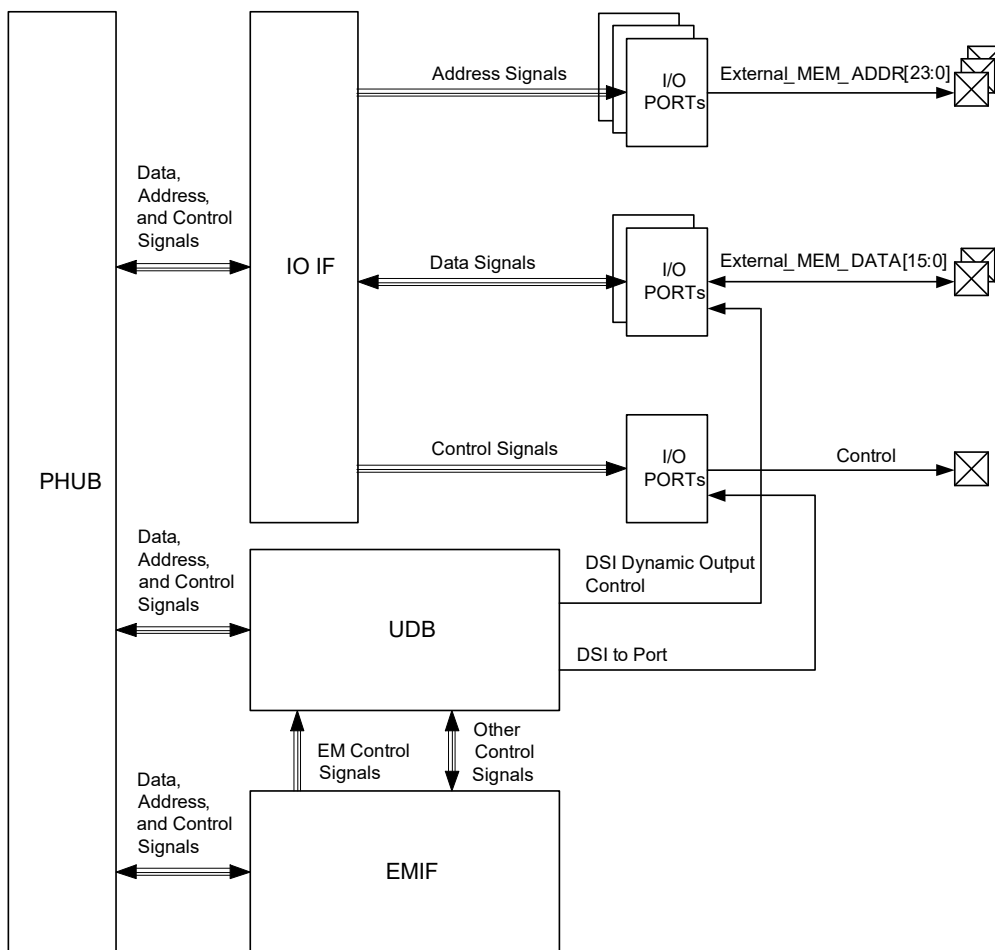
## 5.7 Memory Map

The Cortex-M3 has a fixed address map, which allows peripherals to be accessed by simple memory access instructions.

### 5.7.1 Address Map

The 4-GB address space is divided into the ranges shown in Table 5-4:

**Table 5-4. Address Map**

| Address Range | Size | Use |
|---|---|---|
| 0x00000000– 0x1FFFFFFF | 0.5 GB | Program code. This includes the exception vector table at power up, which starts at address 0. |
| 0x20000000– 0x3FFFFFFF | 0.5 GB | Static RAM. This includes a 1 MByte bit-band region starting at 0x20000000 and a 32 Mbyte bit-band alias region starting at 0x22000000. |
| 0x40000000– 0x5FFFFFFF | 0.5 GB | Peripherals. |
| 0x60000000– 0x9FFFFFFF | 1 GB | External RAM. |
| 0xA0000000– 0xDFFFFFFF | 1 GB | External peripherals. |
| 0xE0000000– 0xFFFFFFFF | 0.5 GB | Internal peripherals, including the NVIC and debug and trace modules. |

**Table 5-5. Peripheral Data Address Map**

| Address Range | Purpose |
|---|---|
| 0x00000000–0x0003FFFF | 256 K Flash |
| 0x1FFF8000–0x1FFFFFFF | 32 K SRAM in Code region |
| 0x20000000–0x20007FFF | 32 K SRAM in SRAM region |
| 0x40004000–0x400042FF | Clocking, PLLs, and oscillators |
| 0x40004300–0x400043FF | Power management |
| 0x40004500–0x400045FF | Ports interrupt control |
| 0x40004700–0x400047FF | Flash programming interface |
| 0x40004800–0x400048FF | Cache controller |
| 0x40004900–0x400049FF | I²C controller |

**Table 5-5. Peripheral Data Address Map** (continued)

| Address Range | Purpose |
|---|---|
| 0x40004E00–0x40004EFF | Decimator |
| 0x40004F00–0x40004FFF | Fixed timer/counter/PWMs |
| 0x40005000–0x400051FF | I/O ports control |
| 0x40005400–0x400054FF | External Memory Interface (EMIF) control registers |
| 0x40005800–0x40005FFF | Analog Subsystem Interface |
| 0x40006000–0x400060FF | USB Controller |
| 0x40006400–0x40006FFF | UDB Working Registers |
| 0x40007000–0x40007FFF | PHUB Configuration |
| 0x40008000–0x400087FF | EEPROM |
| 0x4000A000–0x4000A400 | Reserved |
| 0x40010000–0x4001FFFF | Digital Interconnect Configuration |
| 0x48000000–0x48007FFF | Flash ECC Bytes |
| 0x60000000–0x60FFFFFF | External Memory Interface (EMIF) |
| 0xE0000000–0xE00FFFFF | Cortex-M3 PPB Registers, including NVIC, debug, and trace |

The bit-band feature allows individual bits in SRAM to be read or written as atomic operations. This is done by reading or writing bit 0 of corresponding words in the bit-band alias region. For example, to set bit 3 in the word at address 0x20000000, write a 1 to address 0x2200000C. To test the value of that bit, read address 0x2200000C and the result is either 0 or 1 depending on the value of the bit.

Most memory accesses done by the Cortex-M3 are aligned, that is, done on word (4-byte) boundary addresses. Unaligned accesses of words and 16-bit half-words on nonword boundary addresses can also be done, although they are less efficient.

### 5.7.2 Address Map and Cortex-M3 Buses

The ICode and DCode buses are used only for accesses within the Code address range, 0–0x1FFFFFFF.

The system bus is used for data accesses and debug accesses within the ranges 0x20000000–0xDFFFFFFF and 0xE0100000–0xFFFFFFFF. Instruction fetches can also be done within the range 0x20000000–0x3FFFFFFF, although these can be slower than instruction fetches via the ICode bus.

The private peripheral bus (PPB) is used within the Cortex-M3 to access system control registers and debug and trace module registers.

# 6. System Integration

## 6.1 Clocking System

The clocking system generates, divides, and distributes clocks throughout the PSoC system. For the majority of systems, no external crystal is required. The IMO and PLL together can generate up to a 80-MHz clock, accurate to ±2% over voltage and temperature. Additional internal and external clock sources allow each design to optimize accuracy, power, and cost. All of the system clock sources can be used to generate other clock frequencies in the 16-bit clock dividers and UDBs for anything you want, for example a UART baud rate generator.

Clock generation and distribution is automatically configured through the PSoC Creator IDE graphical interface. This is based on the complete system's requirements. It greatly speeds the design process. PSoC Creator allows designers to build clocking systems with minimal input. You can specify desired clock frequencies and accuracies, and the software locates or builds a clock that meets the required specifications. This is possible because of the programmability inherent in PSoC.

Key features of the clocking system include:

- Seven general purpose clock sources
  - 3- to 74-MHz IMO, ±2% at 3 MHz
  - 4- to 25-MHz external crystal oscillator (MHzECO)
  - Clock doubler provides a doubled clock frequency output for the USB block, see USB Clock Domain on page 26.
  - DSI signal from an external I/O pin or other logic
  - 24- to 80-MHz fractional phase-locked loop (PLL) sourced from IMO, MHzECO, or DSI
  - 1-kHz, 33-kHz, 100-kHz ILO for watchdog timer (WDT) and Sleep Timer
  - 32.768-kHz external crystal oscillator (ECO) for RTC
- IMO has a USB mode that auto-locks to the USB bus clock requiring no external crystal for USB. (USB equipped parts only)
- Independently sourced clock in all clock dividers
- Eight 16-bit clock dividers for the digital system
- Four 16-bit clock dividers for the analog system
- Dedicated 16-bit divider for the CPU bus and CPU clock
- Automatic clock configuration in PSoC Creator

**Table 6-1. Oscillator Summary**

| Source | Fmin | Tolerance at Fmin | Fmax | Tolerance at Fmax | Startup Time |
|---|---|---|---|---|---|
| IMO | 3 MHz | ±2% over voltage and temperature | 74 MHz | ±7% | 13 µs max |
| MHzECO | 4 MHz | Crystal dependent | 25 MHz | Crystal dependent | 5 ms typ, max is crystal dependent |
| DSI | 0 MHz | Input dependent | 33 MHz | Input dependent | Input dependent |
| PLL | 24 MHz | Input dependent | 80 MHz | Input dependent | 250 µs max |
| Doubler | 48 MHz | Input dependent | 48 MHz | Input dependent | 1 µs max |
| ILO | 1 kHz | −50%, +100% | 100 kHz | −55%, +100% | 15 ms max in lowest power mode |
| kHzECO | 32 kHz | Crystal dependent | 32 kHz | Crystal dependent | 500 ms typ, max is crystal dependent |

## Figure 6-1. Clocking Subsystem
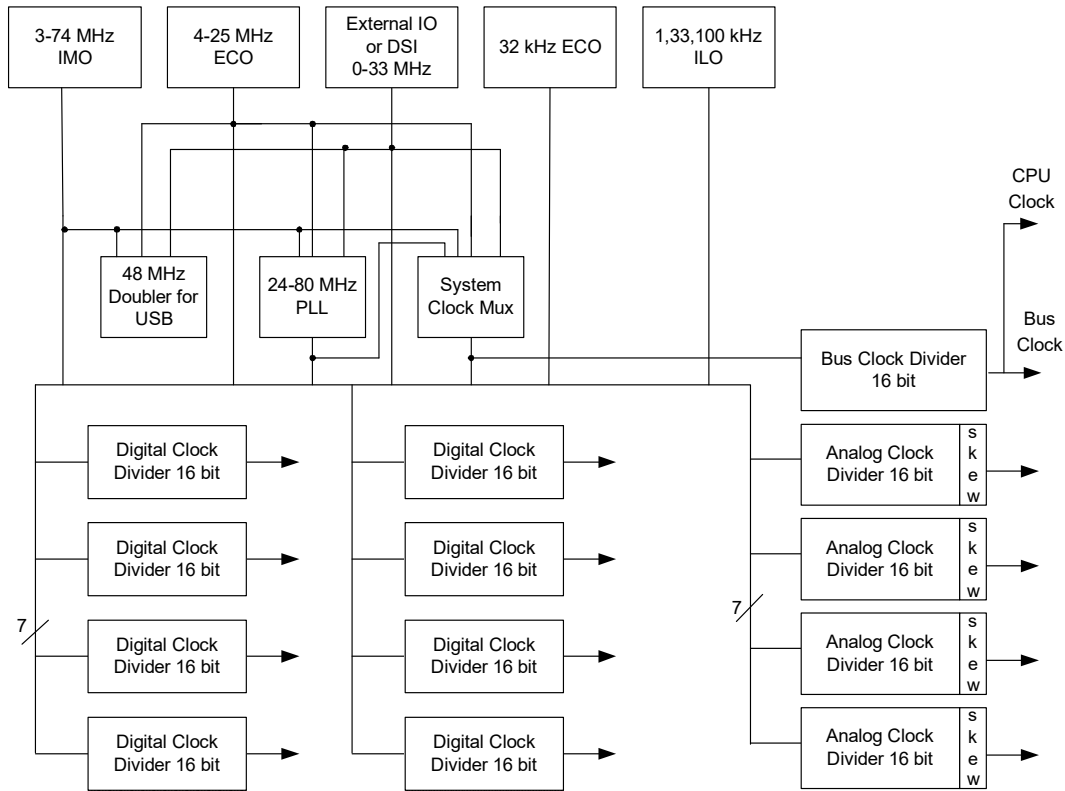


### 6.1.1 Internal Oscillators

Figure 6-1 shows that there are two internal oscillators. They can be routed directly or divided. The direct routes may not have a 50% duty cycle. Divided clocks have a 50% duty cycle.

#### 6.1.1.1 Internal Main Oscillator

In most designs the IMO is the only clock source required, due to its ±2% accuracy. The IMO operates with no external components and outputs a stable clock. A factory trim for each frequency range is stored in the device. With the factory trim, tolerance varies from ±2% at 3 MHz, up to ±7% at 74 MHz. The IMO, in conjunction with the PLL, allows generation of CPU and system clocks up to the device's maximum frequency (see USB Clock Domain). The IMO provides clock outputs at 3, 6, 12, 24, 48 and 74 MHz.

#### 6.1.1.2 Clock Doubler

The clock doubler outputs a clock at twice the frequency of the input clock. The doubler works at input frequency of 24 MHz, providing 48 MHz for the USB. It can be configured to use a clock from the IMO, MHzECO, or the DSI (external pin).

#### 6.1.1.3 Phase-Locked Loop

The PLL allows low frequency, high accuracy clocks to be multiplied to higher frequencies. This is a tradeoff between higher clock frequency and accuracy and, higher power consumption and increased startup time. The PLL block provides a mechanism for generating clock frequencies based upon a variety of input sources. The PLL outputs clock frequencies in the range of 24 to 80 MHz. Its input and feedback dividers supply

4032 discrete ratios to create almost any desired system clock frequency. The accuracy of the PLL output depends on the accuracy of the PLL input source. The most common PLL use is to multiply the IMO clock at 3 MHz, where it is most accurate, to generate the CPU and system clocks up to the device's maximum frequency.

The PLL achieves phase lock within 250 µs (verified by bit setting). It can be configured to use a clock from the IMO, MHzECO, or DSI (external pin). The PLL clock source can be used until lock is complete and signaled with a lock bit. The lock signal can be routed through the DSI to generate an interrupt. Disable the PLL before entering low power modes.

#### 6.1.1.4 Internal Low Speed Oscillator

The ILO provides clock frequencies for low power consumption, including the watchdog timer, and sleep timer. The ILO generates up to three different clocks: 1 kHz, 33 kHz, and 100 kHz.

The 1-kHz clock (CLK1K) is typically used for a background 'heartbeat' timer. This clock inherently lends itself to low power supervisory operations such as the watchdog timer and long sleep intervals using the central timewheel (CTW).

The central timewheel is a 1-kHz, free-running, 13-bit counter clocked by the ILO. The central timewheel is always enabled except in hibernate mode and when the CPU is stopped during debug on chip mode. It can be used to generate periodic interrupts for timing purposes or to wake the system from a low power mode. Firmware can reset the central timewheel.

The central timewheel can be programmed to wake the system periodically and optionally issue an interrupt. This enables flexible, periodic wakeups from low power modes or coarse timing applications. Systems that require accurate timing should use the RTC capability instead of the central timewheel.

The 100-kHz clock (CLK100K) can be used as a low power system clock to run the CPU. It can also generate time intervals using the fast timewheel.

The fast timewheel is a 5-bit counter, clocked by the 100-kHz clock. It features programmable settings and automatically resets when the terminal count is reached. An optional interrupt can be generated each time the terminal count is reached. This enables flexible, periodic interrupts of the CPU at a higher rate than is allowed using the central timewheel.

The 33-kHz clock (CLK33K) comes from a divide-by-3 operation on CLK100K. This output can be used as a reduced accuracy version of the 32.768-kHz ECO clock with no need for a crystal.
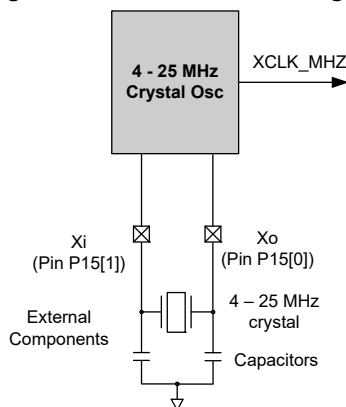
### 6.1.2   External Oscillators

Figure 6-1 shows that there are two external oscillators. They can be routed directly or divided. The direct routes may not have a 50% duty cycle. Divided clocks have a 50% duty cycle.

#### 6.1.2.1   MHz External Crystal Oscillator

The MHzECO provides high frequency, high precision clocking using an external crystal (see Figure 6-2). It supports a wide variety of crystal types, in the range of 4 to 25 MHz. When used in conjunction with the PLL, it can generate CPU and system clocks up to the device's maximum frequency (see Internal Low Speed Oscillator). The GPIO pins connecting to the external crystal and capacitors are fixed. MHzECO accuracy depends on the crystal chosen.

**Figure 6-2. MHzECO Block Diagram**



#### 6.1.2.2   32.768 kHz ECO

The 32.768-kHz external crystal oscillator (32kHzECO) provides precision timing with minimal power consumption using an external 32.768-kHz watch crystal (see Figure 6-3). The 32kHzECO also connects directly to the sleep timer and provides the source for the RTC. The RTC uses a 1-second interrupt to implement the RTC functionality in firmware.

The oscillator works in two distinct power modes. This allows you to trade off power consumption with noise immunity from neighboring circuits. The GPIO pins connected to the external crystal and capacitors are fixed.

**Figure 6-3. 32kHzECO Block Diagram**
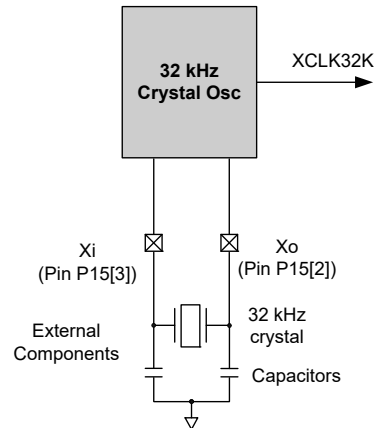


It is recommended that the external 32.768-kHz watch crystal have a load capacitance (CL) of 6 pF or 12.5 pF. Check the crystal manufacturer's datasheet. The two external capacitors, CL1 and CL2, are typically of the same value, and their total capacitance, CL1CL2 / (CL1 + CL2), including pin and trace capacitance, should equal the crystal CL value. For more information, refer to application note AN54439: PSoC 3 and PSoC 5 External Oscillators. See also pin capacitance specifications in the "GPIO" section on page 68.

#### 6.1.2.3   Digital System Interconnect

The DSI provides routing for clocks taken from external clock oscillators connected to I/O. The oscillators can also be generated within the device in the digital system and UDBs.

While the primary DSI clock input provides access to all clocking resources, up to eight other DSI clocks (internally or externally generated) may be routed directly to the eight digital clock dividers. This is only possible if there are multiple precision clock sources.

### 6.1.3   Clock Distribution

All seven clock sources are inputs to the central clock distribution system. The distribution system is designed to create multiple high precision clocks. These clocks are customized for the design's requirements and eliminate the common problems found with limited resolution prescalers attached to peripherals. The clock distribution system generates several types of clock trees.

- The system clock is used to select and supply the fastest clock in the system for general system clock requirements and clock synchronization of the PSoC device.
- Bus Clock 16-bit divider uses the system clock to generate the system's bus clock used for data transfers and the CPU. The CPU clock is directly derived from the bus clock.
- Eight fully programmable 16-bit clock dividers generate digital system clocks for general use in the digital system, as configured by the design's requirements. Digital system clocks can generate custom clocks derived from any of the seven clock sources for any purpose. Examples include baud rate generators, accurate PWM periods, and timer clocks, and many others. If more than eight digital clock dividers are required, the Universal Digital Blocks (UDBs) and fixed function timer/counter/PWMs can also generate clocks.

■ Four 16-bit clock dividers generate clocks for the analog system components that require clocking, such as the ADC. The analog clock dividers include skew control to ensure that critical analog events do not occur simultaneously with digital switching events. This is done to reduce analog system noise. Each clock divider consists of an 8-input multiplexer, a 16-bit clock divider (divide by 2 and higher) that generates ~50% duty cycle clocks, system clock resynchronization logic, and deglitch logic. The outputs from each digital clock tree can be routed into the digital system interconnect and then brought back into the clock system as an input, allowing clock chaining of up to 32 bits.

### 6.1.4 USB Clock Domain

The USB clock domain is unique in that it operates largely asynchronously from the main clock network. The USB logic contains a synchronous bus interface to the chip, while running on an asynchronous clock to process USB data. The USB logic requires a 48 MHz frequency. This frequency is generated from the doubled value of 24 MHz from internal oscillator, DSI signal, or crystal oscillator.

## 6.2 Power System

The power system consists of separate analog, digital, and I/O supply pins, labeled VDDA, VDDD, and VDDIOX, respectively. It also includes two internal 1.8 V regulators that provide the digital (VCCD) and analog (VCCA) supplies for the internal core logic. The output pins of the regulators (VCCD and VCCA) and the VDDIO pins must have capacitors connected as shown in Figure 6-4. The two VCCD pins must be shorted together, with as short a trace as possible, and connected to a 1 µF ±10% X5R capacitor. The power system also contains a sleep regulator, an I²C regulator, and a hibernate regulator.

**Figure 6-4. PSoC Power System**
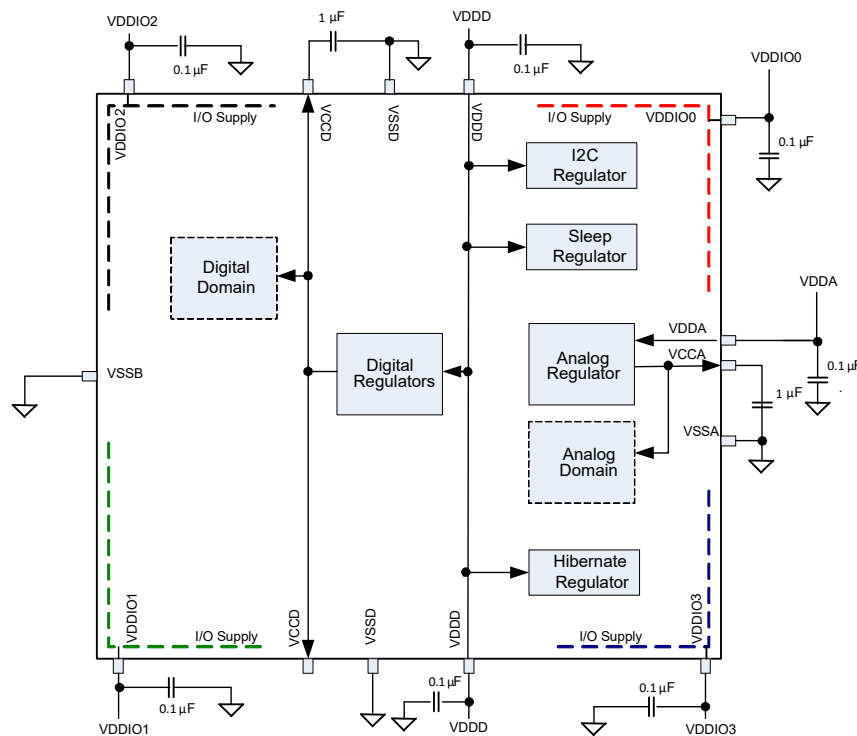


**Notes**

■ The two $V_{CCD}$ pins must be connected together with as short a trace as possible. A trace under the device is recommended, as shown in Figure 2-6.

■ You can power the device in internally regulated mode, where the voltage applied to the $V_{DDx}$ pins is as high as 5.5 V, and the internal regulators provide the core voltages. **In this mode, do not apply power to the $V_{CCx}$ pins, and do not tie the $V_{DDx}$ pins to the $V_{CCx}$ pins.**

■ You can also power the device in externally regulated mode, that is, by directly powering the $V_{CCD}$ and $V_{CCA}$ pins. In this configuration, the $V_{DDD}$ pins should be shorted to the $V_{CCD}$ pins and the $V_{DDA}$ pin should be shorted to the $V_{CCA}$ pin. The allowed supply range in this configuration is 1.71 V to 1.89 V. After power up in this configuration, the internal regulators are on by default, and should be disabled to reduce power consumption.

■ It is good practice to check the datasheets for your bypass capacitors, specifically the working voltage and the DC bias specifications. With some capacitors, the actual capacitance can decrease considerably when the DC bias ($V_{DDX}$ or $V_{CCX}$ in Figure 6-4) is a significant percentage of the rated working voltage.

### 6.2.1 Power Modes

PSoC 5LP devices have four different power modes, as shown in Table 6-2 and Table 6-3. The power modes allow a design to easily provide required functionality and processing power while simultaneously minimizing power consumption and maximizing battery life in low power and portable devices.

PSoC 5LP power modes, in order of decreasing power consumption are:
- Active
- Alternate Active
- Sleep
- Hibernate

Active is the main processing mode. Its functionality is configurable. Each power controllable subsystem is enabled or disabled by using separate power configuration template registers. In alternate active mode, fewer subsystems are enabled, reducing power. In sleep mode most resources are disabled regardless of the template settings. Sleep mode is optimized to provide timed sleep intervals and Real Time Clock functionality. The lowest power mode is hibernate, which retains register and SRAM state, but no clocks, and allows wakeup only from I/O pins. Figure 6-5 illustrates the allowable transitions between power modes. Sleep and hibernate modes should not be entered until all VDDIO supplies are at valid voltage levels.

**Table 6-2. Power Modes**

| Power Modes | Description | Entry Condition | Wakeup Source | Active Clocks | Regulator |
|---|---|---|---|---|---|
| Active | Primary mode of operation, all peripherals available (program-mable) | Wakeup, reset, manual register entry | Any interrupt | Any (program-mable) | All regulators available. Digital and analog regulators can be disabled if external regulation used. |
| Alternate Active | Similar to Active mode, and is typically configured to have fewer peripherals active to reduce power. One possible configuration is to use the UDBs for processing, with the CPU turned off | Manual register entry | Any interrupt | Any (program-mable) | All regulators available. Digital and analog regulators can be disabled if external regulation used. |
| Sleep | All subsystems automatically disabled | Manual register entry | Comparator, PICU, I$^2$C, RTC, CTW, LVD | ILO/kHzECO | Both digital and analog regulators buzzed. Digital and analog regulators can be disabled if external regulation used. |
| Hibernate | All subsystems automatically disabled. Lowest power consuming mode with all peripherals and internal regulators disabled, except hibernate regulator is enabled. Configuration and memory contents retained | Manual register entry | PICU | | Only hibernate regulator active. |

**Table 6-3. Power Modes Wakeup Time and Power Consumption**

| Sleep Modes | Wakeup Time | Current (Typ) | Code Execution | Digital Resources | Analog Resources | Clock Sources Available | Wakeup Sources | Reset Sources |
|---|---|---|---|---|---|---|---|---|
| Active | – | 3.1 mA[8] | Yes | All | All | All | – | All |
| Alternate Active | – | – | User defined | All | All | All | – | All |
| Sleep | <25 µs | 2 µA | No | I$^2$C | Comparator | ILO/kHzECO | Comparator, PICU, I$^2$C, RTC, CTW, LVD | XRES, LVD, WDR |
| Hibernate | <200 µs | 300 nA | No | None | None | None | PICU | XRES |

**Note**
8. Bus clock off. Execute from CPU instruction buffer at 6 MHz. See Table 11-2 on page 61

**Figure 6-5. Power Mode Transitions**
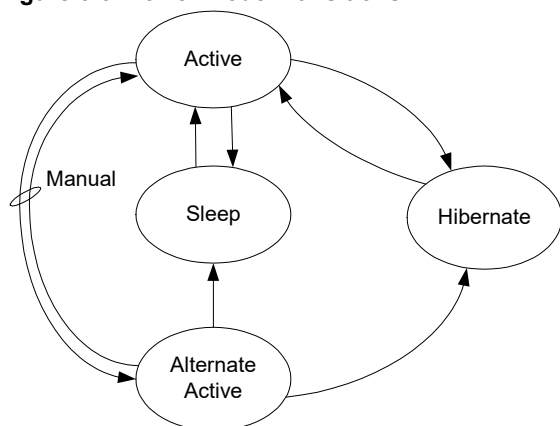


### 6.2.1.1 Active Mode

Active mode is the primary operating mode of the device. When in active mode, the active configuration template bits control which available resources are enabled or disabled. When a resource is disabled, the digital clocks are gated, analog bias currents are disabled, and leakage currents are reduced as appropriate. User firmware can dynamically control subsystem power by setting and clearing bits in the active configuration template. The CPU can disable itself, in which case the CPU is automatically reenabled at the next wakeup event.

When a wakeup event occurs, the global mode is always returned to active, and the CPU is automatically enabled, regardless of its template settings. Active mode is the default global power mode upon boot.

### 6.2.1.2 Alternate Active Mode

Alternate Active mode is very similar to Active mode. In alternate active mode, fewer subsystems are enabled, to reduce power consumption. One possible configuration is to turn off the CPU and flash, and run peripherals at full speed.

### 6.2.1.3 Sleep Mode

Sleep mode reduces power consumption when a resume time of 15 µs is acceptable. The wake time is used to ensure that the regulator outputs are stable enough to directly enter active mode.

### 6.2.1.4 Hibernate Mode

In hibernate mode nearly all of the internal functions are disabled. Internal voltages are reduced to the minimal level to keep vital systems alive. Configuration state is preserved in hibernate mode and SRAM memory is retained. GPIOs configured as digital outputs maintain their previous values and external GPIO pin interrupt settings are preserved. The device can only return from hibernate mode in response to an external I/O interrupt. The resume time from hibernate mode is less than 100 µs.

To achieve an extremely low current, the hibernate regulator has limited capacity. This limits the frequency of any signal present on the input pins; no GPIO should toggle at a rate greater than 10 kHz while in hibernate mode. If pins must be toggled at a high rate while in a low power mode, use sleep mode instead.

### 6.2.1.5 Wakeup Events

Wakeup events are configurable and can come from an interrupt or device reset. A wakeup event restores the system to active mode. Firmware enabled interrupt sources include internally generated interrupts, power supervisor, central timewheel, and I/O interrupts. Internal interrupt sources can come from a variety of peripherals, such as analog comparators and UDBs. The central timewheel provides periodic interrupts to allow the system to wake up, poll peripherals, or perform real-time functions. Reset event sources include the external reset pin (XRES), WDT, and precision reset (PRES).
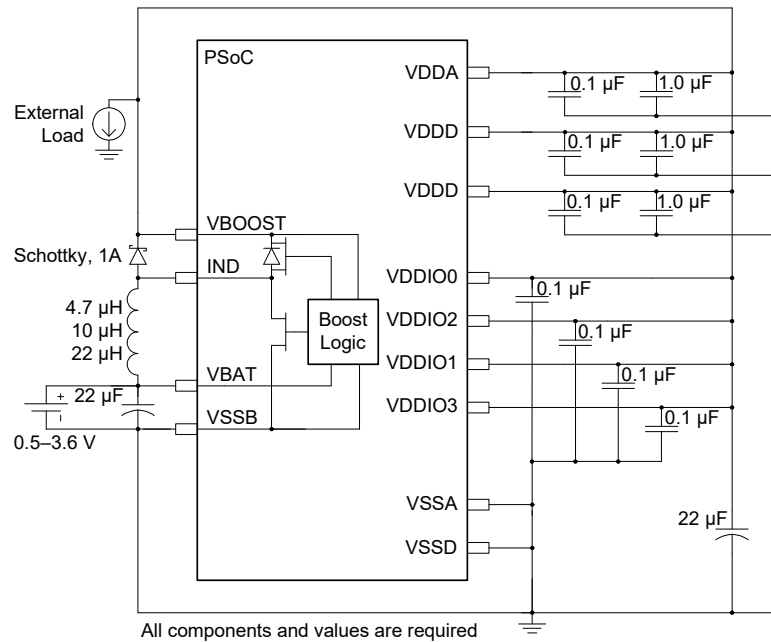
### 6.2.2 Boost Converter

Applications that use a supply voltage of less than 1.71 V, such as solar panels or single cell battery supplies, may use the on-chip boost converter to generate a minimum of 1.8 V supply voltage. The boost converter may also be used in any system that requires a higher operating voltage than the supply provides such as driving 5.0 V LCD glass in a 3.3 V system. With the addition of an inductor, Schottky diode, and capacitors, it produces a selectable output voltage sourcing enough current to operate the PSoC and other on-board components.

The boost converter accepts an input voltage $V_{BAT}$ from 0.5 V to 3.6 V, and can start up with $V_{BAT}$ as low as 0.5 V. The converter provides a user configurable output voltage of 1.8 to 5.0 V ($V_{OUT}$) in 100 mV increments. $V_{BAT}$ is typically less than $V_{OUT}$; if $V_{BAT}$ is greater than or equal to $V_{OUT}$, then $V_{OUT}$ will be slightly less than $V_{BAT}$ due to resistive losses in the boost converter. The block can deliver up to 50 mA ($I_{BOOST}$) depending on configuration to both the PSoC device and external components. The sum of all current sinks in the design including the PSoC device, PSoC I/O pin loads, and external component loads must be less than the $I_{BOOST}$ specified maximum current.

Four pins are associated with the boost converter: VBAT, VSSB, VBOOST, and IND. The boosted output voltage is sensed at the VBOOST pin and must be connected directly to the chip's supply inputs; VDDA, VDDD, and VDDIO if used to power the PSoC device.

The boost converter requires four components in addition to those required in a non-boost design, as shown in Figure 6-6 on page 29. A 22-µF capacitor ($C_{BAT}$) is required close to the VBAT pin to provide local bulk storage of the battery voltage and provide regulator stability. A diode between the battery and VBAT pin should not be used for reverse polarity protection because the diodes forward voltage drop reduces the $V_{BAT}$ voltage. Between the VBAT and IND pins, an inductor of 4.7 µH, 10 µH, or 22 µH is required. The inductor value can be optimized to increase the boost converter efficiency based on input voltage, output voltage, temperature, and current. Inductor size is determined by following the design guidance in this chapter and electrical specifications. The Inductor must be placed within 1 cm of the VBAT and IND pins and have a minimum saturation current of 750 mA. Between the IND and VBOOST pins a Schottky diode must be placed within 1 cm of the pins. The Schottky diode shall have a forward current rating of at least 1.0 A and a reverse voltage of at least 20 V. A 22 µF bulk capacitor ($C_{BOOST}$) must be connected close to VBOOST to provide regulator output stability. It is important to sum the total capacitance connected to the VBOOST pin and ensure the maximum $C_{BOOST}$ specification is not exceeded. All capacitors must be rated for a minimum of 10 V to minimize capacitive losses due to voltage de-rating.

**Figure 6-6. Application of Boost Converter powering PSoC device**



All components and values are required

The boost converter may also generate a supply that is not used directly by the PSoC device. An example of this use case is boosting a 1.8 V supply to 4.0 V to drive a white LED. If the boost converter is not supplying the PSoC devices $V_{DDA}$, $V_{DDD}$, and $V_{DDIO}$ it must comply with the same design rules as supplying the PSoC device, but with a change to the bulk capacitor requirements. A parallel arrangement 22 µF, 1.0 µF, and 0.1 µF capacitors are all required on the Vout supply and must be placed within 1 cm of the VBOOST pin to ensure regulator stability.

**Figure 6-7. Application of Boost Converter not powering PSoC device**
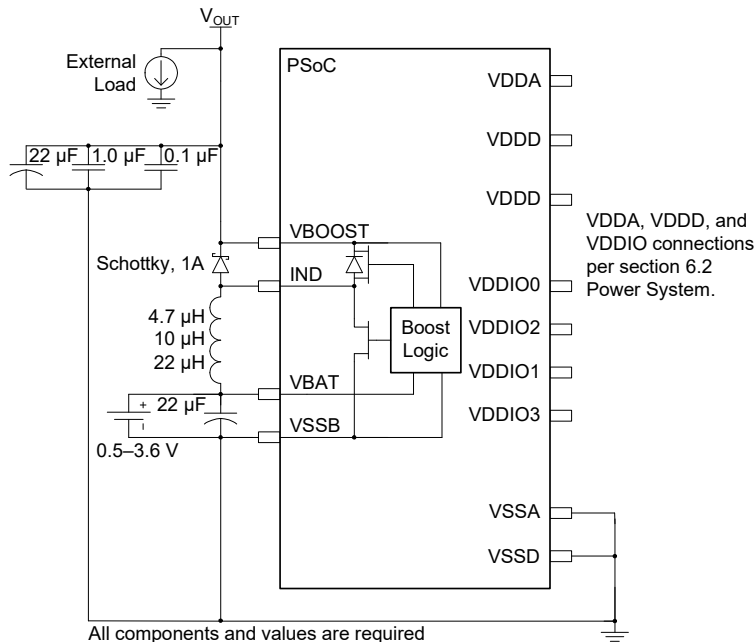


All components and values are required

The switching frequency is set to 400 kHz using an oscillator integrated into the boost converter. The boost converter can be operated in two different modes: active and standby. Active mode is the normal mode of operation where the boost regulator actively generates a regulated output voltage. In standby mode, most boost functions are disabled, thus reducing power

consumption of the boost circuit. Only minimal power is provided, typically < 5 μA to power the PSoC device in Sleep mode. The boost typically draws 250 μA in active mode and 25 μA in standby mode. The boost operating modes must be used in conjunction with chip power modes to minimize total power consumption. Table 6-4 lists the boost power modes available in different chip power modes.

**Table 6-4. Chip and Boost Power Modes Compatibility**

| Chip Power Modes | Boost Power Modes |
|---|---|
| Chip-active or alternate active mode | Boost must be operated in its active mode. |
| Chip-sleep mode | Boost can be operated in either active or standby mode. In boost standby mode, the chip must wake up periodically for boost active-mode refresh. |
| Chip-hibernate mode | Boost can be operated in its active mode. However, it is recommended not to use the boost in chip hibernate mode due to the higher current consumption in boost active mode. |

### 6.2.2.1 Boost Firmware Requirements

To ensure boost inrush current is within specification at startup, the **Enable Fast IMO During Startup** value must be unchecked in the PSoC Creator IDE. The **Enable Fast IMO During Startup** option is found in PSoC Creator in the design wide resources (cydwr) file **System** tab. Un-checking this option configures the device to run at 12 MHz vs 48 MHz during startup while configuring the device. The slower clock speed results in reduced current draw through the boost circuit.

### 6.2.2.2 Boost Design Process

Correct operation of the boost converter requires specific component values determined for each designs unique operating conditions. The $C_{BAT}$ capacitor, Inductor, Schottky diode, and $C_{BOOST}$ capacitor components are required with the values specified in the electrical specifications, Table 11-7 on page 66. The only variable component value is the inductor $L_{BOOST}$ which is primarily sized for correct operation of the boost across operating conditions and secondarily for efficiency. Additional operating region constraints exist for $V_{OUT}$, $V_{BAT}$, $I_{OUT}$, and $T_A$.

The following steps must be followed to determine boost converter operating parameters and $L_{BOOST}$ value.

1. Choose desired $V_{BAT}$, $V_{OUT}$, $T_A$, and $I_{OUT}$ operating condition ranges for the application.
2. Determine if $V_{BAT}$ and $V_{OUT}$ ranges fit the boost operating range based on the **$T_A$ range over $V_{BAT}$ and $V_{OUT}$** chart,

Figure 11-8 on page 66. If the operating ranges are not met, modify the operating conditions or use an external boost regulator.
3. Determine if the desired ambient temperature ($T_A$) range fits the ambient temperature operating range based on the **$T_A$ range over $V_{BAT}$ and $V_{OUT}$** chart, Figure 11-8 on page 66. If the temperature range is not met, modify the operating conditions and return to step 2, or use an external boost regulator.
4. Determine if the desired output current ($I_{OUT}$) range fits the output current operating range based on the **$I_{OUT}$ range over $V_{BAT}$ and $V_{OUT}$** chart, Figure 11-9 on page 66. If the output current range is not met, modify the operating conditions and return to step 2, or use an external boost regulator.
5. Find the allowed inductor values based on the **$L_{BOOST}$ values over $V_{BAT}$ and $V_{OUT}$** chart, Figure 11-10 on page 66.
6. Based on the allowed inductor values, inductor dimensions, inductor cost, boost efficiency, and $V_{RIPPLE}$ choose the optimum inductor value for the system. Boost efficiency and $V_{RIPPLE}$ typical values are provided in the **Efficiency vs $V_{BAT}$** and **$V_{RIPPLE}$ vs $V_{BAT}$** charts, Figure 11-11 on page 67 through Figure 11-14 on page 67. In general, if high efficiency and low $V_{RIPPLE}$ are most important, then the highest allowed inductor value should be used. If low inductor cost or small inductor size are most important, then one of the smaller allowed inductor values should be used. If the allowed inductor(s) efficiency, $V_{RIPPLE}$, cost or dimensions are not acceptable for the application than an external boost regulator should be used.

## 6.3 Reset

CY8C52LP has multiple internal and external reset sources available. The reset sources are:

- Power source monitoring: The analog and digital power voltages, VDDA, VDDD, VCCA, and VCCD are monitored in several different modes during power up, active mode, and sleep mode (buzzing). If any of the voltages goes outside predetermined ranges then a reset is generated. The monitors are programmable to generate an interrupt to the processor under certain conditions before reaching the reset thresholds.
- External: The device can be reset from an external source by pulling the reset pin (XRES) low. The XRES pin includes an internal pull up to VDDIO1. VDDD, VDDA, and VDDIO1 must all have voltage applied before the part comes out of reset.
- Watchdog timer: A watchdog timer monitors the execution of instructions by the processor. If the watchdog timer is not reset by firmware within a certain period of time, the watchdog timer generates a reset.
- Software: The device can be reset under program control.

**Figure 6-8. Resets**
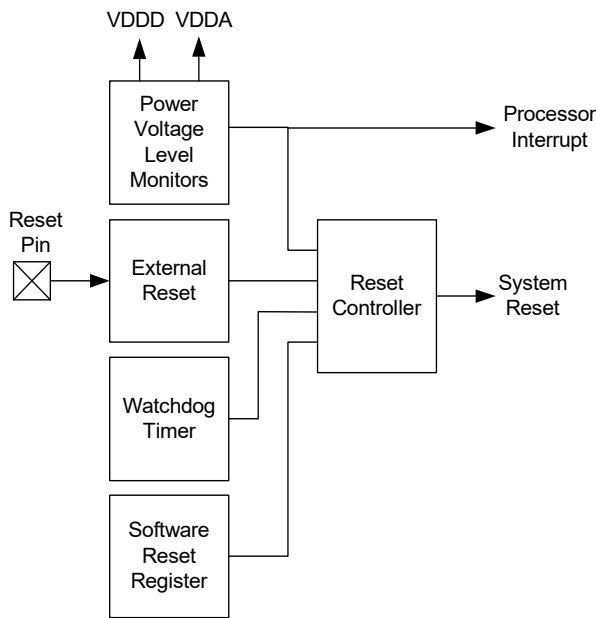


The term **system reset** indicates that the processor as well as analog and digital peripherals and registers are reset.

A reset status register shows some of the resets or power voltage monitoring interrupts. The program may examine this register to detect and report certain exception conditions. This register is cleared after a power-on reset. For details see the Technical Reference Manual.

### 6.3.1 Reset Sources

#### 6.3.1.1 Power Voltage Level Monitors

■ IPOR - Initial Power-on-Reset

At initial power on, IPOR monitors the power voltages $V_{DDD}$, $V_{DDA}$, $V_{CCD}$ and $V_{CCA}$. The trip level is not precise. It is set to approximately 1 volt (0.75 V to 1.45 V). This is below the lowest specified operating voltage but high enough for the internal circuits to be reset and to hold their reset state. The monitor generates a reset pulse that is at least 150 ns wide. It may be much wider if one or more of the voltages ramps up slowly.

After boot, the IPOR circuit is disabled and voltage supervision is handed off to the precise low-voltage reset (PRES) circuit.

■ PRES - Precise Low-Voltage Reset

This circuit monitors the outputs of the analog and digital internal regulators after power up. The regulator outputs are compared to a precise reference voltage. The response to a PRES trip is identical to an IPOR reset.

In normal operating mode, the program cannot disable the digital PRES circuit. The analog regulator can be disabled, which also disables the analog portion of the PRES. The PRES circuit is disabled automatically during sleep and hibernate modes, with one exception: During sleep mode the regulators are periodically activated (buzzed) to provide supervisory services and to reduce wakeup time. At these times the PRES circuit is also buzzed to allow periodic voltage monitoring.

■ ALVI, DLVI, AHVI - Analog/Digital Low Voltage Interrupt, Analog High Voltage Interrupt

Interrupt circuits are available to detect when VDDA and VDDD go outside a voltage range. For AHVI, $V_{DDA}$ is compared to a fixed trip level. For ALVI and DLVI, VDDA and VDDD are compared to trip levels that are programmable, as listed in Table 6-5. ALVI and DLVI can also be configured to generate a device reset instead of an interrupt.

**Table 6-5. Analog/Digital Low Voltage Interrupt, Analog High Voltage Interrupt**

| Interrupt | Supply | Normal Voltage Range | Available Trip Settings |
|---|---|---|---|
| DLVI | VDDD | 1.71 V-5.5 V | 1.70 V-5.45 V in 250 mV increments |
| ALVI | VDDA | 1.71 V-5.5 V | 1.70 V-5.45 V in 250 mV increments |
| AHVI | VDDA | 1.71 V-5.5 V | 5.75 V |

The monitors are disabled until after IPOR. During sleep mode these circuits are periodically activated (buzzed). If an interrupt occurs during buzzing then the system first enters its wakeup sequence. The interrupt is then recognized and may be serviced.

The buzz frequency is adjustable, and should be set to be less than the minimum time that any voltage is expected to be out of range. For details on how to adjust the buzz frequency, see the TRM.

#### 6.3.1.2 Other Reset Sources

■ XRES - External Reset

PSoC 5LP has a dedicated XRES pin, which holds the part in reset while held active (low). The response to an XRES is the same as to an IPOR reset.

The external reset is active low. It includes an internal pull up resistor. XRES is active during sleep and hibernate modes.

After XRES has been deasserted, at least 10 µs must elapse before it can be reasserted.

■ SRES - Software Reset

A reset can be commanded under program control by setting a bit in the software reset register. This is done either directly by the program or indirectly by DMA access. The response to a SRES is the same as after an IPOR reset.

Another register bit exists to disable this function.

■ WRES - Watchdog Timer Reset

The watchdog reset detects when the software program is no longer being executed correctly. To indicate to the watchdog timer that it is running correctly, the program must periodically reset the timer. If the timer is not reset before a user-specified amount of time, then a reset is generated.

**Note** IPOR disables the watchdog function. The program must enable the watchdog function at an appropriate point in the code by setting a register bit. When this bit is set, it cannot be cleared again except by an IPOR power on reset event.

## 6.4 I/O System and Routing

PSoC I/Os are extremely flexible. Every GPIO has analog and digital I/O capability. All I/Os have a large number of drive modes, which are set at POR. PSoC also provides up to four individual I/O voltage domains through the VDDIO pins.

There are two types of I/O pins on every device; those with USB provide a third type. Both GPIO and special I/O (SIO) provide similar digital functionality. The primary differences are their analog capability and drive strength. Devices that include USB also provide two USBIO pins that support specific USB functionality as well as limited GPIO capability.

All I/O pins are available for use as digital inputs and outputs for both the CPU and digital peripherals. In addition, all I/O pins can generate an interrupt. The flexible and advanced capabilities of the PSoC I/O, combined with any signal to any pin routability, greatly simplify circuit design and board layout. All GPIO pins can be used for analog input, CapSense[9], and LCD segment drive, while SIO pins are used for voltages in excess of VDDA and for programmable output voltages.

- Features supported by both GPIO and SIO:
  - User programmable port reset state
  - Separate I/O supplies and voltages for up to four groups of I/O
  - Digital peripherals use DSI to connect the pins
  - Input or output or both for CPU and DMA
  - Eight drive modes
  - Every pin can be an interrupt source configured as rising edge, falling edge or both edges. If required, level sensitive interrupts are supported through the DSI

- Dedicated port interrupt vector for each port
- Slew rate controlled digital output drive mode
- Access port control and configuration registers on either port basis or pin basis
- Separate port read (PS) and write (DR) data registers to avoid read modify write errors
- Special functionality on a pin by pin basis
- Additional features only provided on the GPIO pins:
  - LCD segment drive on LCD equipped devices
  - CapSense[9]
  - Analog input and output capability
  - Continuous 100 µA clamp current capability
  - Standard drive strength down to 1.71 V
- Additional features only provided on SIO pins:
  - Higher drive strength than GPIO
  - Hot swap capability (5 V tolerance at any operating VDD)
  - Programmable and regulated high input and output drive levels down to 1.2 V
  - No analog input, CapSense, or LCD capability
  - Over voltage tolerance up to 5.5 V
  - SIO can act as a general purpose analog comparator
- USBIO features:
  - Full speed USB 2.0 compliant I/O
  - Highest drive strength for general purpose use
  - Input, output, or both for CPU and DMA
  - Input, output, or both for digital peripherals
  - Digital output (CMOS) drive mode
  - Each pin can be an interrupt source configured as rising edge, falling edge, or both edges

**Note**
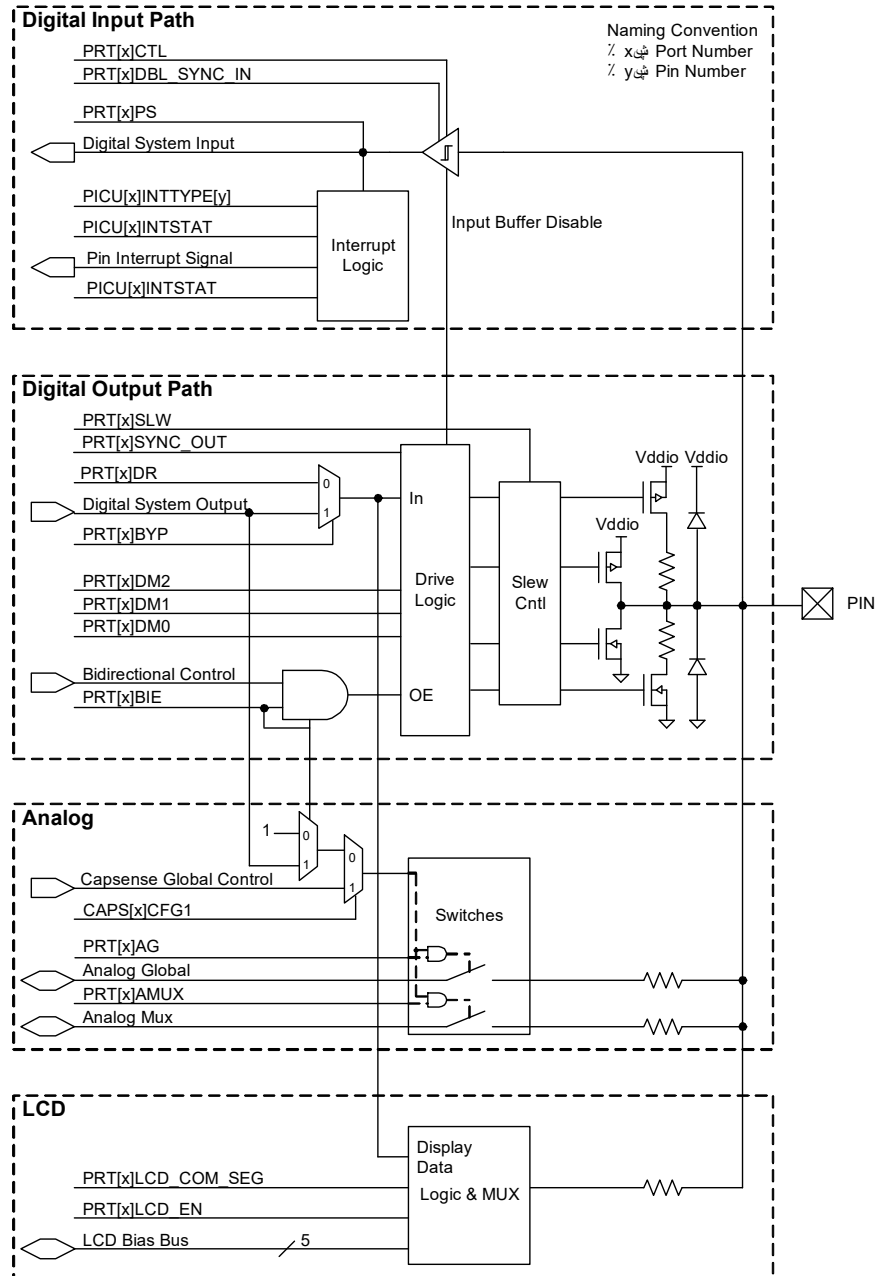9. GPIOs with opamp outputs are not recommended for use with CapSense.

**Figure 6-9. GPIO Block Diagram**
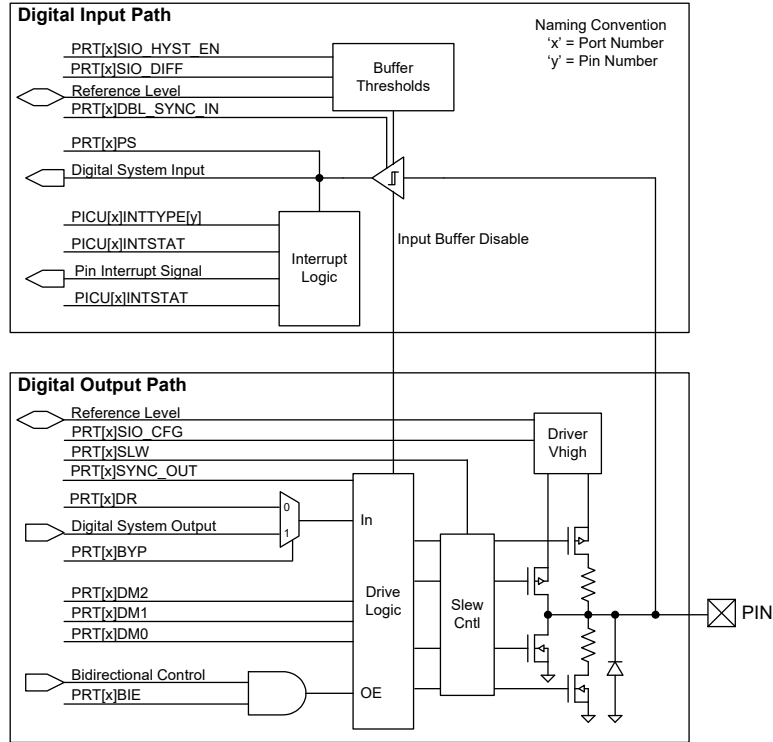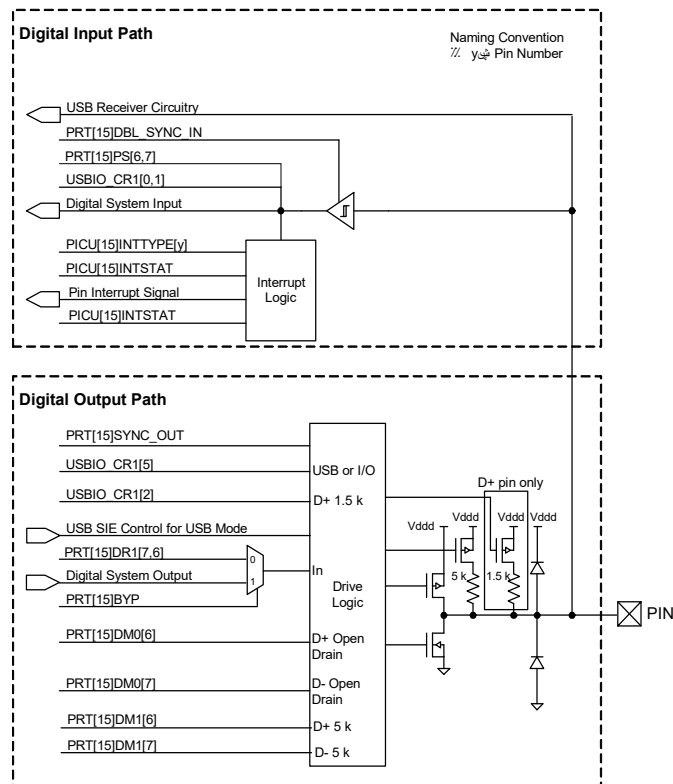
**Figure 6-10. SIO Input/Output Block Diagram**



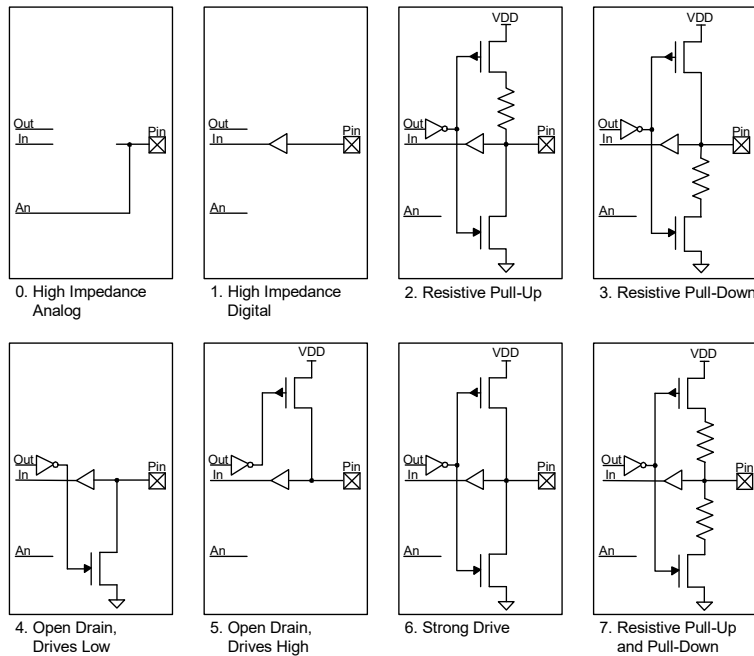**Figure 6-11. USBIO Block Diagram**

### 6.4.1 Drive Modes

Each GPIO and SIO pin is individually configurable into one of the eight drive modes listed in Table 6-6. Three configuration bits are used for each pin (DM[2:0]) and set in the PRTxDM[2:0] registers. Figure 6-12 depicts a simplified pin view based on each of the eight drive modes. Table 6-6 shows the I/O pin's drive state based on the port data register value or digital array signal if bypass mode is selected. Note that the actual I/O pin voltage is determined by a combination of the selected drive mode and the load at the pin. For example, if a GPIO pin is configured for resistive pull up mode and driven high while the pin is floating, the voltage measured at the pin is a high logic state. If the same GPIO pin is externally tied to ground then the voltage unmeasured at the pin is a low logic state.

**Figure 6-12. Drive Mode**



| 0. High Impedance Analog | 1. High Impedance Digital | 2. Resistive Pull-Up | 3. Resistive Pull-Down |
| 4. Open Drain, Drives Low | 5. Open Drain, Drives High | 6. Strong Drive | 7. Resistive Pull-Up and Pull-Down |

The Out connection is driven from either the Digital System (when the Digital Output terminal is connected) or the Data Register (when HW connection is disabled).
The In connection drives the Pin State register, and the Digital System if the Digital Input terminal is enabled and connected.
The An connection connects to the Analog System.

**Table 6-6. Drive Modes**

| Diagram | Drive Mode | PRTxDM2 | PRTxDM1 | PRTxDM0 | PRTxDR = 1 | PRTxDR = 0 |
|---------|-----------|---------|---------|---------|------------|------------|
| 0 | High impedance analog | 0 | 0 | 0 | High-Z | High-Z |
| 1 | High Impedance digital | 0 | 0 | 1 | High-Z | High-Z |
| 2 | Resistive pull-up[10] | 0 | 1 | 0 | Res High (5K) | Strong Low |
| 3 | Resistive pull-down[10] | 0 | 1 | 1 | Strong High | Res Low (5K) |
| 4 | Open drain, drives low | 1 | 0 | 0 | High-Z | Strong Low |
| 5 | Open drain, drive high | 1 | 0 | 1 | Strong High | High-Z |
| 6 | Strong drive | 1 | 1 | 0 | Strong High | Strong Low |
| 7 | Resistive pull-up and pull-down[10] | 1 | 1 | 1 | Res High (5K) | Res Low (5K) |

**Note**
10. Resistive pull up and pull down are not available with SIO in regulated output mode.

The USBIO pins (P15[7] and P15[6]), when enabled for I/O mode, have limited drive mode control. The drive mode is set using the PRT15.DM0[7, 6] register. A resistive pull option is also available at the USBIO pins, which can be enabled using the PRT15.DM1[7, 6] register. When enabled for USB mode, the drive mode control has no impact on the configuration of the USB pins. Unlike the GPIO and SIO configurations, the port wide configuration registers do not configure the USB drive mode bits. Table 6-7 shows the drive mode configuration for the USBIO pins.

**Table 6-7.  USBIO Drive Modes (P15[7] and P15[6])**

| PRT15.DM1[7,6] Pull up enable | PRT15.DM0[7,6] Drive Mode enable | PRT15.DR[7,6] = 1 | PRT15.DR[7,6] = 0 | Description |
|---|---|---|---|---|
| 0 | 0 | High Z | Strong Low | Open Drain, Strong Low |
| 0 | 1 | Strong High | Strong Low | Strong Outputs |
| 1 | 0 | Res High (5k) | Strong Low | Resistive Pull Up, Strong Low |
| 1 | 1 | Strong High | Strong Low | Strong Outputs |

■ High impedance analog

The default reset state with both the output driver and digital input buffer turned off. This prevents any current from flowing in the I/O's digital input buffer due to a floating voltage. This state is recommended for pins that are floating or that support an analog voltage. High impedance analog pins do not provide digital input functionality.

To achieve the lowest chip current in sleep modes, all I/Os must either be configured to the high impedance analog mode, or have their pins driven to a power supply rail by the PSoC device or by external circuitry.

■ High impedance digital

The input buffer is enabled for digital signal input. This is the standard high impedance (HiZ) state recommended for digital inputs.

■ Resistive pull up or resistive pull down

Resistive pull up or pull down, respectively, provides a series resistance in one of the data states and strong drive in the other. Pins can be used for digital input and output in these modes. Interfacing to mechanical switches is a common application for these modes. Resistive pull up and pull down are not available with SIO in regulated output mode.

■ Open drain, drives high and open drain, drives low

Open drain modes provide high impedance in one of the data states and strong drive in the other. Pins can be used for digital input and output in these modes. A common application for these modes is driving the I$^2$C bus signal lines.

■ Strong drive

Provides a strong CMOS output drive in either high or low state. This is the standard output mode for pins. Strong Drive mode pins must not be used as inputs under normal circumstances. This mode is often used to drive digital output signals or external FETs.

■ Resistive pull up and pull down

Similar to the resistive pull up and resistive pull down modes except the pin is always in series with a resistor. The high data state is pull up while the low data state is pull down. This mode is most often used when other signals that may cause shorts can drive the bus. Resistive pull up and pull down are not available with SIO in regulated output mode.

### 6.4.2  Pin Registers

Registers to configure and interact with pins come in two forms that may be used interchangeably.

All I/O registers are available in the standard port form, where each bit of the register corresponds to one of the port pins. This register form is efficient for quickly reconfiguring multiple port pins at the same time.

I/O registers are also available in pin form, which combines the eight most commonly used port register bits into a single register for each pin. This enables very fast configuration changes to individual pins with a single register write.

### 6.4.3  Bidirectional Mode

High speed bidirectional capability allows pins to provide both the high impedance digital drive mode for input signals and a second user selected drive mode such as strong drive (set using PRTxDM[2:0] registers) for output signals on the same pin, based on the state of an auxiliary control bus signal. The bidirectional capability is useful for processor busses and communications interfaces such as the SPI Slave MISO pin that requires dynamic hardware control of the output buffer.

The auxiliary control bus routes up to 16 UDB or digital peripheral generated output enable signals to one or more pins.

### 6.4.4  Slew Rate Limited Mode

GPIO and SIO pins have fast and slow output slew rate options for strong and open drain drive modes, not resistive drive modes. Because it results in reduced EMI, the slow edge rate option is recommended for signals that are not speed critical, generally less than 1 MHz. The fast slew rate is for signals between 1 MHz and 33 MHz. The slew rate is individually configurable for each pin, and is set by the PRTxSLW registers.

### 6.4.5  Pin Interrupts

All GPIO and SIO pins are able to generate interrupts to the system. All eight pins in each port interface to their own port interrupt control unit (PICU) and associated interrupt vector. Each pin of the port is independently configurable to detect rising edge, falling edge, both edge interrupts, or to not generate an interrupt.

Depending on the configured mode for each pin, each time an interrupt event occurs on a pin, its corresponding status bit of the interrupt status register is set to "1" and an interrupt request is sent to the interrupt controller. Each PICU has its own interrupt vector in the interrupt controller and the pin status register providing easy determination of the interrupt source down to the pin level.

Port pin interrupts remain active in all sleep modes allowing the PSoC device to wake from an externally generated interrupt.

While level sensitive interrupts are not directly supported; UDBs provide this functionality to the system when needed.

### 6.4.6  Input Buffer Mode

GPIO and SIO input buffers can be configured at the port level for the default CMOS input thresholds or the optional LVTTL input thresholds. All input buffers incorporate Schmitt triggers for input hysteresis. Additionally, individual pin input buffers can be disabled in any drive mode.

### 6.4.7  I/O Power Supplies

Up to four I/O pin power supplies are provided depending on the device and package. Each I/O supply must be less than or equal to the voltage on the chip's analog (VDDA) pin. This feature allows you to provide different I/O voltage levels for different pins on the device. Refer to the specific device package pinout to determine VDDIO capability for a given port and pin.

The SIO port pins support an additional regulated high output capability, as described in Adjustable Output Level.

### 6.4.8  Analog Connections

These connections apply only to GPIO pins. All GPIO pins may be used as analog inputs or outputs. The analog voltage present on the pin must not exceed the VDDIO supply voltage to which the GPIO belongs. Each GPIO may connect to one of the analog global busses or to one of the analog mux buses to connect any pin to any internal analog resource such as ADC or comparators. In addition, one select pin provides direct connection to the high current DAC.

### 6.4.9  CapSense

This section applies only to GPIO pins. All GPIO pins may be used to create CapSense buttons and sliders[11]. See the "CapSense" section on page 54 for more information.

### 6.4.10  LCD Segment Drive

This section applies only to GPIO pins. All GPIO pins may be used to generate Segment and Common drive signals for direct glass drive of LCD glass. See the "LCD Direct Drive" section on page 53 for details.

### 6.4.11  Adjustable Output Level

This section applies only to SIO pins. SIO port pins support the ability to provide a regulated high output level for interface to external signals that are lower in voltage than the SIO's respective VDDIO. SIO pins are individually configurable to output either the standard VDDIO level or the regulated output, which is based on an internally generated reference. Typically the voltage DAC (VDAC) is used to generate the reference (see Figure 6-13). The DAC on page 54 has more details on VDAC use and reference routing to the SIO pins. Resistive pull up and pull down drive modes are not available with SIO in regulated output mode.
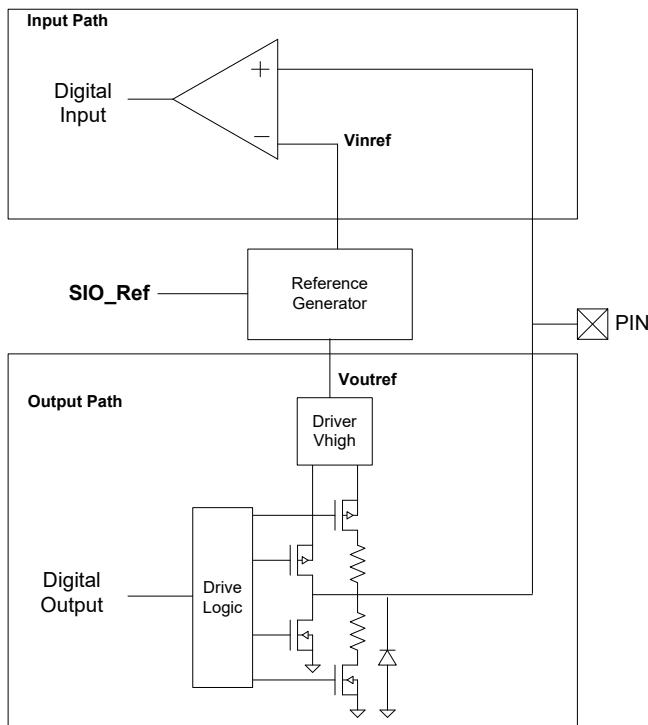
### 6.4.12  Adjustable Input Level

This section applies only to SIO pins. SIO pins by default support the standard CMOS and LVTTL input levels but also support a differential mode with programmable levels. SIO pins are grouped into pairs. Each pair shares a reference generator block which, is used to set the digital input buffer reference level for interface to external signals that differ in voltage from VDDIO. The reference sets the pins voltage threshold for a high logic level (see Figure 6-13). Available input thresholds are:

- $0.5 \times$ VDDIO
- $0.4 \times$ VDDIO
- $0.5 \times$ VREF
- VREF

Typically the voltage DAC (VDAC) generates the VREF reference. The DAC on page 54 has more details on VDAC use and reference routing to the SIO pins.

**Note**
11. GPIOs with opamp outputs are not recommended for use with CapSense.

## Figure 6-13. SIO Reference for Input and Output



### 6.4.13 SIO as Comparator

This section applies only to SIO pins. The adjustable input level feature of the SIOs as explained in the Adjustable Input Level section can be used to construct a comparator. The threshold for the comparator is provided by the SIO's reference generator. The reference generator has the option to set the analog signal routed through the analog global line as threshold for the comparator. Note that a pair of SIO pins share the same threshold.

The digital input path in Figure 6-10 on page 34 illustrates this functionality. In the figure, 'Reference level' is the analog signal routed through the analog global. The hysteresis feature can also be enabled for the input buffer of the SIO, which increases noise immunity for the comparator.

### 6.4.14 Hot Swap

This section applies only to SIO pins. SIO pins support 'hot swap' capability to plug into an application without loading the signals that are connected to the SIO pins even when no power is applied to the PSoC device. This allows the unpowered PSoC to maintain a high impedance load to the external device while also preventing the PSoC from being powered through a SIO pin's protection diode.

Powering the device up or down while connected to an operational I2C bus may cause transient states on the SIO pins. The overall I2C bus design should take this into account.

### 6.4.15 Over Voltage Tolerance

All I/O pins provide an over voltage tolerance feature at any operating VDD.

■ There are no current limitations for the SIO pins as they present a high impedance load to the external circuit.

■ The GPIO pins must be limited to 100 µA using a current limiting resistor. GPIO pins clamp the pin voltage to approximately one diode above the VDDIO supply.

■ In case of a GPIO pin configured for analog input/output, the analog voltage on the pin must not exceed the VDDIO supply voltage to which the GPIO belongs.

A common application for this feature is connection to a bus such as I²C where different devices are running from different supply voltages. In the I²C case, the PSoC chip is configured into the Open Drain, Drives Low mode for the SIO pin. This allows an external pull up to pull the I²C bus voltage above the PSoC pin supply. For example, the PSoC chip could operate at 1.8 V, and an external device could run from 5 V. Note that the SIO pin's VIH and VIL levels are determined by the associated VDDIO supply pin.

The SIO pin must be in one of the following modes: 0 (high impedance analog), 1 (high impedance digital), or 4 (open drain drives low). See Figure 6-12 for details. Absolute maximum ratings for the device must be observed for all I/O pins.

### 6.4.16 Reset Configuration

While reset is active all I/Os are reset to and held in the High Impedance Analog state. After reset is released, the state can be reprogrammed on a port-by-port basis to pull down or pull up. To ensure correct reset operation, the port reset configuration data is stored in special nonvolatile registers. The stored reset data is automatically transferred to the port reset configuration registers at reset release.

### 6.4.17 Low Power Functionality

In all low power modes the I/O pins retain their state until the part is awakened and changed or reset. To awaken the part, use a pin interrupt, because the port interrupt logic continues to function in all low power modes.

### 6.4.18 Special Pin Functionality

Some pins on the device include additional special functionality in addition to their GPIO or SIO functionality. The specific special function pins are listed in "Pinouts" on page 6. The special features are:

■ Digital
  □ 4- to 25-MHz crystal oscillator
  □ 32.768-kHz crystal oscillator
  □ Wake from sleep on I²C address match. Any pin can be used for I²C if wake from sleep is not required.
  □ JTAG interface pins
  □ SWD interface pins
  □ SWV interface pins
  □ TRACEPORT interface pins
  □ External reset

■ Analog
  □ High current IDAC output
  □ External reference inputs

### 6.4.19 JTAG Boundary Scan

The device supports standard JTAG boundary scan chains on all pins for board level test.

# 7. Digital Subsystem

The digital programmable system creates application specific combinations of both standard and advanced digital peripherals and custom logic functions. These peripherals and logic are then interconnected to each other and to any pin on the device, providing a high level of design flexibility and IP security.

The features of the digital programmable system are outlined here to provide an overview of capabilities and architecture. You do not need to interact directly with the programmable digital system at the hardware and register level. PSoC Creator provides a high level schematic capture graphical interface to automatically place and route resources similar to PLDs.

The main components of the digital programmable system are:

■ Universal digital blocks (UDB) - These form the core functionality of the digital programmable system. UDBs are a collection of uncommitted logic (PLD) and structural logic (Datapath) optimized to create all common embedded peripherals and customized functionality that are application or design specific.

■ Universal digital block array - UDB blocks are arrayed within a matrix of programmable interconnect. The UDB array structure is homogeneous and allows for flexible mapping of digital functions onto the array. The array supports extensive and flexible routing interconnects between UDBs and the digital system interconnect.

■ Digital system interconnect (DSI) - Digital signals from UDBs, fixed function peripherals, I/O pins, interrupts, DMA, and other system core signals are attached to the DSI to implement full featured device connectivity. The DSI allows any digital function to any pin or other feature routability when used with the UDB array.

**Figure 7-1. CY8C52LP Digital Programmable Architecture**
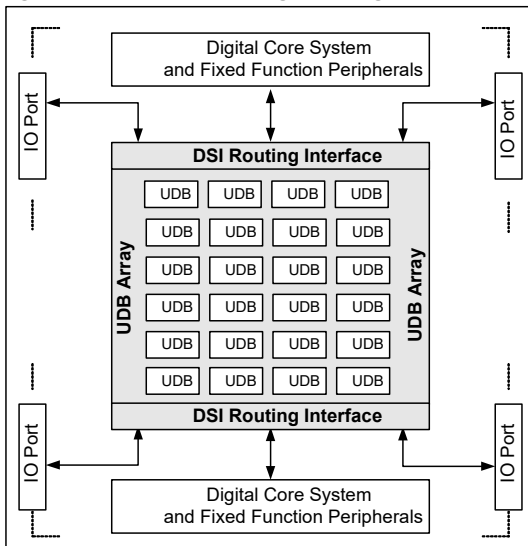


## 7.1 Example Peripherals

The flexibility of the CY8C52LP family's UDBs and analog blocks allow you to create a wide range of components (peripherals). The most common peripherals were built and characterized by Cypress and are shown in the PSoC Creator component catalog. However, you may also create your own custom components using PSoC Creator. Using PSoC Creator, you may also create their own components for reuse within their organization, for example sensor interfaces, proprietary algorithms, and display interfaces.

The number of components available through PSoC Creator is too numerous to list in the datasheet, and the list is always growing. An example of a component available for use in CY8C52LP family, but, not explicitly called out in this datasheet is the UART component.

### 7.1.1 Example Digital Components

The following is a sample of the digital components available in PSoC Creator for the CY8C52LP family. The exact amount of hardware resources (UDBs, routing, RAM, flash) used by a component varies with the features selected in PSoC Creator for the component.

■ Communications
  □ I$^2$C
  □ UART
  □ SPI
■ Functions
  □ EMIF
  □ PWMs
  □ Timers
  □ Counters
  □
■ Logic
  □ NOT
  □ OR
  □ XOR
  □ AND

### 7.1.2 Example Analog Components

The following is a sample of the analog components available in PSoC Creator for the CY8C52LP family. The exact amount of hardware resources (SC/CT blocks, routing, RAM, flash) used by a component varies with the features selected in PSoC Creator for the component.

■ ADC
  □ Successive Approximation (SAR ADC)
■ DACs
  □ Current
  □ Voltage
  □ PWM
■ Comparators

### 7.1.3  Example System Function Components

The following is a sample of the system function components available in PSoC Creator for the CY8C52LP family. The exact amount of hardware resources (UDBs, routing, RAM, flash) used by a component varies with the features selected in PSoC Creator for the component.

- CapSense
- LCD drive
- LCD control
- Filters

### 7.1.4  Designing with PSoC Creator

#### 7.1.4.1  More Than a Typical IDE

A successful design tool allows for the rapid development and deployment of both simple and complex designs. It reduces or eliminates any learning curve. It makes the integration of a new design into the production stream straightforward.

PSoC Creator is that design tool.

PSoC Creator is a full featured Integrated Development Environment (IDE) for hardware and software design. It is optimized specifically for PSoC devices and combines a modern, powerful software development platform with a sophisticated graphical design tool. This unique combination of tools makes PSoC Creator the most flexible embedded design platform available.

Graphical design entry simplifies the task of configuring a particular part. You can select the required functionality from an extensive catalog of components and place it in your design. All components are parameterized and have an editor dialog that allows you to tailor functionality to your needs.

PSoC Creator automatically configures clocks and routes the I/O to the selected pins and then generates APIs to give the application complete control over the hardware. Changing the PSoC device configuration is as simple as adding a new component, setting its parameters, and rebuilding the project.

At any stage of development you are free to change the hardware configuration and even the target processor. To retarget your application (hardware and software) to new devices, even from 8- to 32-bit families, just select the new device and rebuild.

You also have the ability to change the C compiler and evaluate an alternative. Components are designed for portability and are validated against all devices, from all families, and against all supported tool chains. Switching compilers is as easy as editing the from the project options and rebuilding the application with no errors from the generated APIs or boot code.

#### 7.1.4.2  Component Catalog

The component catalog is a repository of reusable design elements that select device functionality and customize your PSoC device. It is populated with an impressive selection of content; from simple primitives such as logic gates and device registers, through the digital timers, counters and PWMs, plus analog components such as ADC and DAC, and communication protocols such as I$^2$C and USB. See "Example Peripherals" section on page 39 for more details about available peripherals. All content is fully characterized and carefully documented in datasheets with code examples, AC/DC specifications, and user code ready APIs.

#### 7.1.4.3  Design Reuse

The symbol editor gives you the ability to develop reusable components that can significantly reduce future design time. Just draw a symbol and associate that symbol with your proven design. PSoC Creator allows for the placement of the new symbol anywhere in the component catalog along with the content provided by Cypress. You can then reuse your content as many times as you want, and in any number of projects, without ever having to revisit the details of the implementation.

#### 7.1.4.4  Software Development

Anchoring the tool is a modern, highly customizable user interface. It includes project management and integrated editors for C and assembler source code, as well the design entry tools.

Project build control leverages compiler technology from top commercial vendors such as Arm Limited, Keil™, and CodeSourcery (GNU). Free versions of Keil C51 and GNU C Compiler (GCC) for Arm, with no restrictions on code size or end product distribution, are included with the tool distribution. Upgrading to more optimizing compilers is a snap with support for the professional Keil C51 product and Arm RealView™ compiler.

#### 7.1.4.5  Nonintrusive Debugging

With JTAG (4-wire) and SWD (2-wire) debug connectivity available on all devices, the PSoC Creator debugger offers full control over the target device with minimum intrusion. Breakpoints and code execution commands are all readily available from toolbar buttons and an impressive lineup of windows—register, locals, watch, call stack, memory and peripherals – make for an unparalleled level of visibility into the system.

PSoC Creator contains all the tools necessary to complete a design, and then to maintain and extend that design for years to come. All steps of the design flow are carefully integrated and optimized for ease-of-use and to maximize productivity.

## 7.2 Universal Digital Block

The universal digital block (UDB) represents an evolutionary step to the next generation of PSoC embedded digital peripheral functionality. The architecture in first generation PSoC digital blocks provides coarse programmability in which a few fixed functions with a small number of options are available. The new UDB architecture is the optimal balance between configuration granularity and efficient implementation. A cornerstone of this approach is to provide the ability to customize the devices digital operation to match application requirements.

To achieve this, UDBs consist of a combination of uncommitted logic (PLD), structured logic (datapath), and a flexible routing scheme to provide interconnect between these elements, I/O connections, and other peripherals. UDB functionality ranges from simple self contained functions that are implemented in one UDB, or even a portion of a UDB (unused resources are available for other functions), to more complex functions that require multiple UDBs. Examples of basic functions are timers, counters, CRC generators, PWMs, dead band generators, and communications functions, such as UARTs, SPI, and I²C. Also, the PLD blocks and connectivity provide full featured general purpose programmable logic within the limits of the available resources.

**Figure 7-2. UDB Block Diagram**
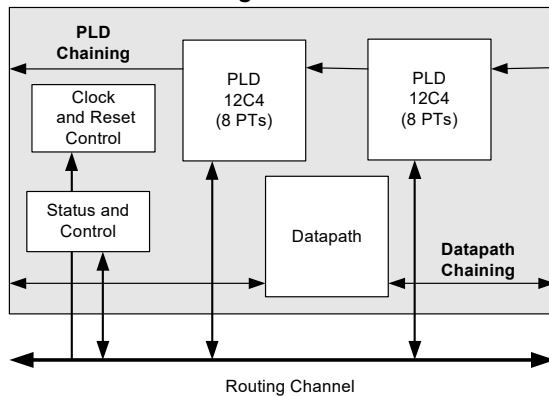


The main component blocks of the UDB are:
- **PLD blocks:** There are two small PLDs per UDB. These blocks take inputs from the routing array and form registered or combinational sum-of-products logic. PLDs are used to implement state machines, state bits, and combinational logic equations. PLD configuration is automatically generated from graphical primitives.
- **Datapath module:** This 8-bit wide datapath contains structured logic to implement a dynamically configurable ALU, a variety of compare configurations and condition generation. This block also contains input/output FIFOs, which are the primary parallel data interface between the CPU/DMA system and the UDB.
- **Status and control module:** The primary role of this block is to provide a way for CPU firmware to interact and synchronize with UDB operation.
- **Clock and reset module:** This block provides the UDB clocks and reset selection and control.

### 7.2.1 PLD Module

The primary purpose of the PLD blocks is to implement logic expressions, state machines, sequencers, look up tables, and decoders. In the simplest use model, consider the PLD blocks as a standalone resource onto which general purpose RTL is synthesized and mapped. The more common and efficient use model is to create digital functions from a combination of PLD and datapath blocks, where the PLD implements only the random logic and state portion of the function while the datapath (ALU) implements the more structured elements.

**Figure 7-3. PLD 12C4 Structure**
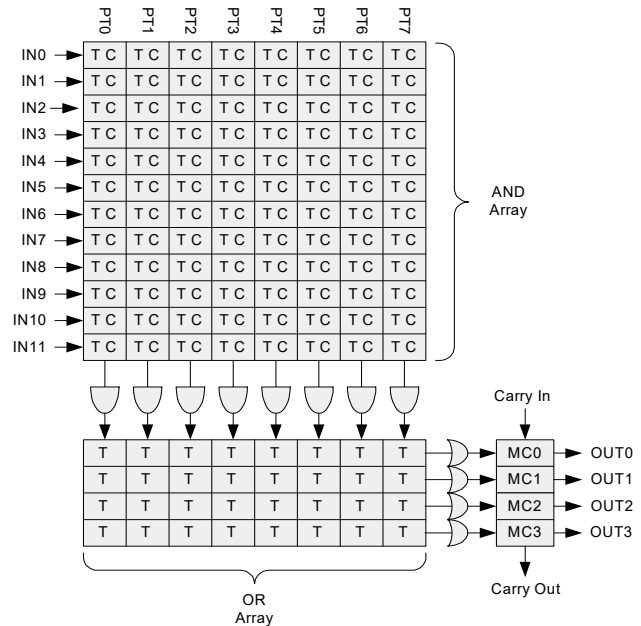


One 12C4 PLD block is shown in Figure 7-3. This PLD has 12 inputs, which feed across eight product terms. Each product term (AND function) can be from 1 to 12 inputs wide, and in a given product term, the true (T) or complement (C) of each input can be selected. The product terms are summed (OR function) to create the PLD outputs. A sum can be from 1 to 8 product terms wide. The 'C' in 12C4 indicates that the width of the OR gate (in this case 8) is constant across all outputs (rather than variable as in a 22V10 device). This PLA like structure gives maximum flexibility and insures that all inputs and outputs are permutable for ease of allocation by the software tools. There are two 12C4 PLDs in each UDB.

### 7.2.2 Datapath Module

The datapath contains an 8-bit single cycle ALU, with associated compare and condition generation logic. This datapath block is optimized to implement embedded functions, such as timers, counters, integrators, PWMs, PRS, CRC, shifters and dead band generators and many others.

Figure 7-4. Datapath Top Level
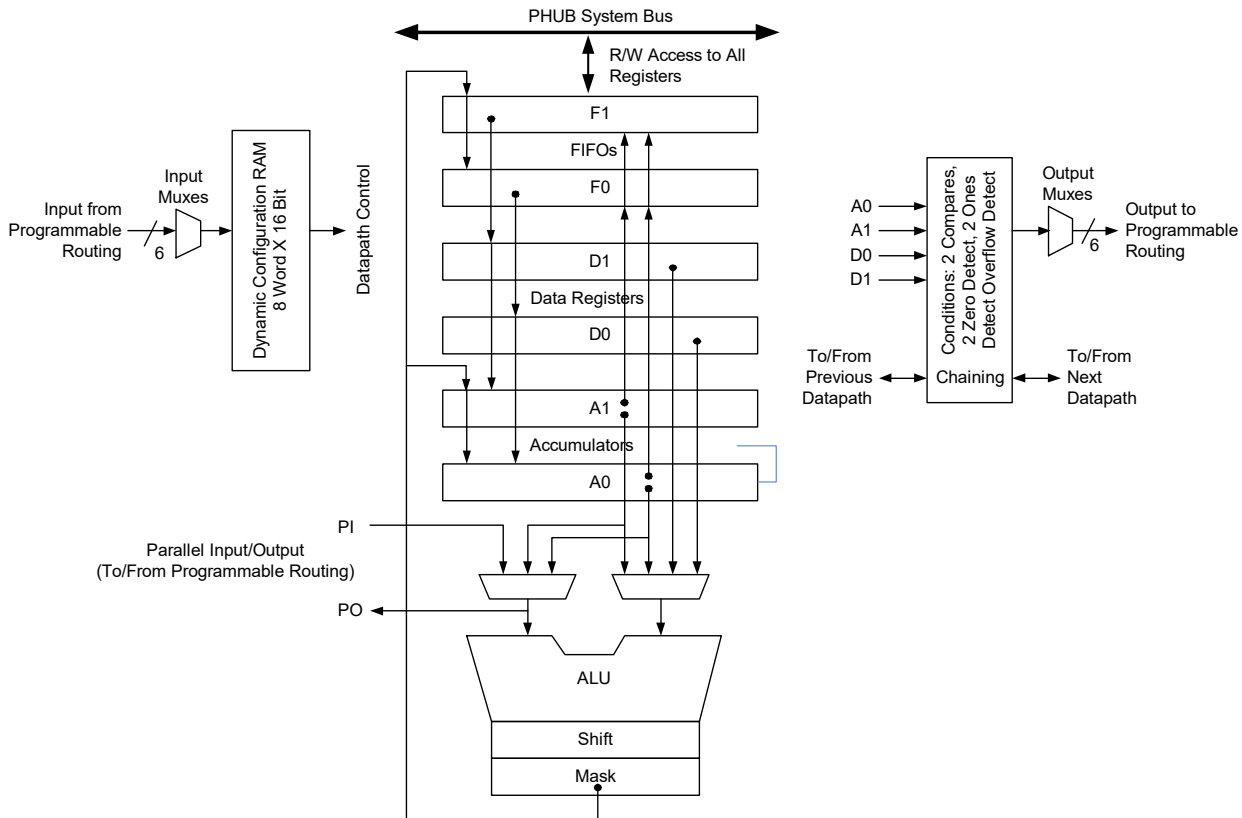
### 7.2.2.1 Working Registers

The datapath contains six primary working registers, which are accessed by CPU firmware or DMA during normal operation.

**Table 7-1. Working Datapath Registers**

| Name | Function | Description |
|------|----------|-------------|
| A0 and A1 | Accumulators | These are sources and sinks for the ALU and also sources for the compares. |
| D0 and D1 | Data Registers | These are sources for the ALU and sources for the compares. |
| F0 and F1 | FIFOs | These are the primary interface to the system bus. They can be a data source for the data registers and accumulators or they can capture data from the accumulators or ALU. Each FIFO is four bytes deep. |

### 7.2.2.2 Dynamic Configuration RAM

Dynamic configuration is the ability to change the datapath function and internal configuration on a cycle-by-cycle basis, under sequencer control. This is implemented using the 8-word × 16-bit configuration RAM, which stores eight unique 16-bit wide configurations. The address input to this RAM controls the sequence, and can be routed from any block connected to the

UDB routing matrix, most typically PLD logic, I/O pins, or from the outputs of this or other datapath blocks.

### ALU

The ALU performs eight general purpose functions. They are:

- Increment
- Decrement
- Add
- Subtract
- Logical AND
- Logical OR
- Logical XOR
- Pass, used to pass a value through the ALU to the shift register, mask, or another UDB register

Independent of the ALU operation, these functions are available:

- Shift left
- Shift right
- Nibble swap
- Bitwise OR mask

### 7.2.2.3 Conditionals

Each datapath has two compares, with bit masking options. Compare operands include the two accumulators and the two data registers in a variety of configurations. Other conditions include zero detect, all ones detect, and overflow. These conditions are the primary datapath outputs, a selection of which can be driven out to the UDB routing matrix. Conditional computation can use the built in chaining to neighboring UDBs to operate on wider data widths without the need to use routing resources.

### 7.2.2.4 Variable MSB

The most significant bit of an arithmetic and shift function can be programmatically specified. This supports variable width CRC and PRS functions, and in conjunction with ALU output masking, can implement arbitrary width timers, counters and shift blocks.

### 7.2.2.5 Built in CRC/PRS

The datapath has built in support for single cycle Cyclic Redundancy Check (CRC) computation and Pseudo Random Sequence (PRS) generation of arbitrary width and arbitrary polynomial. CRC/PRS functions longer than 8 bits may be implemented in conjunction with PLD logic, or built in chaining may be use to extend the function into neighboring UDBs.

### 7.2.2.6 Input/Output FIFOs

Each datapath contains two four-byte deep FIFOs, which can be independently configured as an input buffer (system bus writes to the FIFO, datapath internal reads the FIFO), or an output buffer (datapath internal writes to the FIFO, the system bus reads from the FIFO). The FIFOs generate status that are selectable as datapath outputs and can therefore be driven to the routing, to interact with sequencers, interrupts, or DMA.

**Figure 7-5. Example FIFO Configurations**
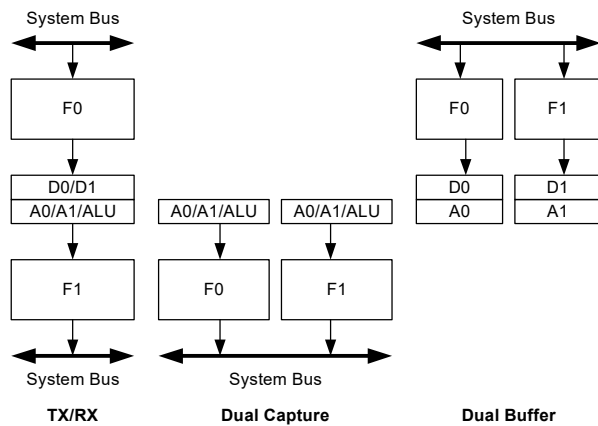


### 7.2.2.7 Chaining

The datapath can be configured to chain conditions and signals such as carries and shift data with neighboring datapaths to create higher precision arithmetic, shift, CRC/PRS functions.

### 7.2.2.8 Time Multiplexing

In applications that are over sampled, or do not need high clock rates, the single ALU block in the datapath can be efficiently shared with two sets of registers and condition generators. Carry and shift out data from the ALU are registered and can be selected as inputs in subsequent cycles. This provides support for 16-bit functions in one (8-bit) datapath.

### 7.2.2.9 Datapath I/O

There are six inputs and six outputs that connect the datapath to the routing matrix. Inputs from the routing provide the configuration for the datapath operation to perform in each cycle, and the serial data inputs. Inputs can be routed from other UDB blocks, other device peripherals, device I/O pins, and so on. The outputs to the routing can be selected from the generated conditions, and the serial data outputs. Outputs can be routed to other UDB blocks, device peripherals, interrupt and DMA controller, I/O pins, and so on.

### 7.2.3 Status and Control Module

The primary purpose of this circuitry is to coordinate CPU firmware interaction with internal UDB operation.

**Figure 7-6. Status and Control Registers**
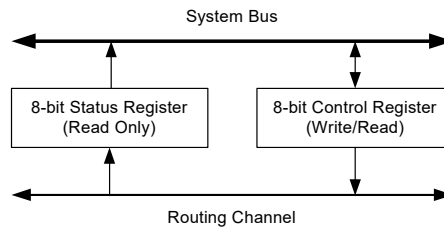


The bits of the control register, which may be written to by the system bus, are used to drive into the routing matrix, and thus provide firmware with the opportunity to control the state of UDB processing. The status register is read-only and it allows internal UDB state to be read out onto the system bus directly from internal routing. This allows firmware to monitor the state of UDB processing. Each bit of these registers has programmable connections to the routing matrix and routing connections are made depending on the requirements of the application.

### 7.2.3.10 Usage Examples

As an example of control input, a bit in the control register can be allocated as a function enable bit. There are multiple ways to enable a function. In one method the control bit output would be routed to the clock control block in one or more UDBs and serve as a clock enable for the selected UDB blocks. A status example is a case where a PLD or datapath block generated a condition, such as a "compare true" condition that is captured and latched by the status register and then read (and cleared) by CPU firmware.

### 7.2.3.11 Clock Generation

Each subcomponent block of a UDB including the two PLDs, the datapath, and Status and Control, has a clock selection and control block. This promotes a fine granularity with respect to allocating clocking resources to UDB component blocks and allows unused UDB resources to be used by other functions for maximum system efficiency.

## 7.3 UDB Array Description

Figure 7-7 shows an example of a 16 UDB array. In addition to the array core, there are a DSI routing interfaces at the top and bottom of the array. Other interfaces that are not explicitly shown include the system interfaces for bus and clock distribution. The UDB array includes multiple horizontal and vertical routing channels each comprised of 96 wires. The wire connections to UDBs, at horizontal/vertical intersection and at the DSI interface are highly permutable providing efficient automatic routing in PSoC Creator. Additionally the routing allows wire by wire segmentation along the vertical and horizontal routing to further increase routing flexibility and capability.

**Figure 7-7. Digital System Interface Structure**
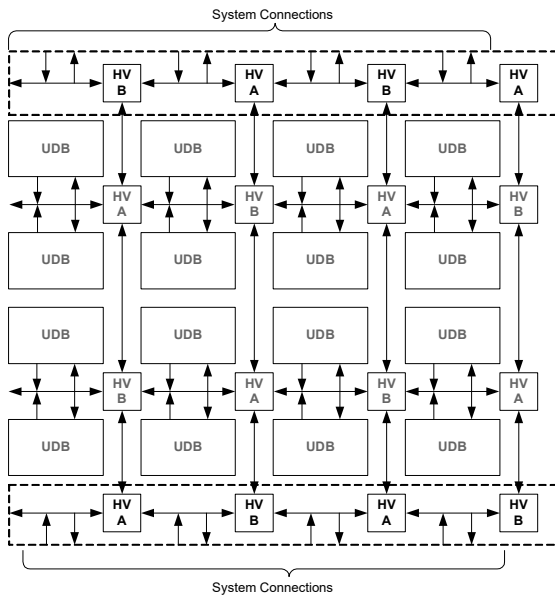


### 7.3.1 UDB Array Programmable Resources

Figure 7-8 shows an example of how functions are mapped into a bank of 16 UDBs. The primary programmable resources of the UDB are two PLDs, one datapath and one status/control register. These resources are allocated independently, because they have independently selectable clocks, and therefore unused blocks are allocated to other unrelated functions.

An example of this is the 8-bit timer in the upper left corner of the array. This function only requires one datapath in the UDB, and therefore the PLD resources may be allocated to another function. A function such as a Quadrature Decoder may require more PLD logic than one UDB can supply and in this case can

utilize the unused PLD blocks in the 8-bit timer UDB. Programmable resources in the UDB array are generally homogeneous so functions can be mapped to arbitrary boundaries in the array.

**Figure 7-8. Function Mapping Example in a Bank of UDBs**
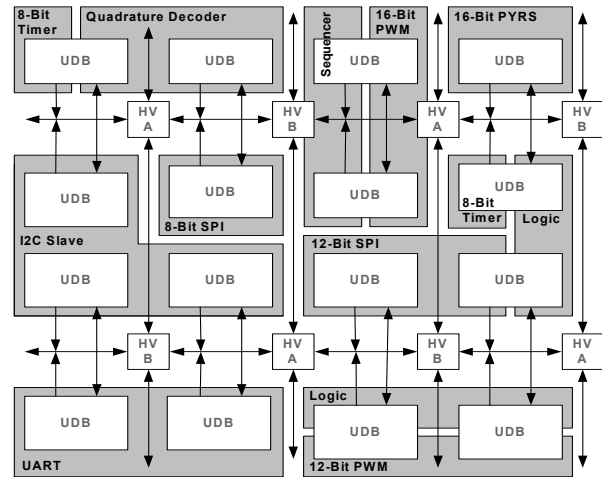


## 7.4 DSI Routing Interface Description

The DSI routing interface is a continuation of the horizontal and vertical routing channels at the top and bottom of the UDB array core. It provides general purpose programmable routing between device peripherals, including UDBs, I/Os, analog peripherals, interrupts, DMA and fixed function peripherals.

Figure 7-9 illustrates the concept of the digital system interconnect, which connects the UDB array routing matrix with other device peripherals. Any digital core or fixed function peripheral that needs programmable routing is connected to this interface.

Signals in this category include:

■ Interrupt requests from all digital peripherals in the system.

■ DMA requests from all digital peripherals in the system.

■ Digital peripheral data signals that need flexible routing to I/Os.

■ Digital peripheral data signals that need connections to UDBs.

■ Connections to the interrupt and DMA controllers.

■ Connection to I/O pins.

■ Connection to analog system digital signals.

**Figure 7-9. Digital System Interconnect**
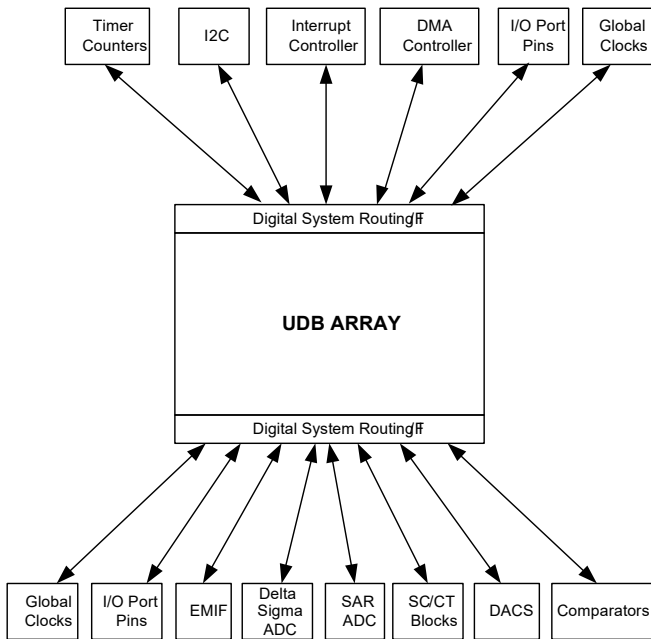


Interrupt and DMA routing is very flexible in the CY8C52LP programmable architecture. In addition to the numerous fixed function peripherals that can generate interrupt requests, any data signal in the UDB array routing can also be used to generate a request. A single peripheral may generate multiple independent interrupt requests simplifying system and firmware design. Figure 7-10 shows the structure of the IDMUX (Interrupt/DMA Multiplexer).

**Figure 7-10. Interrupt and DMA Processing in the IDMUX**
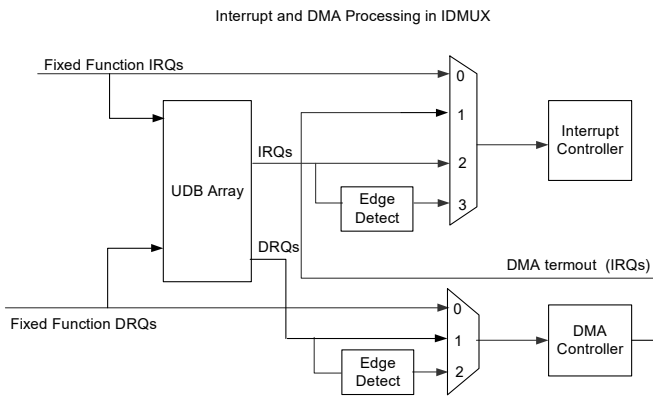


### 7.4.1 I/O Port Routing

There are a total of 20 DSI routes to a typical 8-bit I/O port, 16 for data and four for drive strength control.

When an I/O pin is connected to the routing, there are two primary connections available, an input and an output. In conjunction with drive strength control, this can implement a

bidirectional I/O pin. A data output signal has the option to be single synchronized (pipelined) and a data input signal has the option to be double synchronized. The synchronization clock is the system clock (see Figure 6-1). Normally all inputs from pins are synchronized as this is required if the CPU interacts with the signal or any signal derived from it. Asynchronous inputs have rare uses. An example of this is a feed through of combinational PLD logic from input pins to output pins.
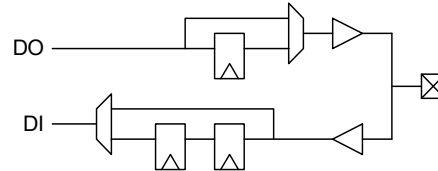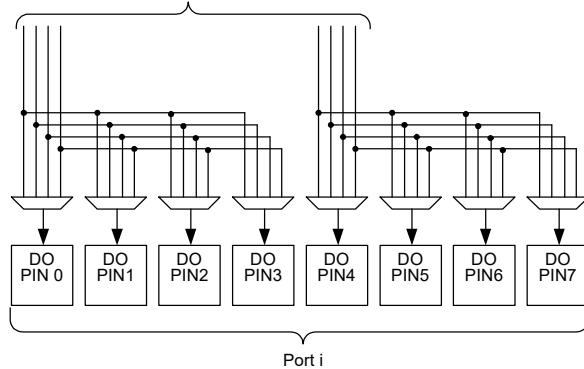
**Figure 7-11. I/O Pin Synchronization Routing**



**Figure 7-12. I/O Pin Output Connectivity**



There are four more DSI connections to a given I/O port to implement dynamic output enable control of pins. This connectivity gives a range of options, from fully ganged 8-bits controlled by one signal, to up to four individually controlled pins. The output enable signal is useful for creating tristate bidirectional pins and buses.

**Figure 7-13. I/O Pin Output Enable Connectivity**
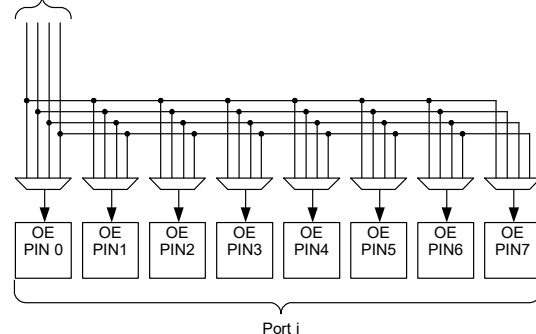
## 7.5 USB

PSoC includes a dedicated Full-Speed (12 Mbps) USB 2.0 transceiver supporting all four USB transfer types: control, interrupt, bulk, and isochronous. PSoC Creator provides full configuration support. USB interfaces to hosts through two dedicated USBIO pins, which are detailed in the "I/O System and Routing" section on page 32.

USB includes the following features:

- Eight unidirectional data endpoints
- One bidirectional control endpoint 0 (EP0)
- Shared 512-byte buffer for the eight data endpoints
- Dedicated 8-byte buffer for EP0
- Three memory modes
  □ Manual Memory Management with No DMA Access
  □ Manual Memory Management with Manual DMA Access
  □ Automatic Memory Management with Automatic DMA Access
- Internal 3.3 V regulator for transceiver
- Internal 48 MHz oscillator that auto locks to USB bus clock, requiring no external crystal for USB (USB equipped parts only)
- Interrupts on bus and each endpoint event, with device wakeup
- USB Reset, Suspend, and Resume operations
- Bus powered and self powered modes

**Figure 7-14. USB**
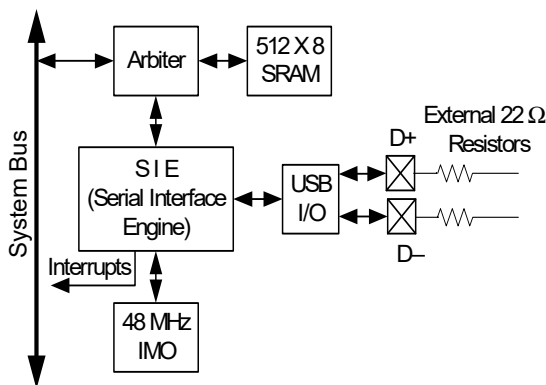


## 7.6 Timers, Counters, and PWMs

The Timer/Counter/PWM peripheral is a 16-bit dedicated peripheral providing three of the most common embedded peripheral features. As almost all embedded systems use some combination of timers, counters, and PWMs. Four of them have been included on this PSoC device family. Additional and more advanced functionality timers, counters, and PWMs can also be instantiated in Universal Digital Blocks (UDBs) as required. PSoC Creator allows you to choose the timer, counter, and PWM features that they require. The tool set utilizes the most optimal resources available.

The Timer/Counter/PWM peripheral can select from multiple clock sources, with input and output signals connected through the DSI routing. DSI routing allows input and output connections to any device pin and any internal digital signal accessible through the DSI. Each of the four instances has a compare output, terminal count output (optional complementary compare output), and programmable interrupt request line. The Timer/Counter/PWMs are configurable as free running, one shot, or Enable input controlled. The peripheral has timer reset and capture inputs, and a kill input for control of the comparator outputs. The peripheral supports full 16-bit capture.

Timer/Counter/PWM features include:

- 16-bit timer/counter/PWM (down count only)
- Selectable clock source
- PWM comparator (configurable for LT, LTE, EQ, GTE, GT)
- Period reload on start, reset, and terminal count
- Interrupt on terminal count, compare true, or capture
- Dynamic counter reads
- Timer capture mode
- Count while enable signal is asserted mode
- Free run mode
- One Shot mode (stop at end of period)
- Complementary PWM outputs with deadband
- PWM output kill

**Figure 7-15. Timer/Counter/PWM**
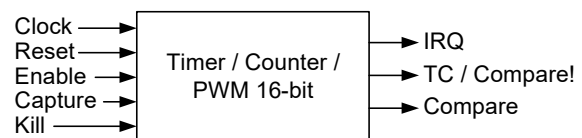
## 7.7 I²C

PSoC includes a single fixed-function I²C peripheral. Additional I²C interfaces can be instantiated using Universal Digital Blocks (UDBs) in PSoC Creator, as required.

The I²C peripheral provides a synchronous two-wire interface designed to interface the PSoC device with a two-wire I²C serial communication bus. It is compatible[13] with I²C Standard-mode, Fast-mode, and Fast-mode Plus devices as defined in the NXP I²C-bus specification and user manual (UM10204). The I²C bus I/O may be implemented with GPIO or SIO in open-drain modes.

To eliminate the need for excessive CPU intervention and overhead, I²C specific support is provided for status detection and generation of framing bits. I²C operates as a slave, a master, or multimaster (Slave and Master)[13]. In slave mode, the unit always listens for a start condition to begin sending or receiving data. Master mode supplies the ability to generate the Start and Stop conditions and initiate transactions. Multimaster mode provides clock synchronization and arbitration to allow multiple masters on the same bus. If Master mode is enabled and Slave mode is not enabled, the block does not generate interrupts on externally generated Start conditions. I²C interfaces through the DSI routing and allows direct connections to any GPIO or SIO pins.

I²C provides hardware address detect of a 7-bit address without CPU intervention. Additionally the device can wake from low power modes on a 7-bit hardware address match. If wakeup

functionality is required, I²C pin connections are limited to one of two specific pairs of SIO pins. See descriptions of SCL and SDA pins in Pin Descriptions on page 11.

I²C features include:

- Slave and master, transmitter, and receiver operation
- Byte processing for low CPU overhead
- Interrupt or polling CPU interface
- Support for bus speeds up to 1 Mbps
- 7 or 10-bit addressing (10-bit addressing requires firmware support)
- SMBus operation (through firmware support - SMBus supported in hardware in UDBs)
- 7-bit hardware address compare
- Wake from low power modes on address match
- Glitch filtering (active and alternate-active modes only)

Data transfers follow the format shown in Figure 7-16. After the START condition (S), a slave address is sent. This address is 7 bits long followed by an eighth bit which is a data direction bit (R/W) - a 'zero' indicates a transmission (WRITE), a 'one' indicates a request for data (READ). A data transfer is always terminated by a STOP condition (P) generated by the master.

### Figure 7-16. I²C Complete Transfer Timing
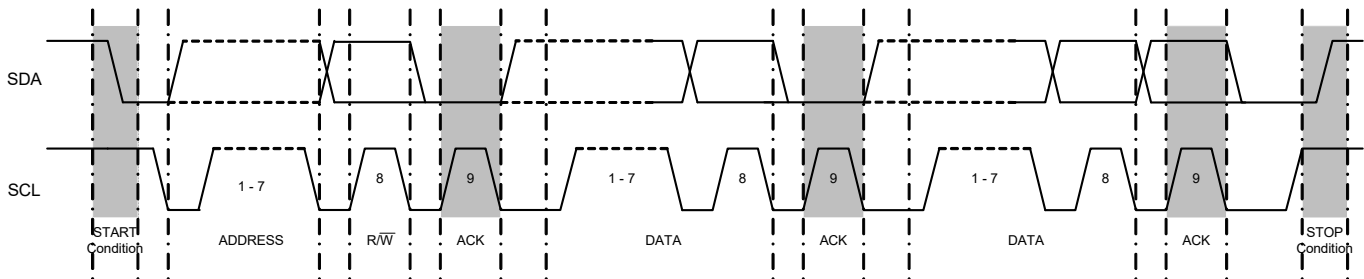


### 7.7.1 External Electrical Connections

As Figure 7-17 shows, the I²C bus requires external pull-up resistors ($R_P$). These resistors are primarily determined by the supply voltage, bus speed, and bus capacitance. For detailed information on how to calculate the optimum pull-up resistor value for your design, we recommend using the UM10204 I2C-bus specification and user manual Rev 6, or newer, available from the NXP website at www.nxp.com.

### Figure 7-17. Connection of Devices to the I²C Bus
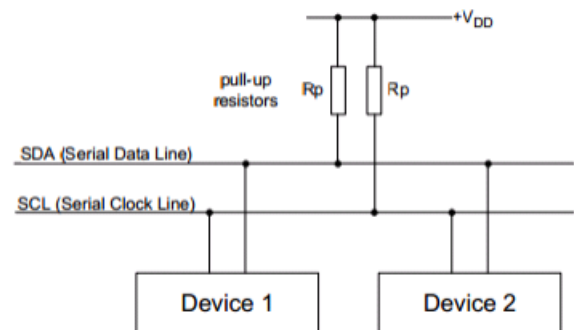


**Notes**
12. The I²C peripheral is non-compliant with the NXP I²C specification in the following areas: analog glitch filter, I/O $V_{OL}/I_{OL}$, I/O hysteresis. The I²C Block has a digital glitch filter (not available in sleep mode). The Fast-mode minimum fall-time specification can be met by setting the I/Os to slow speed mode. See the I/O Electrical Specifications in "Inputs and Outputs" section on page 68 for details.
13. Fixed-block I²C does not support undefined bus conditions, nor does it support Repeated Start in Slave mode. These conditions should be avoided, or the UDB-based I²C component should be used instead.

For most designs, the default values in Table 7-2 will provide excellent performance without any calculations. The default values were chosen to use standard resistor values between the minimum and maximum limits. The values in Table 7-2 work for designs with 1.8 V to 5.0V $V_{DD}$, less than 200-pF bus capacitance ($C_B$), up to 25 μA of total input leakage ($I_{IL}$), up to 0.4 V output voltage level ($V_{OL}$), and a max $V_{IH}$ of 0.7 * $V_{DD}$. Standard Mode and Fast Mode can use either GPIO or SIO PSoC pins. Fast Mode Plus requires use of SIO pins to meet the $V_{OL}$ spec at 20 mA. Calculation of custom pull-up resistor values is required; if your design does not meet the default assumptions, you use series resistors (RS) to limit injected noise, or you need to maximize the resistor value for low power consumption.

**Table 7-2. Recommended default Pull-up Resistor Values**

|  | $R_P$ | Units |
|---|---|---|
| Standard Mode – 100 kbps | 4.7 k, 5% | Ω |
| Fast Mode – 400 kbps | 1.74 k, 1% | Ω |
| Fast Mode Plus – 1 Mbps | 620, 5% | Ω |

Calculation of the ideal pull-up resistor value involves finding a value between the limits set by three equations detailed in the NXP I²C specification. These equations are:

**Equation 1:**

$$R_{PMIN} = (V_{DD}(max) - V_{OL}(max))/(I_{OL}(min))$$

**Equation 2:**

$$R_{PMAX} = T_R(max)/0.8473 \times C_B(max)$$

**Equation 3:**

$$R_{PMAX} = V_{DD}(min) - V_{IH}(min) + V_{NH}(min)/I_{IH}(max)$$

Equation parameters:

$V_{DD}$ = Nominal supply voltage for I²C bus

$V_{OL}$ = Maximum output low voltage of bus devices.

$I_{OL}$ = Low-level output current from I²C specification

$T_R$ = Rise Time of bus from I²C specification

$C_B$ = Capacitance of each bus line including pins and PCB traces

$V_{IH}$ = Minimum high-level input voltage of all bus devices

$V_{NH}$ = Minimum high-level input noise margin from I²C specification

$I_{IH}$ = Total input leakage current of all devices on the bus

The supply voltage ($V_{DD}$) limits the minimum pull-up resistor value due to bus devices maximum low output voltage ($V_{OL}$) specifications. Lower pull-up resistance increases current though the pins and can, therefore, exceed the spec conditions of $V_{OH}$. Equation 1 is derived using Ohm's law to determine the minimum resistance that will still meet the $V_{OL}$ specification at 3 mA for standard and fast modes, and 20 mA for fast mode plus at the given $V_{DD}$.

Equation 2 determines the maximum pull-up resistance due to bus capacitance. Total bus capacitance is comprised of all pin, wire, and trace capacitance on the bus. The higher the bus capacitance, the lower the pull-up resistance required to meet the specified bus speeds rise time due to RC delays. Choosing a pull-up resistance higher than allowed can result in failing timing requirements resulting in communication errors. Most designs with five or less I²C devices and up to 20 centimeters of bus trace length have less than 100 pF of bus capacitance.

A secondary effect that limits the maximum pull-up resistor value is total bus leakage calculated in Equation 3. The primary source of leakage is I/O pins connected to the bus. If leakage is too high, the pull-ups will have difficulty maintaining an acceptable $V_{IH}$ level causing communication errors. Most designs with five or less I²C devices on the bus have less than 10 μA of total leakage current.

# 8. Analog Subsystem

The analog programmable system creates application specific combinations of both standard and advanced analog signal processing blocks. These blocks are then interconnected to each other and also to any pin on the device, providing a high level of design flexibility and IP security. The features of the analog subsystem are outlined here to provide an overview of capabilities and architecture.

■ Flexible, configurable analog routing architecture provided by analog globals, analog mux bus, and analog local buses

■ Successive approximation (SAR) ADC

■ One 8-bit DAC that provides either voltage or current output

■ Two comparators with optional connection to configurable LUT outputs

■ CapSense subsystem to enable capacitive touch sensing

■ Precision reference for generating an accurate analog voltage for internal analog blocks

**Figure 8-1. Analog Subsystem Block Diagram**
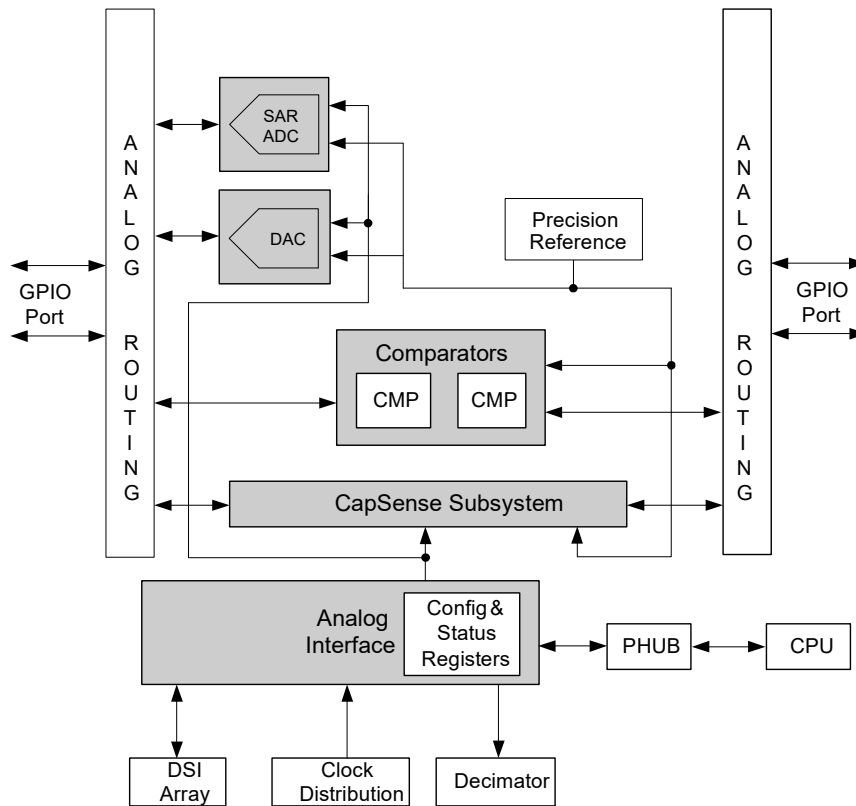
The PSoC Creator software program provides a user-friendly interface to configure the analog connections between the GPIO and various analog resources and also connections from one analog resource to another. PSoC Creator also provides component libraries that allow you to configure the various analog blocks to perform application specific functions. The tool also generates API interface libraries that allow you to write firmware that allows the communication between the analog peripheral and CPU/Memory.

## 8.1 Analog Routing

The PSoC 5LP family of devices has a flexible analog routing architecture that provides the capability to connect GPIOs and different analog blocks, and also route signals between different analog blocks. One of the strong points of this flexible routing architecture is that it allows dynamic routing of input and output connections to the different analog blocks.

For information on how to make pin selections for optimal analog routing, refer to the application note, AN58304 - PSoC® 3 and PSoC® 5 - Pin Selection for Analog Designs.

### 8.1.1 Features

- Flexible, configurable analog routing architecture

- 16 Analog globals (AG) and two analog mux buses (AMUXBUS) to connect GPIOs and the analog blocks

- Each GPIO is connected to one analog global and one analog mux bus

- 8 Analog local buses (abus) to route signals between the different analog blocks

- Multiplexers and switches for input and output selection of the analog blocks

### 8.1.2 Functional Description

Analog globals (AGs) and analog mux buses (AMUXBUS) provide analog connectivity between GPIOs and the various analog blocks. There are 16 AGs in the PSoC 5LP family. The analog routing architecture is divided into four quadrants as shown in Figure 8-2. Each quadrant has four analog globals (AGL[0..3], AGL[4..7], AGR[0..3], AGR[4..7]). Each GPIO is connected to the corresponding AG through an analog switch. The analog mux bus is a shared routing resource that connects to every GPIO through an analog switch. There are two AMUXBUS routes in PSoC 5LP, one in the left half (AMUXBUSL) and one in the right half (AMUXBUSR), as shown in Figure 8-2.

Analog local buses (abus) are routing resources located within the analog subsystem and are used to route signals between different analog blocks. There are eight abus routes in PSoC 5LP, four in the left half (abusl [0:3]) and four in the right half (abusr [0:3]) as shown in Figure 8-2. Using the abus saves the analog globals and analog mux buses from being used for interconnecting the analog blocks.

Multiplexers and switches exist on the various buses to direct signals into and out of the analog blocks. A multiplexer can have only one connection on at a time, whereas a switch can have multiple connections on simultaneously. In Figure 8-2, multiplexers are indicated by grayed ovals and switches are indicated by transparent ovals.

**Figure 8-2. CY8C52LP Analog Interconnect**
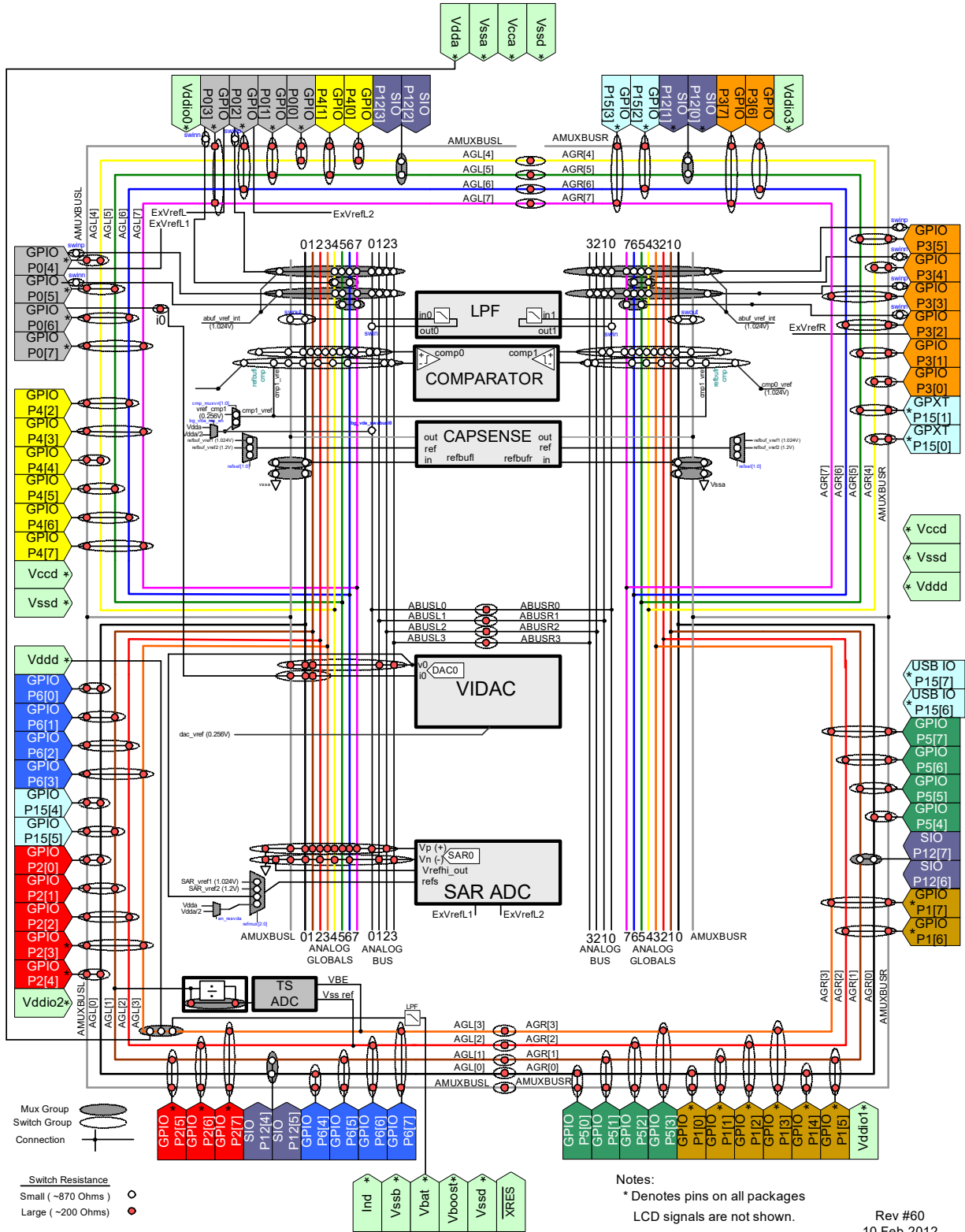


To preserve detail of this image, this image is best viewed with a PDF display program or printed on 11" × 17" paper.

## 8.2 Successive Approximation ADC

The PSoC 5LP family of devices has a SAR ADC. This ADC is 12-bit at up to 1 Msps, with single-ended or differential inputs, making it useful for a wide variety of sampling and control applications.

### 8.2.1 Functional Description

In a SAR ADC an analog input signal is sampled and compared with the output of a DAC. A binary search algorithm is applied to the DAC and used to determine the output bits in succession from MSB to LSB. A block diagram of the SAR ADC is shown in Figure 8-3.

**Figure 8-3. SAR ADC Block Diagram**
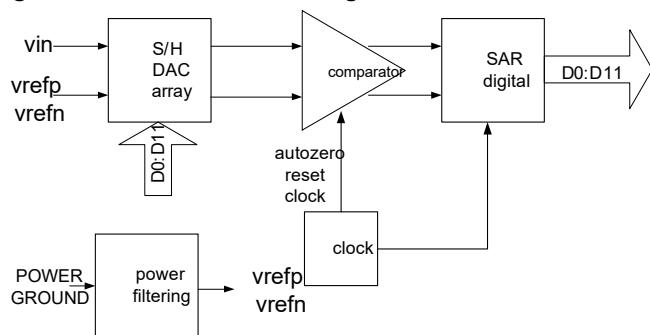


The input is connected to the analog globals and muxes. The frequency of the clock is 18 times the sample rate; the clock rate ranges from 1 to 18 MHz.

### 8.2.2 Conversion Signals

Writing a start bit or assertion of a start of frame (SOF) signal is used to start a conversion. SOF can be used in applications where the sampling period is longer than the conversion time, or when the ADC needs to be synchronized to other hardware. This signal is optional and does not need to be connected if the SAR ADC is running in a continuous mode. A digital clock or UDB output can be used to drive this input. When the SAR is first

powered up or awakened from any of the sleeping modes, there is a power up wait time of 10 µs before it is ready to start the first conversion.

When the conversion is complete, a status bit is set and the output signal end of frame (EOF) asserts and remains asserted until the value is read by either the DMA controller or the CPU. The EOF signal may be used to trigger an interrupt or a DMA request.

### 8.2.3 Operational Modes

A ONE_SHOT control bit is used to set the SAR ADC conversion mode to either continuous or one conversion per SOF signal. DMA transfer of continuous samples, without CPU intervention, is supported.
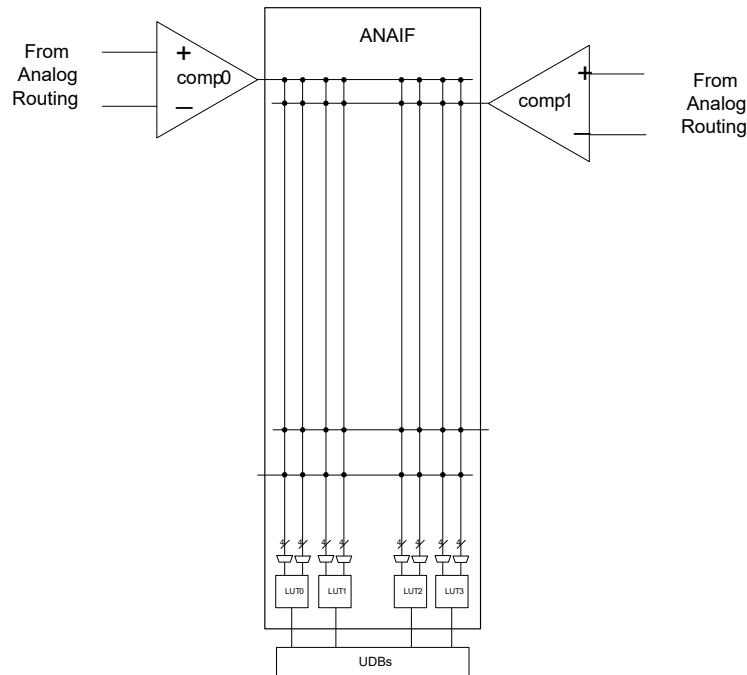
## 8.3 Comparators

The CY8C52LP family of devices contains two comparators in a device. Comparators have these features:

■ Input offset factory trimmed to less than 5 mV

■ Rail-to-rail common mode input range ($V_{SSA}$ to $V_{DDA}$)

■ Speed and power can be traded off by using one of three modes: fast, slow, or ultra low power

■ Comparator outputs can be routed to look up tables to perform simple logic functions and then can also be routed to digital blocks

■ The positive input of the comparators may be optionally passed through a low pass filter. Two filters are provided

■ Comparator inputs can be connected to GPIO or DAC output

### 8.3.1 Input and Output Interface

The positive and negative inputs to the comparators come from the analog global buses, the analog mux line, the analog local bus and precision reference through multiplexers. The output from each comparator could be routed to any of the two input LUTs. The output of that LUT is routed to the UDB digital system interface.

**Figure 8-4. Analog Comparator**



### 8.3.2 LUT

The CY8C52LP family of devices contains two LUTs. The LUT is a two input, one output lookup table that is driven by one or two of the comparators in the chip. The output of any LUT is routed to the digital system interface of the UDB array. From the digital system interface of the UDB array, these signals can be connected to UDBs, DMA controller, I/O, or the interrupt controller. The LUT control word written to a register sets the logic function on the output. The available LUT functions and the associated control word is shown in Table 8-1.

**Table 8-1. LUT Function vs. Program Word and Inputs**

| Control Word | Output (A and B are LUT inputs) |
|---|---|
| 0000b | **FALSE** ('0') |
| 0001b | **A** AND **B** |
| 0010b | **A** AND (NOT **B**) |
| 0011b | A |
| 0100b | (NOT **A**) AND **B** |
| 0101b | B |
| 0110b | **A** XOR **B** |
| 0111b | **A** OR **B** |
| 1000b | **A** NOR **B** |
| 1001b | **A** XNOR **B** |
| 1010b | NOT **B** |
| 1011b | **A** OR (NOT **B**) |
| 1100b | NOT **A** |
| 1101b | (NOT **A**) OR **B** |
| 1110b | **A** NAND **B** |
| 1111b | **TRUE** ('1') |

### 8.4 LCD Direct Drive

The PSoC LCD driver system is a highly configurable peripheral designed to allow PSoC to directly drive a broad range of LCD glass. All voltages are generated on chip, eliminating the need for external components. With a high multiplex ratio of up to 1/16, the CY8C52LP family LCD driver system can drive a maximum of 736 segments. The PSoC LCD driver module was also designed with the conservative power budget of portable devices in mind, enabling different LCD drive modes and power down modes to conserve power.

PSoC Creator provides an LCD segment drive component. The component wizard provides easy and flexible configuration of LCD resources. You can specify pins for segments and commons along with other options. The software configures the device to meet the required specifications. This is possible because of the programmability inherent to PSoC devices.

Key features of the PSoC LCD segment system are:

■ LCD panel direct driving
■ Type A (standard) and Type B (low power) waveform support
■ Wide operating voltage range support (2 V to 5 V) for LCD panels
■ Static, 1/2, 1/3, 1/4, 1/5 bias voltage levels
■ Internal bias voltage generation through internal resistor ladder
■ Up to 62 total common and segment outputs
■ Up to 1/16 multiplex for a maximum of 16 backplane/common outputs
■ Up to 62 front plane/segment outputs for direct drive
■ Drives up to 736 total segments (16 backplane × 46 front plane)
■ Up to 64 levels of software controlled contrast
■ Ability to move display data from memory buffer to LCD driver through DMA (without CPU intervention)

- Adjustable LCD refresh rate from 10 Hz to 150 Hz
- Ability to invert LCD display for negative image
- Three LCD driver drive modes, allowing power optimization

**Figure 8-5. LCD System**
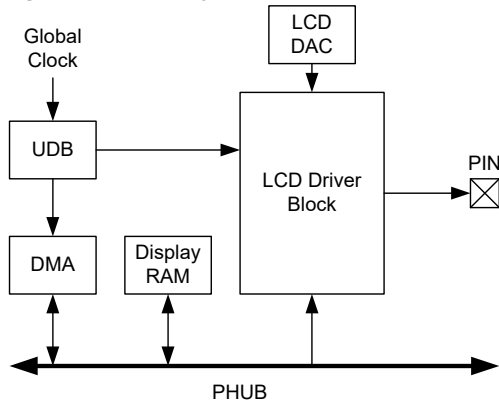


### 8.4.1 LCD Segment Pin Driver

Each GPIO pin contains an LCD driver circuit. The LCD driver buffers the appropriate output of the LCD DAC to directly drive the glass of the LCD. A register setting determines whether the pin is a common or segment. The pin's LCD driver then selects one of the six bias voltages to drive the I/O pin, as appropriate for the display data.

### 8.4.2 Display Data Flow

The LCD segment driver system reads display data and generates the proper output voltages to the LCD glass to produce the desired image. Display data resides in a memory buffer in the system SRAM. Each time you need to change the common and segment driver voltages, the next set of pixel data moves from the memory buffer into the Port Data Registers via DMA.

### 8.4.3 UDB and LCD Segment Control

A UDB is configured to generate the global LCD control signals and clocking. This set of signals is routed to each LCD pin driver through a set of dedicated LCD global routing channels. In addition to generating the global LCD control signals, the UDB also produces a DMA request to initiate the transfer of the next frame of LCD data.

### 8.4.4 LCD DAC

The LCD DAC generates the contrast control and bias voltage for the LCD system. The LCD DAC produces up to five LCD drive voltages plus ground, based on the selected bias ratio. The bias voltages are driven out to GPIO pins on a dedicated LCD bias bus, as required.

## 8.5 CapSense

The CapSense system provides a versatile and efficient means for measuring capacitance in applications such as touch sense buttons, sliders, proximity detection, etc. The CapSense system uses a configuration of system resources, including a few hardware functions primarily targeted for CapSense. Specific resource usage is detailed in the CapSense component in PSoC Creator.

A capacitive sensing method using a delta-sigma modulator (CSD) is used. It provides capacitance sensing using a switched capacitor technique with a delta-sigma modulator to convert the sensing current to a digital code.

## 8.6 Temp Sensor

Die temperature is used to establish programming parameters for writing flash. Die temperature is measured using a dedicated sensor based on a forward biased transistor. The temperature sensor has its own auxiliary ADC.

## 8.7 DAC

The CY8C32 parts contain a DAC. The DAC is 8-bit and can be configured for either voltage or current output. The DAC supports CapSense, power supply regulation, and waveform generation. The DAC has the following features:

- Adjustable voltage or current output in 255 steps
- Programmable step size (range selection)
- Eight bits of calibration to correct ± 25% of gain error
- Source and sink option for current output
- 8-Msps conversion rate for current output
- 1-Msps conversion rate for voltage output
- Monotonic in nature
- Data and strobe inputs can be provided by the CPU or DMA, or routed directly from the DSI
- Dedicated low-resistance output pin for high-current mode

**Figure 8-6. DAC Block Diagram**
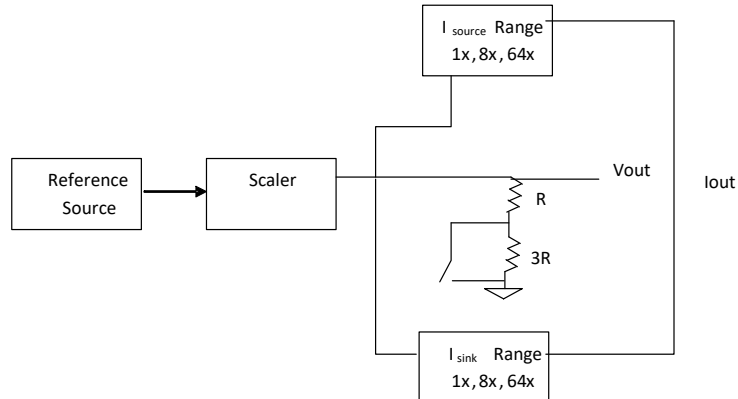


### 8.7.1  Current DAC

The IDAC can be configured for the ranges 0 to 31.875 µA, 0 to 255 µA, and 0 to 2.04 mA. The IDAC can be configured to source or sink current.

### 8.7.2  Voltage DAC

For the VDAC, the current DAC output is routed through resistors. The two ranges available for the VDAC are 0 to 1.02 V and 0 to 4.08 V. In voltage mode any load connected to the output of a DAC should be purely capacitive (the output of the VDAC is not buffered).

# 9.  Programming, Debug Interfaces, Resources

The Cortex-M3 has internal debugging components, tightly integrated with the CPU, providing the following features:

■ JTAG or SWD access

■ FPB block for implementing breakpoints and code patches

■ DWT block for implementing watchpoints, trigger resources, and system profiling

■ ETM for instruction trace

■ ITM for support of printf-style debugging

PSoC devices include extensive support for programming, testing, debugging, and tracing both hardware and firmware. Four interfaces are available: JTAG, SWD, SWV, and TRACEPORT. JTAG and SWD support all programming and debug features of the device. JTAG also supports standard JTAG scan chains for board level test and chaining multiple JTAG devices to a single JTAG connection. The SWV and TRACEPORT provide trace output from the DWT, ETM, and ITM. TRACEPORT is faster but uses more pins. SWV is slower but uses only one pin.

For more information on PSoC 5 programming, refer to the PSoC 5 Device Programming Specification.

Cortex-M3 debug and trace functionality enables full device debugging in the final system using the standard production device. It does not require special interfaces, debugging pods, simulators, or emulators. Only the standard programming connections are required to fully support debug.

The PSoC Creator IDE software provides fully integrated programming and debug support for PSoC devices. The low cost MiniProg3 programmer and debugger is designed to provide full programming and debug support of PSoC devices in conjunction with the PSoC Creator IDE. PSoC JTAG, SWD, and SWV interfaces are fully compatible with industry standard third party tools.

All Cortex-M3 debug and trace modules are disabled by default and can only be enabled in firmware. If not enabled, the only way to reenable them is to erase the entire device, clear flash protection, and reprogram the device with new firmware that enables them. Disabling debug and trace features, robust flash protection, and hiding custom analog and digital functionality inside the PSoC device provide a level of security not possible with multichip application solutions. Additionally, all device interfaces can be permanently disabled (Device Security) for applications concerned about phishing attacks due to a maliciously reprogrammed device. Permanently disabling interfaces is not recommended in most applications because the designer then cannot access the device later. Because all programming, debug, and test interfaces are disabled when Device Security is enabled, PSoCs with Device Security enabled may not be returned for failure analysis.

## 9.1  JTAG Interface

The IEEE 1149.1 compliant JTAG interface exists on four or five pins (the nTRST pin is optional). The JTAG clock frequency can be up to 12 MHz, or 1/3 of the CPU clock frequency for 8 and 16-bit transfers, or 1/5 of the CPU clock frequency for 32-bit

transfers, whichever is least. By default, the JTAG pins are enabled on new devices but the JTAG interface can be disabled, allowing these pins to be used as General Purpose I/O (GPIO) instead. The JTAG interface is used for programming the flash memory, debugging, I/O scan chains, and JTAG device chaining.

**Figure 9-1. JTAG Interface Connections between PSoC 5LP and Programmer**
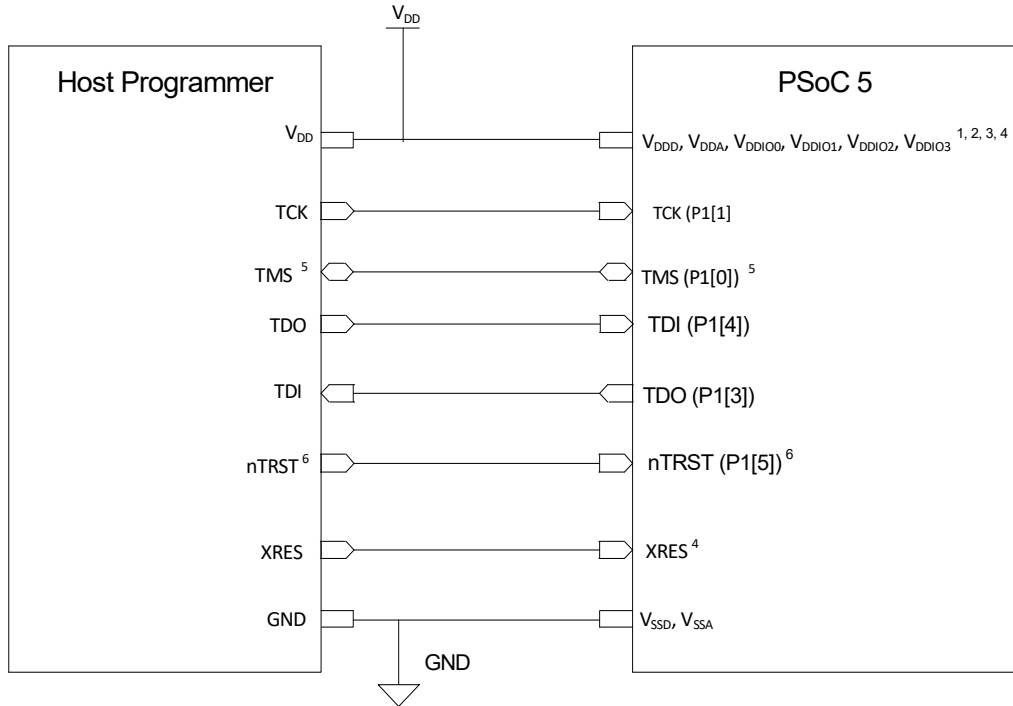


[1] The voltage levels of Host Programmer and the PSoC 5 voltage domains involved in Programming should be same. The Port 1 JTAG pins and XRES pin are powered by $V_{DDIO1}$. So, $V_{DDIO1}$ of PSoC 5 should be at same voltage level as host $V_{DD}$. Rest of PSoC 5 voltage domains ($V_{DDD}$, $V_{DDA}$, $V_{DDIO0}$, $V_{DDIO2}$, $V_{DDIO3}$) need not be at the same voltage level as host Programmer.

[2] Vdda must be greater than or equal to all other power supplies (Vddd, Vddios) in PSoC 5.

[3] For Power cycle mode Programming, XRES pin is not required. But the Host programmer must have the capability to toggle power (Vddd, Vdda, All Vddios) to PSoC 5. This may typically require external interface circuitry to toggle power which will depend on the programming setup. The power supplies can be brought up in any sequence, however, once stable, VDDA must be greater than or equal to all other supplies.

[4] For JTAG Programming, Device reset can also be done without connecting to the XRES pin or Power cycle mode by using the TMS, TCK, TDI, TDO pins of PSoC 5, and writing to a specific register. But this requires that the DPS setting in NVL is not equal to Debug Ports Disabled.

[5] By default, PSoC 5 is configured for 4-wire JTAG mode unless user changes the DPS setting. So the TMS pin is unidirectional. But if the DPS setting is changed to non-JTAG mode, the TMS pin in JTAG is bi-directional as the SWD Protocol has to be used for acquiring the PSoC 5 device initially. After switching from SWD to JTAG mode, the TMS pin will be uni-directional. In such a case, unidirectional buffer should not be used on TMS line.

[6] nTRST JTAG pin (P1[5]) cannot be used to reset the JTAG TAP controller during first time programming of PSoC 5 as the default setting is 4-wire JTAG (nTRST disabled). Use the TMS, TCK pins to do a reset of JTAG TAP controller.

## 9.2 SWD Interface

The SWD interface is the preferred alternative to the JTAG interface. It requires only two pins instead of the four or five needed by JTAG. SWD provides all of the programming and debugging features of JTAG at the same speed. SWD does not provide access to scan chains or device chaining. The SWD clock frequency can be up to 1/3 of the CPU clock frequency.

SWD uses two pins, either two of the JTAG pins (TMS and TCK) or the USBIO D+ and D- pins. The USBIO pins are useful for in system programming of USB solutions that would otherwise require a separate programming connector. One pin is used for the data clock and the other is used for data input and output.

SWD can be enabled on only one of the pin pairs at a time. This only happens if, within 8 µs (key window) after reset, that pin pair (JTAG or USB) receives a predetermined acquire sequence of 1s and 0s. If the NVL latches are set for SWD (see *Section 5.5*), this sequence need not be applied to the JTAG pin pair. The acquire sequence must always be applied to the USB pin pair.

SWD is used for debugging or for programming the flash memory.

The SWD interface can be enabled from the JTAG interface or disabled, allowing its pins to be used as GPIO. Unlike JTAG, the SWD interface can always be reacquired on any device during the key window. It can then be used to reenable the JTAG interface, if desired. When using SWD or JTAG pins as standard GPIO, make sure that the GPIO functionality and PCB circuits do not interfere with SWD or JTAG use.

**Figure 9-2. SWD Interface Connections between PSoC 5LP and Programmer**
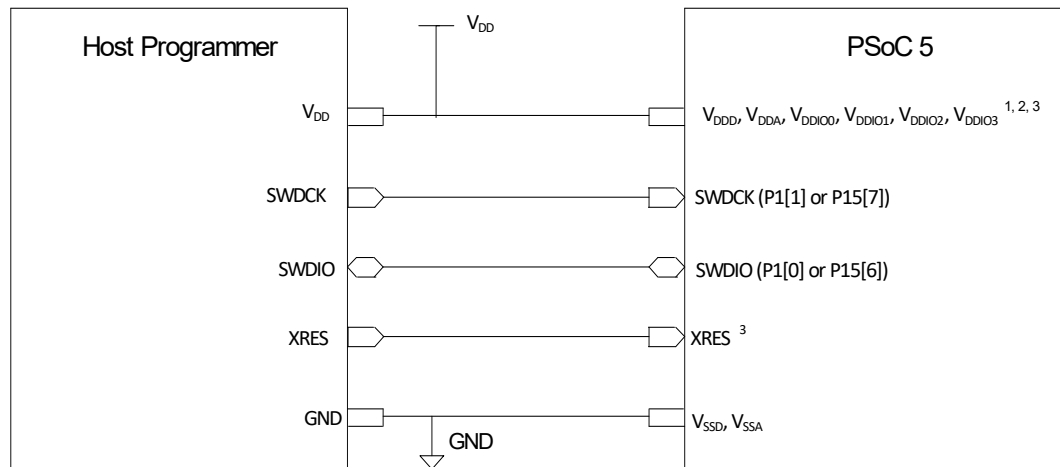


[1] The voltage levels of the Host Programmer and the PSoC 5 voltage domains involved in programming should be the same. The XRES pin is powered by $V_{DDIO1}$. The USB SWD pins are powered by $V_{DDD}$. So for Programming using the USB SWD pins with XRES pin, the $V_{DDD}$, $V_{DDIO1}$ of PSoC 5 should be at the same voltage level as Host $V_{DD}$. Rest of PSoC 5 voltage domains ($V_{DDA}$, $V_{DDIO0}$, $V_{DDIO2}$, $V_{DDIO3}$) need not be at the same voltage level as host Programmer. The Port 1 SWD pins are powered by $V_{DDIO1}$. So $V_{DDIO1}$ of PSoC 5 should be at same voltage level as host $V_{DD}$ for Port 1 SWD programming. Rest of PSoC 5 voltage domains ($V_{DDD}$, $V_{DDA}$, $V_{DDIO0}$, $V_{DDIO2}$, $V_{DDIO3}$) need not be at the same voltage level as host Programmer.

[2] Vdda must be greater than or equal to all other power supplies (Vddd, Vddio s) in PSoC 5.

[3] For Power cycle mode Programming, XRES pin is not required. But the Host programmer must have the capability to toggle power (Vddd, Vdda, All Vddio s) to PSoC 5. This may typically require external interface circuitry to toggle power which will depend on the programming setup. The power supplies can be brought up in any sequence, however, once stable, VDDA must be greater than or equal to all other supplies.

## 9.3 Debug Features

The CY8C52LP supports the following debug features:

- Halt and single-step the CPU

- View and change CPU and peripheral registers, and RAM addresses

- Six program address breakpoints and two literal access breakpoints

- Data watchpoint events to CPU

- Patch and remap instruction from flash to SRAM

- Debugging at the full speed of the CPU

- Compatible with PSoC Creator and MiniProg3 programmer and debugger

- Standard JTAG programming and debugging interfaces make CY8C52LP compatible with other popular third-party tools (for example, Arm / Keil)

## 9.4 Trace Features

The following trace features are supported:

- Instruction trace

- Data watchpoint on access to data address, address range, or data value

- Trace trigger on data watchpoint

- Debug exception trigger

- Code profiling

- Counters for measuring clock cycles, folded instructions, load/store operations, sleep cycles, cycles per instruction, interrupt overhead

- Interrupt events trace

- Software event monitoring, "printf-style" debugging

## 9.5 SWV and TRACEPORT Interfaces

The SWV and TRACEPORT interfaces provide trace data to a debug host via the Cypress MiniProg3 or an external trace port analyzer. The 5 pin TRACEPORT is used for rapid transmission of large trace streams. The single pin SWV mode is used to minimize the number of trace pins. SWV is shared with a JTAG pin. If debugging and tracing are done at the same time then SWD may be used with either SWV or TRACEPORT, or JTAG may be used with TRACEPORT, as shown in Table 9-1.

**Table 9-1. Debug Configurations**

| Debug and Trace Configuration | GPIO Pins Used |
|---|---|
| All debug and trace disabled | 0 |
| JTAG | 4 or 5 |
| SWD | 2 |
| SWV | 1 |
| TRACEPORT | 5 |
| JTAG + TRACEPORT | 9 or 10 |
| SWD + SWV | 3 |
| SWD + TRACEPORT | 7 |

## 9.6 Programming Features

The JTAG and SWD interfaces provide full programming support. The entire device can be erased, programmed, and verified. Designers can increase flash protection levels to protect firmware IP. Flash protection can only be reset after a full device erase. Individual flash blocks can be erased, programmed, and verified, if block security settings permit.

## 9.7 Device Security

PSoC 5LP offers an advanced security feature called device security, which permanently disables all test, programming, and debug ports, protecting your application from external access. The device security is activated by programming a 32-bit key (0x50536F43) to a Write Once Latch (WOL).

The Write Once Latch is a type of nonvolatile latch (NVL). The cell itself is an NVL with additional logic wrapped around it. Each WOL device contains four bytes (32 bits) of data. The wrapper outputs a '1' if a super-majority (28 of 32) of its bits match a pre-determined pattern (0x50536F43); it outputs a '0' if this majority is not reached. When the output is 1, the Write Once NV latch locks the part out of Debug and Test modes; it also permanently gates off the ability to erase or alter the contents of the latch. Matching all bits is intentionally not required, so that single (or few) bit failures do not deassert the WOL output. The state of the NVL bits after wafer processing is truly random with no tendency toward 1 or 0.

The WOL only locks the part after the correct 32-bit key (0x50536F43) is loaded into the NVL's volatile memory, programmed into the NVL's nonvolatile cells, and the part is reset. The output of the WOL is only sampled on reset and used to disable the access. This precaution prevents anyone from reading, erasing, or altering the contents of the internal memory.

You can write the key into the WOL to lock out external access only if no flash protection is set (see "Flash Security" section on page 19). However, after setting the values in the WOL, you still have access to the part until it is reset. Therefore, you can write the key into the WOL, program the flash protection data, and then reset the part to lock it.

If the device is protected with a WOL setting, Cypress cannot perform failure analysis and, therefore, cannot accept RMAs from customers. The WOL can be read out via Serial Wire Debug (SWD) port to electrically identify protected parts. You can write the key in WOL to lock out external access only if no flash protection is set. For more information on how to take full advantage of the security features in PSoC see the PSoC 5 TRM.

**Disclaimer**

Note the following details of the flash code protection features on Cypress devices.

Cypress products meet the specifications contained in their particular Cypress datasheets. Cypress believes that its family of 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.

## 9.8 CSP Package Bootloader

A factory-installed bootloader program is included in all devices with CSP packages. The bootloader is compatible with PSoC Creator 3.0 bootloadable project files, and has the following features:

- I2C-based

- SCLK and SDAT available at P1[6] and P1[7], respectively

- External pull-up resistors required

- I2C slave, address 4, data rate = 100 kbps

- Single application

- Wait 2 seconds for bootload command

- Other bootloader options are as set by the PSoC Creator 3.0 Bootloader Component default

- Occupies the bottom 9 Kbytes of flash

For more information on this bootloader, see the following Cypress application notes:

- AN73854, PSoC 3 and PSoC 5 LP Introduction to Bootloaders

- AN60317, PSoC 3 and PSoC 5 LP I2C Bootloader

Note that a PSoC Creator bootloadable project must be associated with .hex and .elf files for a bootloader project that is configured for the target device. Bootloader .hex and .elf files can be found at www.cypress.com/go/PSoC5LPdatasheet.

The factory-installed bootloader can be overwritten using JTAG or SWD programming.

## 10. Development Support

The CY8C52LP family has a rich set of documentation, development tools, and online resources to assist you during your development process. Visit psoc.cypress.com/getting-started to find out more.

### 10.1 Documentation

A suite of documentation, to ensure that you can find answers to your questions quickly, supports the CY8C52LP family. This section contains a list of some of the key documents.

**Software User Guide**: A step-by-step guide for using PSoC Creator. The software user guide shows you how the PSoC

Creator build process works in detail, how to use source control with PSoC Creator, and much more.

**Component Datasheets**: The flexibility of PSoC allows the creation of new peripherals (components) long after the device has gone into production. Component datasheets provide all of the information needed to select and use a particular component, including a functional description, API documentation, example code, and AC/DC specifications.

**Application Notes**: PSoC application notes discuss a particular application of PSoC in depth; examples include brushless DC motor control and on-chip filtering. Application notes often include example projects in addition to the application note document.

**Technical Reference Manual**: PSoC Creator makes designing with PSoC as easy as dragging a peripheral onto a schematic, but, when low level details of the PSoC device are required, use the technical reference manual (TRM) as your guide.

**Note** Visit www.arm.com for detailed documentation about the Cortex-M3 CPU.

### 10.2 Online

In addition to print documentation, the Cypress PSoC forums connect you with fellow PSoC users and experts in PSoC from around the world, 24 hours a day, 7 days a week.

### 10.3 Tools

With industry standard cores, programming, and debugging interfaces, the CY8C52LP family is part of a development tool ecosystem. Visit us at www.cypress.com/psoccreator for the latest information on the revolutionary, easy to use PSoC Creator IDE, supported third party compilers, programmers, debuggers, and development kits.

# 11. Electrical Specifications

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted. The unique flexibility of the PSoC UDBs and analog blocks enable many functions to be implemented in PSoC Creator components, see the component datasheets for full AC/DC specifications of individual functions. See the "Example Peripherals" section on page 39 for further explanation of PSoC Creator components.

## 11.1 Absolute Maximum Ratings

**Table 11-1. Absolute Maximum Ratings DC Specifications[14]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{DDA}$ | Analog supply voltage relative to $V_{SSA}$ | | –0.5 | – | 6 | V |
| $V_{DDD}$ | Digital supply voltage relative to $V_{SSD}$ | | –0.5 | – | 6 | V |
| $V_{DDIO}$ | I/O supply voltage relative to $V_{SSD}$ | | –0.5 | – | 6 | V |
| $V_{CCA}$ | Direct analog core voltage input | | –0.5 | – | 1.95 | V |
| $V_{CCD}$ | Direct digital core voltage input | | –0.5 | – | 1.95 | V |
| $V_{SSA}$ | Analog ground voltage | | $V_{SSD} - 0.5$ | – | $V_{SSD} + 0.5$ | V |
| $V_{GPIO}$[15] | DC input voltage on GPIO | Includes signals sourced by $V_{DDA}$ and routed internal to the pin. | $V_{SSD} - 0.5$ | – | $V_{DDIO} + 0.5$ | V |
| $V_{SIO}$ | DC input voltage on SIO | Output disabled | $V_{SSD} - 0.5$ | – | 7 | V |
| | | Output enabled | $V_{SSD} - 0.5$ | – | 6 | V |
| $V_{IND}$ | Voltage at boost converter input | | 0.5 | – | 5.5 | V |
| $V_{BAT}$ | Boost converter supply | | $V_{SSD} - 0.5$ | – | 5.5 | V |
| $I_{VDDIO}$ | Current per $V_{DDIO}$ supply pin | | – | – | 100 | mA |
| $I_{GPIO}$ | GPIO current | | –30 | – | 41 | mA |
| $I_{SIO}$ | SIO current | | –49 | – | 28 | mA |
| $I_{USBIO}$ | USBIO current | | –56 | – | 59 | mA |
| LU | Latch up current[16] | | –140 | – | 140 | mA |
| $ESD_{HBM}$ | Electrostatic discharge voltage | Human body model | 2000 | – | – | V |
| $ESD_{CDM}$ | Electrostatic discharge voltage | Charge device model | 500 | – | – | V |

**Notes**
14. Usage above the absolute maximum conditions listed in Table 11-1 may cause permanent damage to the device. Exposure to Absolute Maximum conditions for extended periods of time may affect device reliability. The Maximum Storage Temperature is 150 °C in compliance with JEDEC Standard JESD22-A103, High Temperature Storage Life. When used below Absolute Maximum conditions but above normal operating conditions, the device may not operate to specification.
15. The $V_{DDIO}$ supply voltage must be greater than the maximum voltage on the associated GPIO pins. Maximum voltage on GPIO pin ≤ $V_{DDIO}$ ≤ $V_{DDA}$
16. Meets or exceeds JEDEC Spec EIA/JESD78 IC Latch-up Test.

## 11.2 Device Level Specifications

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted. Unless otherwise specified, all charts and graphs show typical values.

### 11.2.1 Device Level Specifications

**Table 11-2. DC Specifications**

| Parameter | Description | Conditions | | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|---|
| $V_{DDA}$ | Analog supply voltage and input to analog core regulator | Analog core regulator enabled | | 1.8 | – | 5.5 | V |
| $V_{DDA}$ | Analog supply voltage, analog regulator bypassed | Analog core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| $V_{DDD}$ | Digital supply voltage relative to $V_{SSD}$ | Digital core regulator enabled | | 1.8 | – | $V_{DDA}$[17] | V |
| | | | | – | – | $V_{DDA}$ + 0.1[19] | |
| $V_{DDD}$ | Digital supply voltage, digital regulator bypassed | Digital core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| $V_{DDIO}$[18] | I/O supply voltage relative to $V_{SSIO}$ | | | 1.71 | – | $V_{DDA}$[17] | V |
| | | | | – | – | $V_{DDA}$ + 0.1[19] | |
| $V_{CCA}$ | Direct analog core voltage input (Analog regulator bypass) | Analog core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| $V_{CCD}$ | Direct digital core voltage input (Digital regulator bypass) | Digital core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| **Active Mode** | | | | | | | |
| $I_{DD}$[20] | Sum of digital and analog $I_{DDD}$ + $I_{DDA}$. $I_{DDIOX}$ for I/Os not included. IMO enabled, bus clock and CPU clock enabled. CPU executing complex program from flash | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 3 MHz[21] | T = –40 °C | – | 1.9 | 3.8 | mA |
| | | | T = 25 °C | – | 1.9 | 3.8 | |
| | | | T = 85 °C | – | 2 | 3.8 | |
| | | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 6 MHz | T = –40 °C | – | 3.1 | 5 | |
| | | | T = 25 °C | – | 3.1 | 5 | |
| | | | T = 85 °C | – | 3.2 | 5 | |
| | | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 12 MHz[21] | T = –40 °C | – | 5.4 | 7 | |
| | | | T = 25 °C | – | 5.4 | 7 | |
| | | | T = 85 °C | – | 5.6 | 7 | |
| | | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 24 MHz[21] | T = –40 °C | – | 8.9 | 10.5 | |
| | | | T = 25 °C | – | 8.9 | 10.5 | |
| | | | T = 85 °C | – | 9.1 | 10.5 | |
| | | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 48 MHz[21] | T = –40 °C | – | 15.5 | 17 | |
| | | | T = 25 °C | – | 15.4 | 17 | |
| | | | T = 85 °C | – | 15.7 | 17 | |
| | | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 62 MHz | T = –40 °C | – | 18 | 19.5 | |
| | | | T = 25 °C | – | 18 | 19.5 | |
| | | | T = 85 °C | – | 18.5 | 19.5 | |
| | | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 74 MHz | T = –40 °C | – | 26.5 | 30 | |
| | | | T = 25 °C | – | 26.5 | 30 | |
| | | | T = 85 °C | – | 27 | 30 | |
| | | $V_{DDX}$ = 2.7 V to 5.5 V; $F_{CPU}$ = 80 MHz, IMO = 3 MHz with PLL | T = –40 °C | – | 22 | 25.5 | |
| | | | T = 25 °C | – | 22 | 25.5 | |
| | | | T = 85 °C | – | 22.5 | 25.5 | |

**Notes**
17. The power supplies can be brought up in any sequence however once stable $V_{DDA}$ must be greater than or equal to all other supplies.
18. The $V_{DDIO}$ supply voltage must be greater than the maximum voltage on the associated GPIO pins. Maximum voltage on GPIO pin ≤ $V_{DDIO}$ ≤ $V_{DDA}$.
19. Guaranteed by design, not production tested.
20. The current consumption of additional peripherals that are implemented only in programmed logic blocks can be found in their respective datasheets, available in PSoC Creator, the integrated design environment. To estimate total current, find CPU current at frequency of interest and add peripheral currents for your particular system from the device datasheet and component datasheets.
21. Based on device characterization (Not production tested).

**Table 11-2. DC Specifications** (continued)

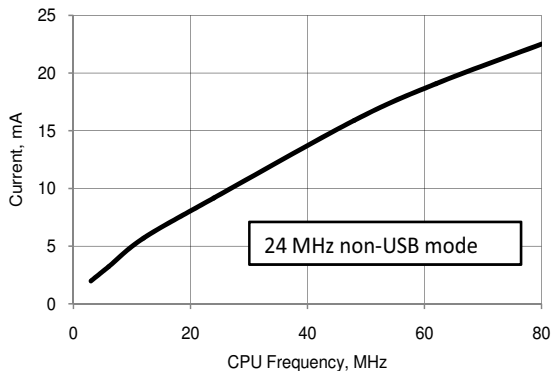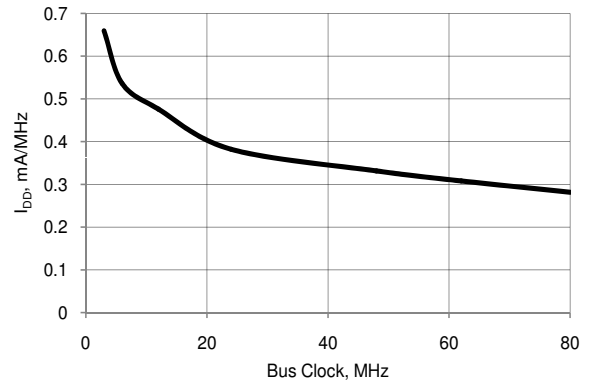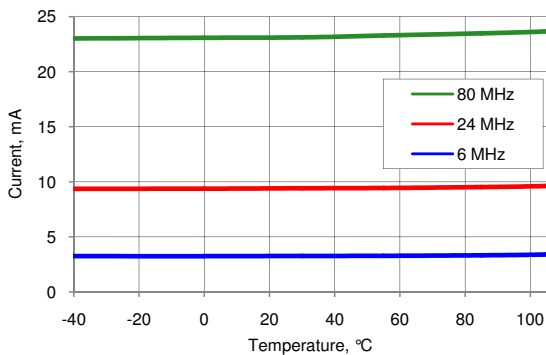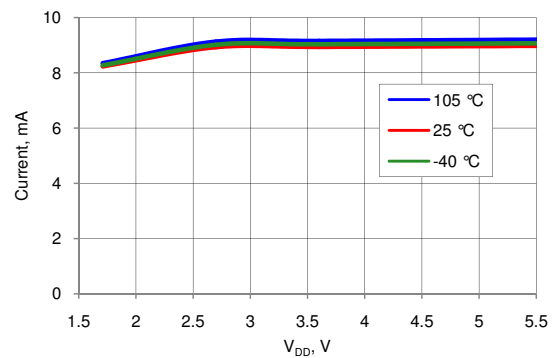| Parameter | Description | Conditions | | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|---|
| $I_{DD}$[22] | **Sleep Mode**[23] | | | | | | |
| | CPU = OFF<br>RTC = ON (= ECO32K ON, in low-power mode)<br>Sleep timer = ON (= ILO ON at 1 kHz)[24]<br>WDT = OFF<br>I²C Wake = OFF<br>Comparator = OFF<br>POR = ON<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO} =$ 4.5–5.5 V | T = –40 °C | – | 1.9 | 3.1 | µA |
| | | | T = 25 °C | – | 2.4 | 3.6 | |
| | | | T = 85 °C | – | 5 | 16 | |
| | | $V_{DD} = V_{DDIO} =$ 2.7–3.6 V | T = –40 °C | – | 1.7 | 3.1 | |
| | | | T = 25 °C | – | 2 | 3.6 | |
| | | | T = 85 °C | – | 4.2 | 16 | |
| | | $V_{DD} = V_{DDIO} =$ 1.71–1.95 V | T = –40 °C | – | 1.6 | 3.1 | |
| | | | T = 25 °C | – | 1.9 | 3.6 | |
| | | | T = 85 °C | – | 4.2 | 16 | |
| | Comparator = ON<br>CPU = OFF<br>RTC = OFF<br>Sleep timer = OFF<br>WDT = OFF<br>I2C Wake = OFF<br>POR = ON<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO} =$ 2.7–3.6 V[25] | T = 25 °C | – | 3 | 4.2 | µA |
| | I2C Wake = ON<br>CPU = OFF<br>RTC = OFF<br>Sleep timer = OFF<br>WDT = OFF<br>Comparator = OFF<br>POR = ON<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO} =$ 2.7–3.6 V[25] | T = 25 °C | – | 1.7 | 3.6 | µA |
| | **Hibernate Mode** | | | | | | |
| | Hibernate mode current<br>All regulators and oscillators off.<br>SRAM retention<br>GPIO interrupts are active<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO} =$ 4.5–5.5 V | T = –40 °C | – | 0.2 | 2 | µA |
| | | | T = 25 °C | – | 0.24 | 2 | |
| | | | T = 85 °C | – | 2.6 | 15 | |
| | | $V_{DD} = V_{DDIO} =$ 2.7–3.6 V | T = –40 °C | – | 0.11 | 2 | |
| | | | T = 25 °C | – | 0.3 | 2 | |
| | | | T = 85 °C | – | 2 | 15 | |
| | | $V_{DD} = V_{DDIO} =$ 1.71–1.95 V | T = –40 °C | – | 0.9 | 2 | |
| | | | T = 25 °C | – | 0.11 | 2 | |
| | | | T = 85 °C | – | 1.8 | 15 | |
| $I_{DDAR}$[25] | Analog current consumption while device is reset | $V_{DDA} \le 3.6$ V | | – | 0.3 | 0.6 | mA |
| | | $V_{DDA} > 3.6$ V | | – | 1.4 | 3.3 | mA |
| $I_{DDDR}$[25] | Digital current consumption while device is reset | $V_{DDD} \le 3.6$ V | | – | 1.1 | 3.1 | mA |
| | | $V_{DDD} > 3.6$ V | | – | 0.7 | 3.1 | mA |
| $I_{DD\_PROG}$[25] | Current consumption while device programming. Sum of digital, analog, and I/Os: $I_{DDD} + I_{DDA} + I_{DDIOX}$ | | | – | 15 | 21 | mA |

**Notes**
22. The current consumption of additional peripherals that are implemented only in programmed logic blocks can be found in their respective datasheets, available in PSoC Creator, the integrated design environment. To estimate total current, find CPU current at frequency of interest and add peripheral currents for your particular system from the device datasheet and component datasheets.
23. If $V_{CCD}$ and $V_{CCA}$ are externally regulated, the voltage difference between $V_{CCD}$ and $V_{CCA}$ must be less than 50 mV.
24. Sleep timer generates periodic interrupts to wake up the CPU. This specification applies only to those times that the CPU is off.
25. Based on device characterization (Not production tested). USBIO pins tied to ground (VSSD).

Figure 11-1. Active Mode Current vs $F_{CPU}$, $V_{DD}$ = 3.3 V, Temperature = 25 °C



Figure 11-2. $I_{DD}$ vs Frequency at 25 °C



Figure 11-3. Active Mode Current vs Temperature and $F_{CPU}$, $V_{DD}$ = 3.3 V



Figure 11-4. Active Mode Current vs $V_{DD}$ and Temperature, $F_{CPU}$ = 24 MHz



Table 11-3. AC Specifications

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{CPU}$ | CPU frequency | 1.71 V ≤ $V_{DDD}$ ≤ 5.5 V | DC | – | 80.01 | MHz |
| $F_{BUSCLK}$ | Bus frequency | 1.71 V ≤ $V_{DDD}$ ≤ 5.5 V | DC | – | 80.01 | MHz |
| $S_{VDD}$ | $V_{DD}$ ramp rate | | – | – | 0.066 | V/µs |
| $T_{IO\_INIT}$[26] | Time from $V_{DDD}$/$V_{DDA}$/$V_{CCD}$/$V_{CCA}$ ≥ IPOR to I/O ports set to their reset states | | – | – | 10 | µs |
| $T_{STARTUP}$[26] | Time from $V_{DDD}$/$V_{DDA}$/$V_{CCD}$/$V_{CCA}$ ≥ PRES to CPU executing code at reset vector | $V_{CCA}$/$V_{DDA}$ = regulated from $V_{DDA}$/$V_{DDD}$, no PLL used, fast IMO boot mode (48 MHz typ.) | – | – | 33 | µs |
| | | $V_{CCA}$/$V_{CCD}$ = regulated from $V_{DDA}$/$V_{DDD}$, no PLL used, IMO boot mode (12 MHz typ.) | – | – | 66 | µs |
| $T_{SLEEP}$[26] | Wakeup from sleep mode – Application of non-LVD interrupt to beginning of execution of next CPU instruction | | – | – | 25 | µs |
| $T_{HIBERNATE}$[26] | Wakeup form hibernate mode – Application of external interrupt to beginning of execution of next CPU instruction | | – | – | 150 | µs |

Note
26. Based on device characterization (not production tested).

## 11.3 Power Regulators

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted.

### 11.3.1 Digital Core Regulator

**Table 11-4. Digital Core Regulator DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|-----------|-------------|------------|-----|-----|-----|-------|
| $V_{DDD}$ | Input voltage | | 1.8 | – | 5.5 | V |
| $V_{CCD}$ | Output voltage | | – | 1.80 | – | V |
| | Regulator output capacitor | ±10%, X5R ceramic or better. The two $V_{CCD}$ pins must be shorted together, with as short a trace as possible, see Power System on page 26 | 0.9 | 1 | 1.1 | µF |

**Figure 11-5. Analog and Digital Regulators, $V_{CC}$ vs $V_{DD}$, 10 mA Load**
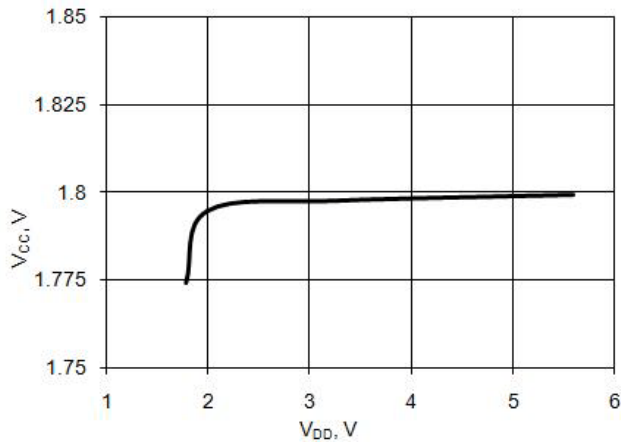


**Figure 11-6. Digital Regulator PSRR vs Frequency and $V_{DD}$**
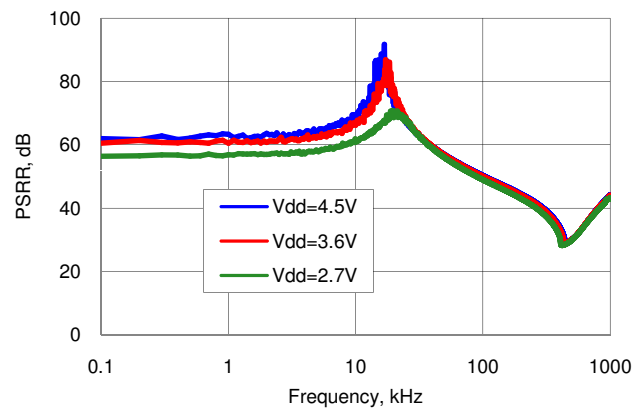


### 11.3.2 Analog Core Regulator

**Table 11-5. Analog Core Regulator DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|-----------|-------------|------------|-----|-----|-----|-------|
| $V_{DDA}$ | Input voltage | | 1.8 | – | 5.5 | V |
| $V_{CCA}$ | Output voltage | | – | 1.80 | – | V |
| | Regulator output capacitor | ±10%, X5R ceramic or better | – | 1 | – | µF |

**Figure 11-7. Analog Regulator PSRR vs Frequency and $V_{DD}$**
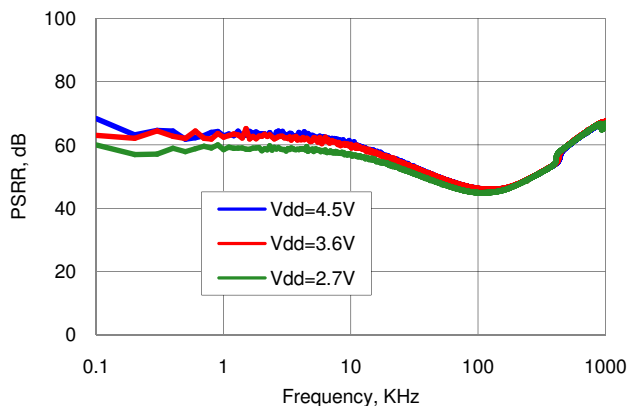
### 11.3.3 Inductive Boost Regulator

Unless otherwise specified, operating conditions are: $V_{BAT}$ = 0.5 V–3.6 V, $V_{OUT}$ = 1.8 V–5.0 V, $I_{OUT}$ = 0 mA–50 mA,
$L_{BOOST}$ = 4.7 μH–22 μH, $C_{BOOST}$ = 22 μF || 3 × 1.0 μF || 3 × 0.1 μF, $C_{BAT}$ = 22 μF, $I_F$ = 1.0 A, Unless otherwise specified, all charts
and graphs show typical values.

**Table 11-6. Inductive Boost Regulator DC Specifications**

| Parameter | Description | Conditions | | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|---|
| $V_{OUT}$ | Boost output voltage[27] | vsel = 1.8 V in register BOOST_CR0 | | 1.71 | 1.8 | 1.89 | V |
| | | vsel = 1.9 V in register BOOST_CR0 | | 1.81 | 1.90 | 2.00 | V |
| | | vsel = 2.0 V in register BOOST_CR0 | | 1.90 | 2.00 | 2.10 | V |
| | | vsel = 2.4 V in register BOOST_CR0 | | 2.16 | 2.40 | 2.64 | V |
| | | vsel = 2.7 V in register BOOST_CR0 | | 2.43 | 2.70 | 2.97 | V |
| | | vsel = 3.0 V in register BOOST_CR0 | | 2.70 | 3.00 | 3.30 | V |
| | | vsel = 3.3 V in register BOOST_CR0 | | 2.97 | 3.30 | 3.63 | V |
| | | vsel = 3.6 V in register BOOST_CR0 | | 3.24 | 3.60 | 3.96 | V |
| | | vsel = 5.0 V in register BOOST_CR0 | | 4.50 | 5.00 | 5.50 | V |
| $V_{BAT}$ | Input voltage to boost[28] | $I_{OUT}$ = 0 mA–5 mA | vsel = 1.8 V–2.0 V, $T_A$ = 0 °C–70 °C | 0.5 | – | 0.8 | V |
| | | $I_{OUT}$ = 0 mA–15 mA | vsel = 1.8 V–5.0 V[29], $T_A$ = –10 °C–85 °C | 1.6 | – | 3.6 | V |
| | | $I_{OUT}$ = 0 mA–25 mA | vsel = 1.8 V–2.7 V, $T_A$ = –10 °C–85 °C | 0.8 | – | 1.6 | V |
| | | $I_{OUT}$ = 0 mA–50 mA | vsel = 1.8 V–3.3 V[29], $T_A$ = –40 °C–85 °C | 1.8 | – | 2.5 | V |
| | | | vsel = 1.8 V–3.3 V[29], $T_A$ = –10 °C–85 °C | 1.3 | – | 2.5 | V |
| | | | vsel = 2.5 V–5.0 V[29], $T_A$ = –10 °C–85 °C | 2.5 | – | 3.6 | V |
| $I_{OUT}$ | Output current | $T_A$ = 0 °C–70 °C | $V_{BAT}$ = 0.5 V–0.8 V | 0 | – | 5 | mA |
| | | $T_A$ = –10 °C–85 °C | $V_{BAT}$ = 1.6 V–3.6 V | 0 | – | 15 | mA |
| | | | $V_{BAT}$ = 0.8 V–1.6 V | 0 | – | 25 | mA |
| | | | $V_{BAT}$ = 1.3 V–2.5 V | 0 | – | 50 | mA |
| | | | $V_{BAT}$ = 2.5 V–3.6 V | 0 | – | 50 | mA |
| | | $T_A$ = –40 °C–85 °C | $V_{BAT}$ = 1.8 V–2.5 V | 0 | – | 50 | mA |
| $I_{LPK}$ | Inductor peak current | | | – | – | 700 | mA |
| $I_Q$ | Quiescent current | Boost active mode | | – | 250 | – | μA |
| | | Boost sleep mode, $I_{OUT}$ < 1 μA | | – | 25 | – | μA |
| $Reg_{LOAD}$ | Load regulation | | | – | – | 10 | % |
| $Reg_{LINE}$ | Line regulation | | | – | – | 10 | % |

**Notes**
27. Listed vsel options are characterized. Additional vsel options are valid and guaranteed by design.
28. The boost will start at all valid $V_{BAT}$ conditions including down to $V_{BAT}$ = 0.5 V.
29. If $V_{BAT}$ is greater than or equal to $V_{OUT}$ boost setting, then $V_{OUT}$ will be less than $V_{BAT}$ due to resistive losses in the boost circuit.

### Table 11-7. Recommended External Components for Boost Circuit

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $L_{BOOST}$ | Boost inductor | 4.7 µH nominal | 3.7 | 4.7 | 5.7 | µH |
| | | 10 µH nominal | 8.0 | 10.0 | 12.0 | µH |
| | | 22 µH nominal | 17.0 | 22.0 | 27.0 | µH |
| $C_{BOOST}$ | Total capacitance sum of $V_{DDD}$, $V_{DDA}$, $V_{DDIO}$[30] | | 17.0 | 26.0 | 31.0 | µF |
| $C_{BAT}$ | Battery filter capacitor | | 17.0 | 22.0 | 27.0 | µF |
| $I_F$ | Schottky diode average forward current | | 1.0 | – | – | A |
| $V_R$ | Schottky reverse voltage | | 20.0 | – | – | V |

Figure 11-8. $T_A$ range over $V_{BAT}$ and $V_{OUT}$
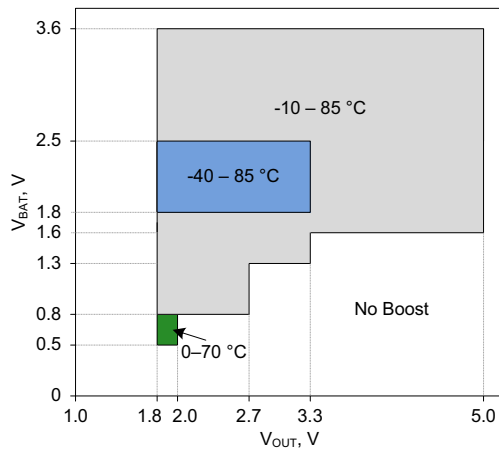


Figure 11-9. $I_{OUT}$ range over $V_{BAT}$ and $V_{OUT}$
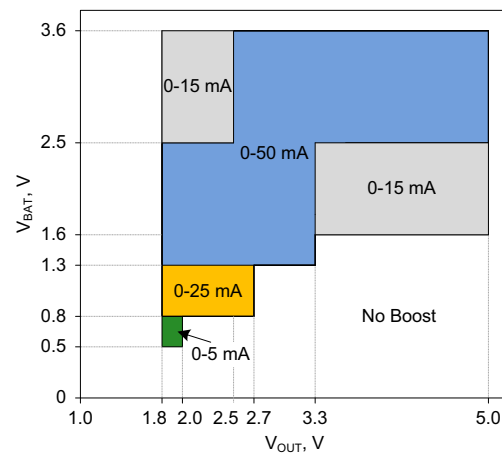


Figure 11-10. $L_{BOOST}$ values over $V_{BAT}$ and $V_{OUT}$
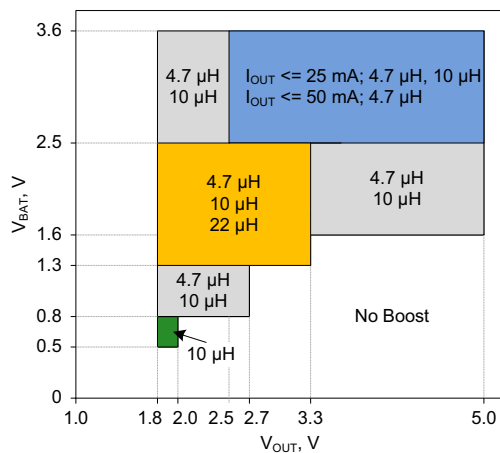


**Note**
30. Based on device characterization (Not production tested).

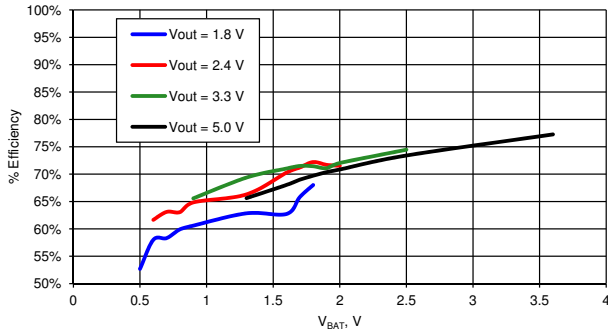**Figure 11-11. Efficiency vs V_BAT, L_BOOST = 4.7 µH** [31]



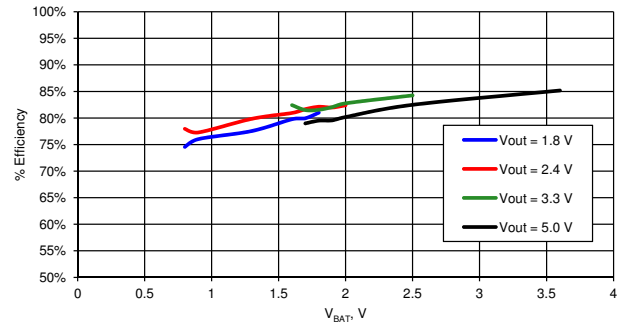**Figure 11-12. Efficiency vs V_BAT, L_BOOST = 10 µH** [31]



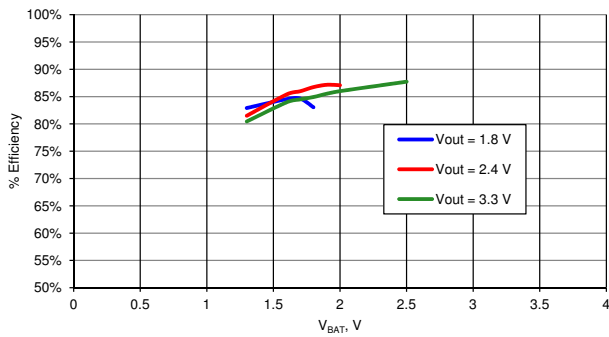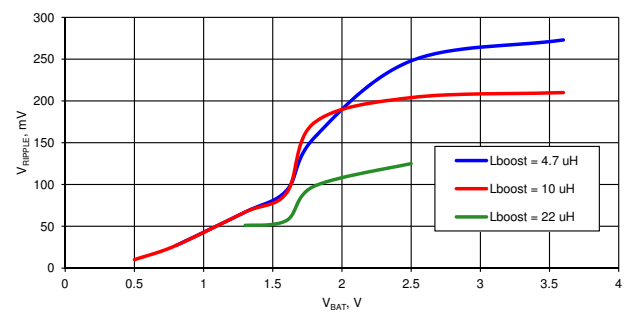**Figure 11-13. Efficiency vs V_BAT, L_BOOST = 22 µH** [31]



**Figure 11-14. V_RIPPLE vs V_BAT** [31]



**Note**
31. Typical example. Actual values may vary depending on external component selection, PCB layout, and other design parameters.

## 11.4 Inputs and Outputs

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted. Unless otherwise specified, all charts and graphs show typical values.

When the power supplies ramp up, the pin voltages are indeterminate until both $V_{DDIO}$ and $V_{DDA}$ reach the IPOR voltage, which can be as high as 1.45 V. At that point, the pins change to their normal NVL settings.

### 11.4.1 GPIO

**Table 11-8. GPIO DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IH}$ | Input voltage high threshold | CMOS Input, PRT[x]CTL = 0 | $0.7 \times V_{DDIO}$ | – | – | V |
| $V_{IL}$ | Input voltage low threshold | CMOS Input, PRT[x]CTL = 0 | – | – | $0.3 \times V_{DDIO}$ | V |
| $V_{IH}$ | Input voltage high threshold | LVTTL Input, PRT[x]CTL = 1, $V_{DDIO}$ < 2.7 V | $0.7 \times V_{DDIO}$ | – | – | V |
| $V_{IH}$ | Input voltage high threshold | LVTTL Input, PRT[x]CTL = 1, $V_{DDIO}$ ≥ 2.7 V | 2.0 | – | – | V |
| $V_{IL}$ | Input voltage low threshold | LVTTL Input, PRT[x]CTL = 1, $V_{DDIO}$ < 2.7 V | – | – | $0.3 \times V_{DDIO}$ | V |
| $V_{IL}$ | Input voltage low threshold | LVTTL Input, PRT[x]CTL = 1, $V_{DDIO}$ ≥ 2.7 V | – | – | 0.8 | V |
| $V_{OH}$ | Output voltage high | $I_{OH}$ = 4 mA at 3.3 $V_{DDIO}$ | $V_{DDIO}$ – 0.6 | – | – | V |
| | | $I_{OH}$ = 1 mA at 1.8 $V_{DDIO}$ | $V_{DDIO}$ – 0.5 | – | – | V |
| $V_{OL}$ | Output voltage low | $I_{OL}$ = 8 mA at 3.3 $V_{DDIO}$ | – | – | 0.6 | V |
| | | $I_{OL}$ = 3 mA at 3.3 $V_{DDIO}$ | – | – | 0.4 | V |
| | | $I_{OL}$ = 4 mA at 1.8 $V_{DDIO}$ | – | – | 0.6 | V |
| Rpullup | Pull up resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| Rpulldown | Pull down resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| $I_{IL}$ | Input leakage current (absolute value)[32] | 25 °C, $V_{DDIO}$ = 3.0 V | – | – | 2 | nA |
| $C_{IN}$ | Input capacitance[32] | P0.0, P0.1, P0.2, P3.6, P3.7 | – | 17 | 20 | pF |
| | | P0.3, P0.4, P3.0, P3.1, P3.2 | – | 10 | 15 | pF |
| | | P0.6, P0.7, P15.0, P15.6, P15.7[33] | – | 7 | 12 | pF |
| | | All other GPIOs | – | 5 | 9 | pF |
| $V_H$ | Input voltage hysteresis (Schmitt-Trigger)[32] | | – | 40 | – | mV |
| Idiode | Current through protection diode to $V_{DDIO}$ and $V_{SSIO}$ | | – | – | 100 | µA |
| Rglobal | Resistance pin to analog global bus | 25 °C, $V_{DDIO}$ = 3.0 V | – | 320 | – | Ω |
| Rmux | Resistance pin to analog mux bus | 25 °C, $V_{DDIO}$ = 3.0 V | – | 220 | – | Ω |

**Notes**
32. Based on device characterization (Not production tested).
33. For information on designing with PSoC oscillators, refer to the application note, AN54439 - PSoC® 3 and PSoC 5 External Oscillator.

**Figure 11-15. GPIO Output High Voltage and Current**
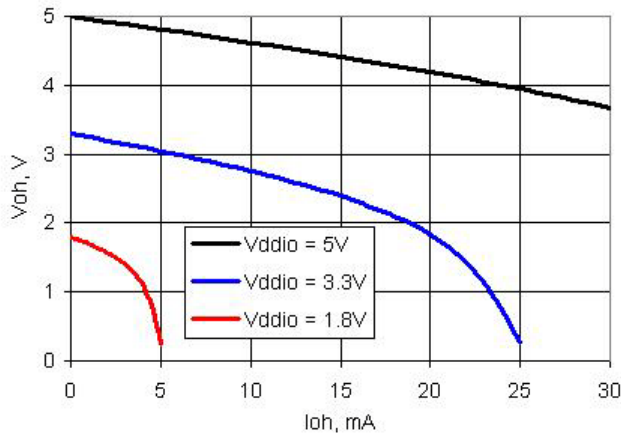


**Figure 11-16. GPIO Output Low Voltage and Current**
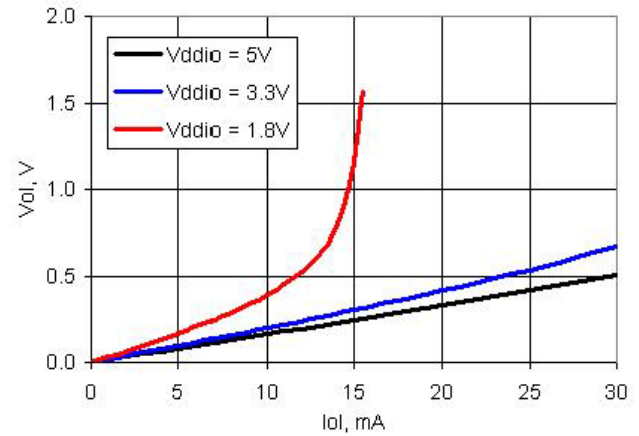


**Table 11-9. GPIO AC Specifications[34]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| TriseF | Rise time in Fast Strong Mode | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 6 | ns |
| TfallF | Fall time in Fast Strong Mode | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 6 | ns |
| TriseS | Rise time in Slow Strong Mode | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 60 | ns |
| TfallS | Fall time in Slow Strong Mode | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 60 | ns |
| Fgpioout | GPIO output operating frequency | | | | | |
| | 2.7 V ≤ $V_{DDIO}$ ≤ 5.5 V, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 33 | MHz |
| | 1.71 V ≤ $V_{DDIO}$ < 2.7 V, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 20 | MHz |
| | 3.3 V ≤ $V_{DDIO}$ ≤ 5.5 V, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 7 | MHz |
| | 1.71 V ≤ $V_{DDIO}$ < 3.3 V, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 3.5 | MHz |
| Fgpioin | GPIO input operating frequency | 90/10% $V_{DDIO}$ | – | – | 33 | MHz |

**Note**
34. Based on device characterization (Not production tested).

### 11.4.2 SIO

**Table 11-10. SIO DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Vinmax | Maximum input voltage | All allowed values of $V_{DDIO}$ and Vddd, see *Section 11.1* | – | – | 5.5 | V |
| Vinref | Input voltage reference (Differential input mode) | | 0.5 | – | $0.52 \times V_{DDIO}$ | V |
| Voutref | Output voltage reference (Regulated output mode) | | | | | |
| | | $V_{DDIO} > 3.7$ | 1 | – | $V_{DDIO} - 1$ | V |
| | | $V_{DDIO} < 3.7$ | 1 | – | $V_{DDIO} - 0.5$ | V |
| $V_{IH}$ | Input voltage high threshold | | | | | |
| | GPIO mode | CMOS input | $0.7 \times V_{DDIO}$ | – | – | V |
| | Differential input mode[35] | Hysteresis disabled | SIO_ref + 0.2 | – | – | V |
| $V_{IL}$ | Input voltage low threshold | | | | | |
| | GPIO mode | CMOS input | – | – | $0.3 \times V_{DDIO}$ | V |
| | Differential input mode[35] | Hysteresis disabled | – | – | SIO_ref – 0.2 | V |
| $V_{OH}$ | Output voltage high | | | | | |
| | Unregulated mode | $I_{OH}$ = 4 mA, $V_{DDIO}$ = 3.3 V | $V_{DDIO} - 0.4$ | – | – | V |
| | Regulated mode[35] | $I_{OH}$ = 1 mA | SIO_ref – 0.65 | – | SIO_ref + 0.2 | V |
| | | $I_{OH}$ = 0.1 mA | SIO_ref – 0.3 | – | SIO_ref + 0.2 | V |
| | | no load, $I_{OH}$ = 0 | SIO_ref – 0.1 | – | SIO_ref + 0.1 | V |
| $V_{OL}$ | Output voltage low | $V_{DDIO}$ = 3.30 V, $I_{OL}$ = 25 mA | – | – | 0.8 | V |
| | | $V_{DDIO}$ = 3.30 V, $I_{OL}$ = 20 mA | – | – | 0.4 | V |
| | | $V_{DDIO}$ = 1.80 V, $I_{OL}$ = 4 mA | – | – | 0.4 | V |
| Rpullup | Pull up resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| Rpulldown | Pull down resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| $I_{IL}$ | Input leakage current (absolute value)[36] | | | | | |
| | | $V_{IH} \leq V_{DDSIO}$ | 25 °C, $V_{DDSIO}$ = 3.0 V, $V_{IH}$ = 3.0 V | – | – | 14 | nA |
| | $V_{IH} > V_{DDSIO}$ | 25 °C, $V_{DDSIO}$ = 0 V, $V_{IH}$ = 3.0 V | – | – | 10 | µA |
| $C_{IN}$ | Input Capacitance[36] | | – | – | 9 | pF |
| $V_H$ | Input voltage hysteresis (Schmitt-Trigger)[36] | Single ended mode (GPIO mode) | – | 115 | – | mV |
| | | Differential mode | – | 50 | – | mV |
| Idiode | Current through protection diode to $V_{SSIO}$ | | – | – | 100 | µA |

**Notes**
35. See *Figure 6-10* on page 34 and *Figure 6-13* on page 38 for more information on SIO reference.
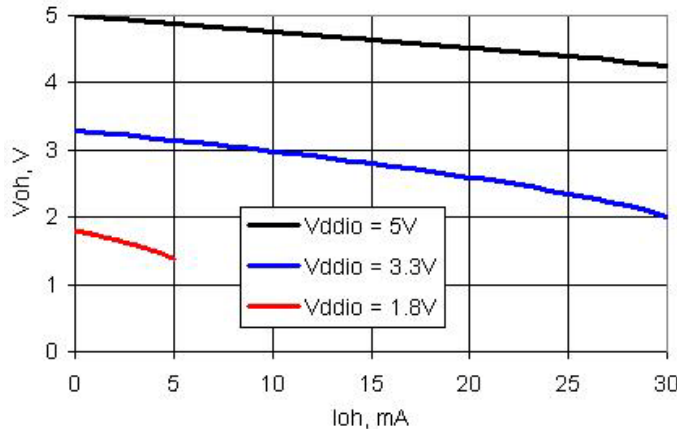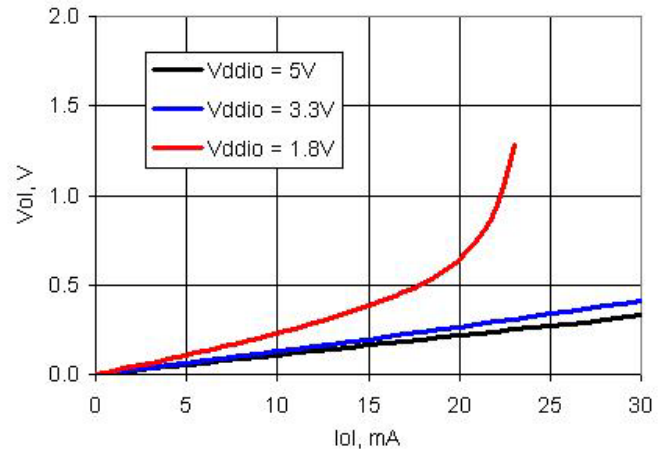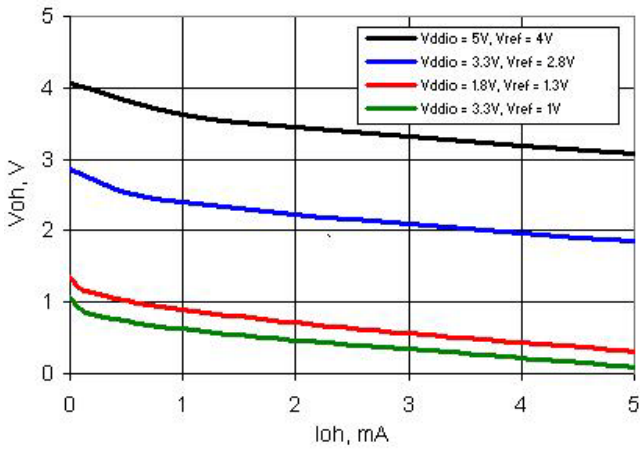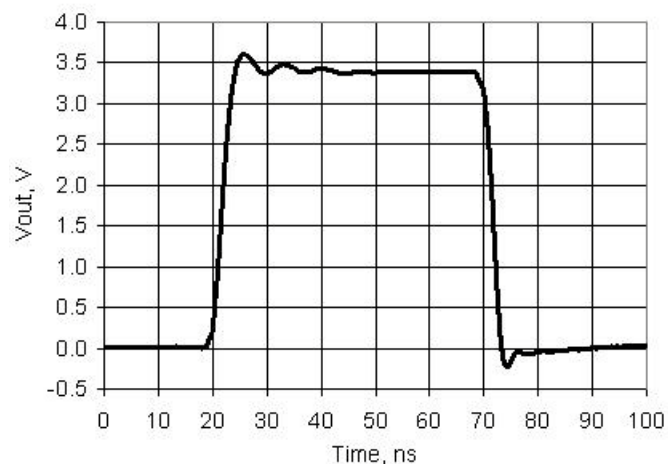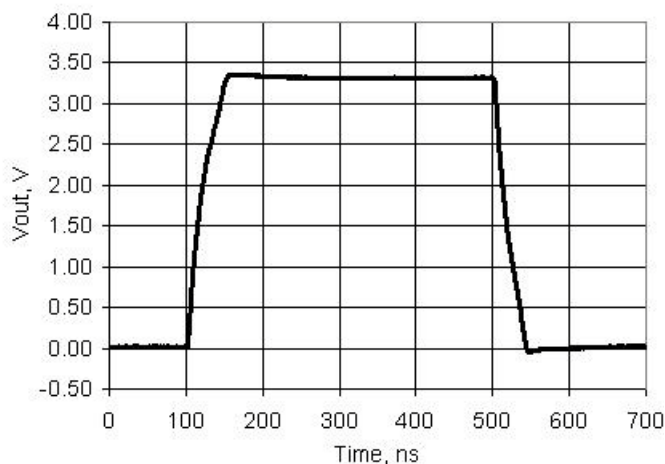36. Based on device characterization (Not production tested).

**Figure 11-17. SIO Output High Voltage and Current, Unregulated Mode**



**Figure 11-18. SIO Output Low Voltage and Current, Unregulated Mode**



**Figure 11-19. SIO Output High Voltage and Current, Regulated Mode**

**Table 11-11. SIO AC Specifications**[37]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| TriseF | Rise time in Fast Strong Mode (90/10%) | Cload = 25 pF, $V_{DDIO}$ = 3.3 V | – | – | 12 | ns |
| TfallF | Fall time in Fast Strong Mode (90/10%) | Cload = 25 pF, $V_{DDIO}$ = 3.3 V | – | – | 12 | ns |
| TriseS | Rise time in Slow Strong Mode (90/10%) | Cload = 25 pF, $V_{DDIO}$ = 3.0 V | – | – | 75 | ns |
| TfallS | Fall time in Slow Strong Mode (90/10%) | Cload = 25 pF, $V_{DDIO}$ = 3.0 V | – | – | 60 | ns |
| Fsioout | SIO output operating frequency | | | | | |
| | 2.7 V < $V_{DDIO}$ < 5.5 V, Unregulated output (GPIO) mode, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 33 | MHz |
| | 1.71 V < $V_{DDIO}$ < 2.7 V, Unregulated output (GPIO) mode, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 16 | MHz |
| | 3.3 V < $V_{DDIO}$ < 5.5 V, Unregulated output (GPIO) mode, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 5 | MHz |
| | 1.71 V < $V_{DDIO}$ < 3.3 V, Unregulated output (GPIO) mode, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 4 | MHz |
| | 2.7 V < $V_{DDIO}$ < 5.5 V, Regulated output mode, fast strong drive mode | Output continuously switching into 25 pF | – | – | 20 | MHz |
| | 1.71 V < $V_{DDIO}$ < 2.7 V, Regulated output mode, fast strong drive mode | Output continuously switching into 25 pF | – | – | 10 | MHz |
| | 1.71 V < $V_{DDIO}$ < 5.5 V, Regulated output mode, slow strong drive mode | Output continuously switching into 25 pF | – | – | 2.5 | MHz |
| Fsioin | SIO input operating frequency | | | | | |
| | 1.71 V ≤ $V_{DDIO}$ ≤ 5.5 V | 90/10% $V_{DDIO}$ | – | – | 33 | MHz |

**Figure 11-20. SIO Output Rise and Fall Times, Fast Strong Mode, $V_{DDIO}$ = 3.3 V, 25 pF Load**



**Figure 11-21. SIO Output Rise and Fall Times, Slow Strong Mode, $V_{DDIO}$ = 3.3 V, 25 pF Load**



**Note**
37. Based on device characterization (Not production tested).

**Table 11-12. SIO Comparator Specifications[38]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Vos | Offset voltage | $V_{DDIO}$ = 2 V | – | – | 68 | mV |
| | | $V_{DDIO}$ = 2.7 V | – | – | 72 | |
| | | $V_{DDIO}$ = 5.5 V | – | – | 82 | |
| TCVos | Offset voltage drift with temp | | – | – | 250 | µV/°C |
| CMRR | Common mode rejection ratio | $V_{DDIO}$ = 2 V | 30 | – | – | dB |
| | | $V_{DDIO}$ = 2.7 V | 35 | – | – | |
| | | $V_{DDIO}$ = 5.5 V | 40 | – | – | |
| Tresp | Response time | | – | – | 30 | ns |

### 11.4.3 USBIO

For operation in GPIO mode, the standard range for $V_{DDD}$ applies, see Device Level Specifications on page 61.

**Table 11-13. USBIO DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Rusbi | USB D+ pull-up resistance[38] | With idle bus | 0.900 | – | 1.575 | kΩ |
| Rusba | USB D+ pull-up resistance[38] | While receiving traffic | 1.425 | – | 3.090 | kΩ |
| Vohusb | Static output high[38] | 15 kΩ ±5% to Vss, internal pull-up enabled | 2.8 | – | 3.6 | V |
| Volusb | Static output low[38] | 15 kΩ ±5% to Vss, internal pull-up enabled | – | – | 0.3 | V |
| Vihgpio | Input voltage high, GPIO mode[38] | $V_{DDD}$ = 1.8 V | 1.5 | – | – | V |
| | | $V_{DDD}$ = 3.3 V | 2 | – | – | V |
| | | $V_{DDD}$ = 5.0 V | 2 | – | – | V |
| Vilgpio | Input voltage low, GPIO mode[38] | $V_{DDD}$ = 1.8 V | – | – | 0.8 | V |
| | | $V_{DDD}$ = 3.3 V | – | – | 0.8 | V |
| | | $V_{DDD}$ = 5.0 V | – | – | 0.8 | V |
| Vohgpio | Output voltage high, GPIO mode[38] | $I_{OH}$ = 4 mA, $V_{DDD}$ = 1.8 V | 1.6 | – | – | V |
| | | $I_{OH}$ = 4 mA, $V_{DDD}$ = 3.3 V | 3.1 | – | – | V |
| | | $I_{OH}$ = 4 mA, $V_{DDD}$ = 5.0 V | 4.2 | – | – | V |
| Volgpio | Output voltage low, GPIO mode[38] | $I_{OL}$ = 4 mA, $V_{DDD}$ = 1.8 V | – | – | 0.3 | V |
| | | $I_{OL}$ = 4 mA, $V_{DDD}$ = 3.3 V | – | – | 0.3 | V |
| | | $I_{OL}$ = 4 mA, $V_{DDD}$ = 5.0 V | – | – | 0.3 | V |
| Vdi | Differential input sensitivity | \|(D+)–(D–)\| | – | – | 0.2 | V |
| Vcm | Differential input common mode range | | 0.8 | – | 2.5 | V |
| Vse | Single ended receiver threshold | | 0.8 | – | 2 | V |
| Rps2 | PS/2 pull-up resistance[38] | In PS/2 mode, with PS/2 pull-up enabled | 3 | – | 7 | kΩ |
| Rext | External USB series resistor[38] | In series with each USB pin | 21.78 (–1%) | 22 | 22.22 (+1%) | Ω |
| Zo | USB driver output impedance[38] | Including Rext | 28 | – | 44 | Ω |
| $C_{IN}$ | USB transceiver input capacitance | | – | – | 20 | pF |
| $I_{IL}$ | Input leakage current (absolute value)[38] | 25 °C, $V_{DDD}$ = 3.0 V | – | – | 2 | nA |

**Note**
38. Based on device characterization (Not production tested).

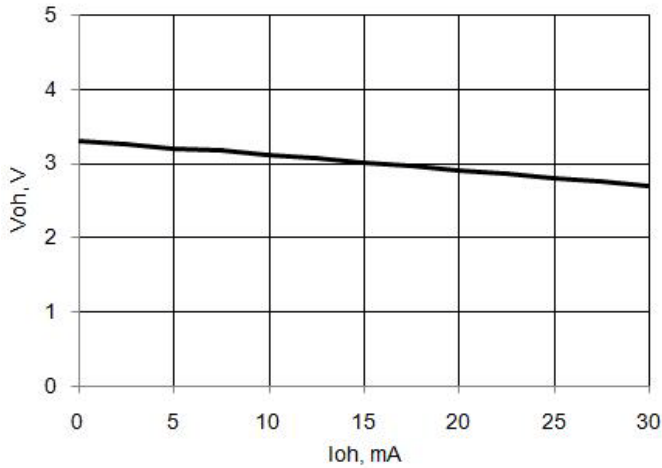**Figure 11-22. USBIO Output High Voltage and Current, GPIO Mode**



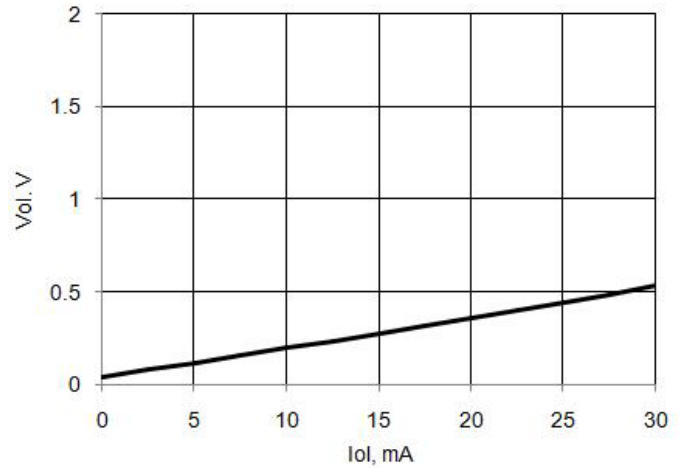**Figure 11-23. USBIO Output Low Voltage and Current, GPIO Mode**



**Table 11-14. USBIO AC Specifications[39]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Tdrate | Full-speed data rate average bit rate | | 12 – 0.25% | 12 | 12 + 0.25% | MHz |
| Tjr1 | Receiver data jitter tolerance to next transition | | –8 | – | 8 | ns |
| Tjr2 | Receiver data jitter tolerance to pair transition | | –5 | – | 5 | ns |
| Tdj1 | Driver differential jitter to next transition | | –3.5 | – | 3.5 | ns |
| Tdj2 | Driver differential jitter to pair transition | | –4 | – | 4 | ns |
| Tfdeop | Source jitter for differential transition to SE0 transition | | –2 | – | 5 | ns |
| Tfeopt | Source SE0 interval of EOP | | 160 | – | 175 | ns |
| Tfeopr | Receiver SE0 interval of EOP | | 82 | – | – | ns |
| Tfst | Width of SE0 interval during differential transition | | – | – | 14 | ns |
| Fgpio_out | GPIO mode output operating frequency | 3 V ≤ $V_{DDD}$ ≤ 5.5 V | – | – | 20 | MHz |
| | | $V_{DDD}$ = 1.71 V | – | – | 6 | MHz |
| Tr_gpio | Rise time, GPIO mode, 10%/90% $V_{DDD}$ | $V_{DDD}$ > 3 V, 25 pF load | – | – | 12 | ns |
| | | $V_{DDD}$ = 1.71 V, 25 pF load | – | – | 40 | ns |
| Tf_gpio | Fall time, GPIO mode, 90%/10% $V_{DDD}$ | $V_{DDD}$ > 3 V, 25 pF load | – | – | 12 | ns |
| | | $V_{DDD}$ = 1.71 V, 25 pF load | – | – | 40 | ns |

**Note**
39. Based on device characterization (Not production tested).

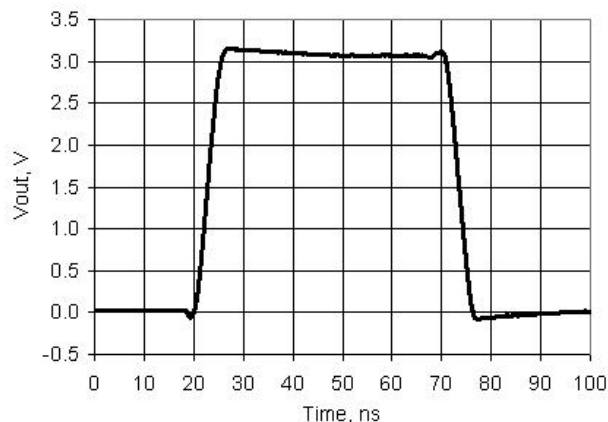**Figure 11-24. USBIO Output Rise and Fall Times, GPIO Mode, $V_{DDD}$ = 3.3 V, 25 pF Load**



**Table 11-15. USB Driver AC Specifications[40]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Tr | Transition rise time | | – | – | 20 | ns |
| Tf | Transition fall time | | – | – | 20 | ns |
| TR | Rise/fall time matching | $V_{USB\_5}$, $V_{USB\_3.3}$, see USB DC Specifications on page 91 | 90% | – | 111% | |
| Vcrs | Output signal crossover voltage | | 1.3 | – | 2 | V |

*11.4.4 XRES*

**Table 11-16. XRES DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IH}$ | Input voltage high threshold | | $0.7 \times V_{DDIO}$ | – | – | V |
| $V_{IL}$ | Input voltage low threshold | | – | – | $0.3 \times V_{DDIO}$ | V |
| Rpullup | Pull up resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| $C_{IN}$ | Input capacitance[40] | | – | 3 | | pF |
| $V_H$ | Input voltage hysteresis (Schmitt-Trigger)[40] | | – | 100 | – | mV |
| Idiode | Current through protection diode to $V_{DDIO}$ and $V_{SSIO}$ | | – | – | 100 | µA |

**Table 11-17. XRES AC Specifications[40]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{RESET}$ | Reset pulse width | | 1 | – | – | µs |

## 11.5 Analog Peripherals

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted.

*11.5.1 Voltage Reference*

**Table 11-18. Voltage Reference Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{REF}$ | Precision reference voltage | Initial trimming, 25 °C | 1.013 (–1%) | 1.024 | 1.035 (+1%) | V |

**Note**
40. Based on device characterization (Not production tested).

### 11.5.2 SAR ADC

**Table 11-19. SAR ADC DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Resolution | | – | – | 12 | bits |
| | Number of channels – single-ended | | – | – | No of GPIO | |
| | Number of channels – differential | Differential pair is formed using a pair of neighboring GPIO. | – | – | No of GPIO/2 | |
| | Monotonicity[41] | | Yes | – | – | |
| Ge | Gain error[42] | External reference | – | – | ±0.1 | % |
| $V_{OS}$ | Input offset voltage | | – | – | ±2 | mV |
| $I_{DD}$ | Current consumption[41] | | – | – | 1 | mA |
| | Input voltage range – single-ended[41] | | $V_{SSA}$ | – | $V_{DDA}$ | V |
| | Input voltage range – differential[41] | | $V_{SSA}$ | – | $V_{DDA}$ | V |
| PSRR | Power supply rejection ratio[41] | | 70 | – | – | dB |
| CMRR | Common mode rejection ratio | | 70 | – | – | dB |
| INL | Integral non linearity[41] | $V_{DDA}$ 1.71 to 5.5 V, 1 Msps, $V_{REF}$ 1 to 5.5 V, bypassed at ExtRef pin | – | – | +2/–1.5 | LSB |
| | | $V_{DDA}$ 2.0 to 3.6 V, 1 Msps, $V_{REF}$ 2 to $V_{DDA}$, bypassed at ExtRef pin | – | – | ±1.2 | LSB |
| | | $V_{DDA}$ 1.71 to 5.5 V, 500 ksps, $V_{REF}$ 1 to 5.5 V, bypassed at ExtRef pin | – | – | ±1.3 | LSB |
| DNL | Differential non linearity[41] | $V_{DDA}$ 1.71 to 5.5 V, 1 Msps, $V_{REF}$ 1 to 5.5 V, bypassed at ExtRef pin | – | – | +2/–1 | LSB |
| | | $V_{DDA}$ 2.0 to 3.6 V, 1 Msps, $V_{REF}$ 2 to $V_{DDA}$, bypassed at ExtRef pin No missing codes | – | – | 1.7/–0.99 | LSB |
| | | $V_{DDA}$ 1.71 to 5.5 V, 500 ksps, $V_{REF}$ 1 to 5.5 V, bypassed at ExtRef pin No missing codes | – | – | +2/–0.99 | LSB |
| $R_{IN}$ | Input resistance[41] | | – | 180 | – | kΩ |

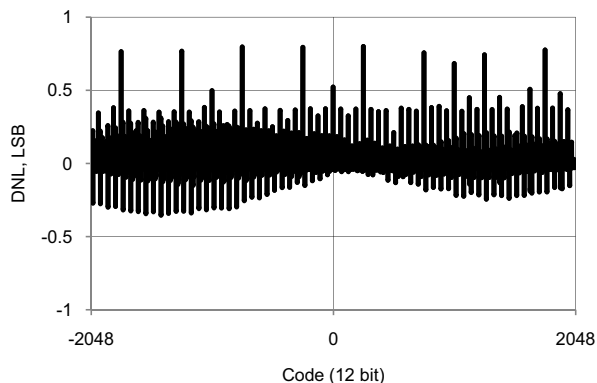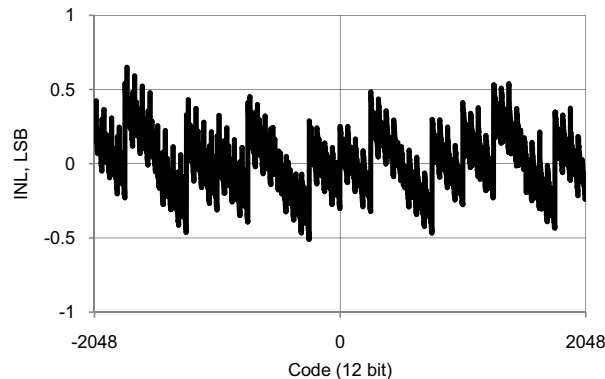**Figure 11-25. SAR ADC DNL vs Output Code, Bypassed Internal Reference Mode**



**Figure 11-26. SAR ADC INL vs Output Code, Bypassed Internal Reference Mode**



**Notes**
41. Based on device characterization (Not production tested).
42. For total analog system Idd < 5 mA, depending on package used. With higher total analog system currents it is recommended that the SAR ADC be used in differential mode.

**Figure 11-27. SAR ADC $I_{DD}$ vs sps, $V_{DDA}$ = 5 V, Continuous Sample Mode, External Reference Mode**
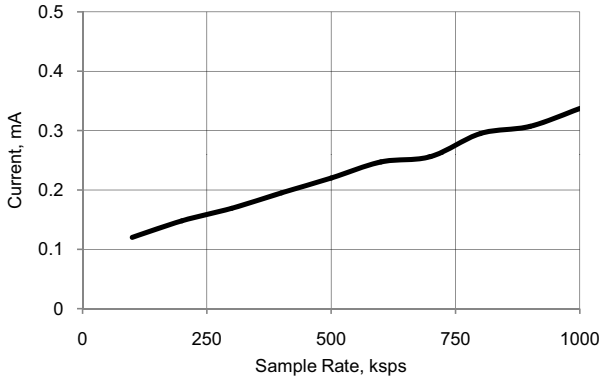


**Table 11-20. SAR ADC AC Specifications[43]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| A_SAMP_1 | Sample rate with external reference bypass cap | | – | – | 1 | Msps |
| A_SAMP_2 | Sample rate with no bypass cap. Reference = $V_{DD}$ | | – | – | 500 | Ksps |
| A_SAMP_3 | Sample rate with no bypass cap. Internal reference | | – | – | 100 | Ksps |
| | Startup time | | – | – | 10 | µs |
| SINAD | Signal-to-noise ratio | | 68 | – | – | dB |
| THD | Total harmonic distortion | | – | – | 0.02 | % |

**Figure 11-28. SAR ADC Noise Histogram, 100 ksps, Internal Reference No Bypass**
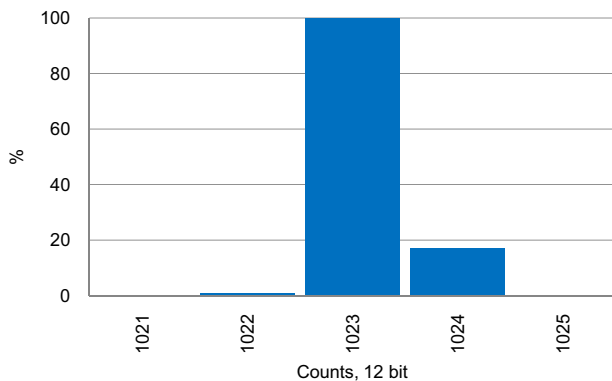


**Figure 11-29. SAR ADC Noise Histogram, 1 msps, Internal Reference Bypassed**
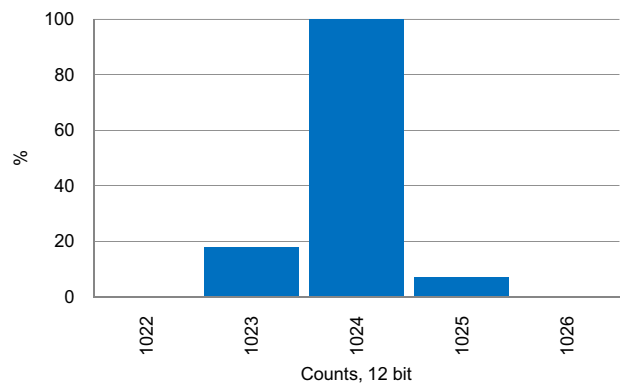


**Note**
43. Based on device characterization (Not production tested).

**Figure 11-30. SAR ADC Noise Histogram, 1 msps, External
Reference**
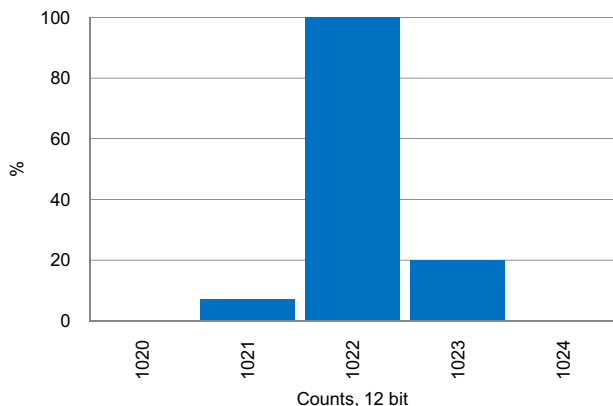


*11.5.3 Analog Globals*

**Table 11-21. Analog Globals DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Rppag | Resistance pin-to-pin through P2[4], AGL0, DSM INP, AGL1, P2[5][44, 46] | $V_{DDA}$ = 3.0 V | – | 1500 | 2200 | Ω |
| | | $V_{DDA}$ = 1.71 V | – | 1200 | 1700 | Ω |
| Rppmuxbus | Resistance pin-to-pin through P2[3], amuxbusL, P2[4][45, 46] | $V_{DDA}$ = 3.0 V | – | 700 | 1100 | Ω |
| | | $V_{DDA}$ = 1.71 V | – | 600 | 900 | Ω |

**Table 11-22. Analog Globals AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Inter-pair crosstalk for analog routes[46] | | 106 | – | – | dB |
| BWag | Analog globals 3 db bandwidth[46] | $V_{DDA}$ = 3.0 V, 25 °C | – | 26 | – | MHz |

**Notes**
44. Based on device characterization (Not production tested).
45. The resistance of the analog global and analog mux bus is high if $V_{DDA} \leq 2.7$ V, and the chip is in either sleep or hibernate mode. Use of analog global and analog mux bus under these conditions is not recommended.
46. Pin P6[4] to del-sig ADC input; calculated, not measured.

### 11.5.4 Comparator

**Table 11-23. Comparator DC Specifications**[47, 48]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{OS}$ | Input offset voltage in fast mode | Factory trim, $V_{DDA}$ > 2.7 V, $V_{IN} \geq$ 0.5 V | – | | 10 | mV |
| | Input offset voltage in slow mode | Factory trim, $V_{IN} \geq$ 0.5 V | – | | 9 | mV |
| $V_{OS}$ | Input offset voltage in fast mode | Custom trim | – | – | 4 | mV |
| | Input offset voltage in slow mode | Custom trim | – | – | 4 | mV |
| $V_{OS}$ | Input offset voltage in ultra low power mode | | – | ±12 | – | mV |
| TCVos | Temperature coefficient, input offset voltage | $V_{CM}$ = $V_{DDA}$ / 2, fast mode | – | 63 | 85 | µV/°C |
| | | $V_{CM}$ = $V_{DDA}$ / 2, slow mode | – | 15 | 20 | |
| $V_{HYST}$ | Hysteresis | Hysteresis enable mode | – | 10 | 32 | mV |
| $V_{ICM}$ | Input common mode voltage | High current / fast mode | $V_{SSA}$ | – | $V_{DDA}$ | V |
| | | Low current / slow mode | $V_{SSA}$ | – | $V_{DDA}$ | V |
| | | Ultra low power mode | $V_{SSA}$ | – | $V_{DDA}$ − 1.15 | V |
| CMRR | Common mode rejection ratio | | – | 50 | – | dB |
| $I_{CMP}$ | High current mode/fast mode | | – | – | 400 | µA |
| | Low current mode/slow mode | | – | – | 100 | µA |
| | Ultra low power mode | | – | 6 | – | µA |

**Table 11-24. Comparator AC Specifications**[47, 48]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{RESP}$ | Response time, high current mode | 50 mV overdrive, measured pin-to-pin | – | 75 | 110 | ns |
| | Response time, low current mode | 50 mV overdrive, measured pin-to-pin | – | 155 | 200 | ns |
| | Response time, ultra low power mode | 50 mV overdrive, measured pin-to-pin | – | 55 | – | µs |

**Notes**
47. The recommended procedure for using a custom trim value for the on-chip comparators are found in the TRM.
48. Based on device characterization (Not production tested).

### 11.5.5 Current Digital-to-analog Converter (IDAC)

All specifications are based on use of the low-resistance IDAC output pins (see Pin Descriptions on page 11 for details). See the IDAC component data sheet in PSoC Creator for full electrical specifications and APIs.

Unless otherwise specified, all charts and graphs show typical values.

**Table 11-25. IDAC DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Resolution | | – | – | 8 | bits |
| $I_{OUT}$ | Output current at code = 255 | Range = 2.04 mA, code = 255, $V_{DDA} \geq 2.7$ V, Rload = 600 Ω | – | 2.04 | – | mA |
| | | Range = 2.04 mA, High mode, code = 255, $V_{DDA} \leq 2.7$ V, Rload = 300 Ω | – | 2.04 | – | mA |
| | | Range = 255 µA, code = 255, Rload = 600 Ω | – | 255 | – | µA |
| | | Range = 31.875 µA, code = 255, Rload = 600 Ω | – | 31.875 | – | µA |
| | Monotonicity | | – | – | Yes | |
| Ezs | Zero scale error | | – | 0 | ±1 | LSB |
| Eg | Gain error | Range = 2.04 mA | – | – | ±2.5 | % |
| | | Range = 255 µA | – | – | ±2.5 | % |
| | | Range = 31.875 µA | – | – | ±3.5 | % |
| TC_Eg | Temperature coefficient of gain error | Range = 2.04 mA | – | – | 0.045 | % / °C |
| | | Range = 255 µA | – | – | 0.045 | % / °C |
| | | Range = 31.875 µA | – | – | 0.05 | % / °C |
| INL | Integral nonlinearity | Sink mode, range = 255 µA, Codes 8–255, Rload = 2.4 kΩ, Cload = 15 pF | – | ±0.9 | ±1 | LSB |
| | | Source mode, range = 255 µA, Codes 8–255, Rload = 2.4 kΩ, Cload = 15 pF | – | ±1.2 | ±1.6 | LSB |
| | | Source mode, range = 31.875 µA, Codes 8–255, Rload = 20 kΩ, Cload = 15 pF[49] | – | ±0.9 | ±2 | LSB |
| | | Sink mode, range = 31.875 µA, Codes 8–255, Rload = 20 kΩ, Cload = 15 pF[49] | – | ±0.9 | ±2 | LSB |
| | | Source mode, range = 2.04 mA, Codes 8–255, Rload = 600 Ω, Cload = 15 pF[49] | – | ±0.9 | ±2 | LSB |
| | | Sink mode, range = 2.04 mA, Codes 8–255, Rload = 600 Ω, Cload = 15 pF[49] | – | ±0.6 | ±1 | LSB |

**Note**
49. Based on device characterization (Not production tested).

**Table 11-25. IDAC DC Specifications** (continued)

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| DNL | Differential nonlinearity | Sink mode, range = 255 µA, Rload = 2.4 kΩ, Cload = 15 pF | – | ±0.3 | ±1 | LSB |
| | | Source mode, range = 255 µA, Rload = 2.4 kΩ, Cload = 15 pF | – | ±0.3 | ±1 | LSB |
| | | Source mode, range = 31.875 µA, Rload = 20 kΩ, Cload = 15 pF[50] | – | ±0.2 | ±1 | LSB |
| | | Sink mode, range = 31.875 µA, Rload = 20 kΩ, Cload = 15 pF[50] | – | ±0.2 | ±1 | LSB |
| | | Source mode, range = 2.0 4 mA, Rload = 600 Ω, Cload = 15 pF[50] | – | ±0.2 | ±1 | LSB |
| | | Sink mode, range = 2.0 4 mA, Rload = 600 Ω, Cload = 15 pF[50] | – | ±0.2 | ±1 | LSB |
| Vcompliance | Dropout voltage, source or sink mode | Voltage headroom at max current, Rload to $V_{DDA}$ or Rload to $V_{SSA}$, $V_{DIFF}$ from $V_{DDA}$ | 1 | – | – | V |
| $I_{DD}$ | Operating current, code = 0 | Slow mode, source mode, range = 31.875 µA | – | 44 | 100 | µA |
| | | Slow mode, source mode, range = 255 µA, | – | 33 | 100 | µA |
| | | Slow mode, source mode, range = 2.04 mA | – | 33 | 100 | µA |
| | | Slow mode, sink mode, range = 31.875 µA | – | 36 | 100 | µA |
| | | Slow mode, sink mode, range = 255 µA | – | 33 | 100 | µA |
| | | Slow mode, sink mode, range = 2.04 mA | – | 33 | 100 | µA |
| | | Fast mode, source mode, range = 31.875 µA | – | 310 | 500 | µA |
| | | Fast mode, source mode, range = 255 µA | – | 305 | 500 | µA |
| | | Fast mode, source mode, range = 2.04 mA | – | 305 | 500 | µA |
| | | Fast mode, sink mode, range = 31.875 µA | – | 310 | 500 | µA |
| | | Fast mode, sink mode, range = 255 µA | – | 300 | 500 | µA |
| | | Fast mode, sink mode, range = 2.04 mA | – | 300 | 500 | µA |

**Note**
50. Based on device characterization (Not production tested).

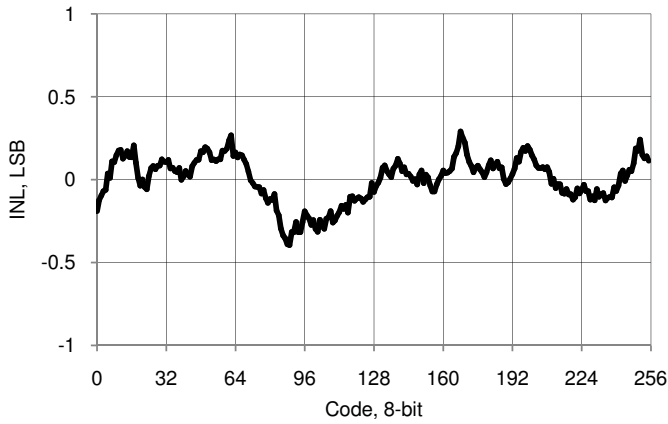**Figure 11-31. IDAC INL vs Input Code, Range = 255 μA, Source Mode**



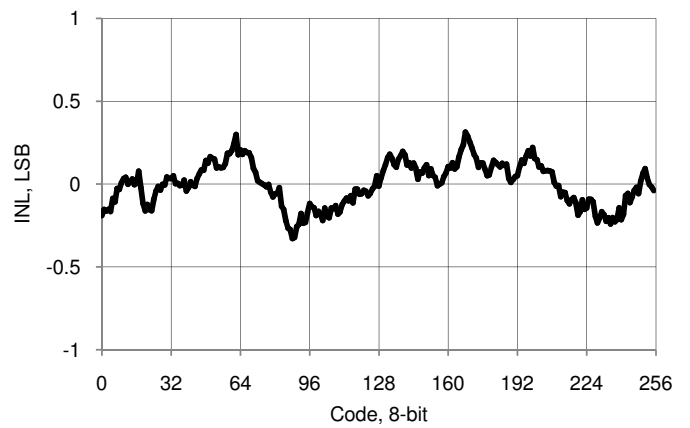**Figure 11-32. IDAC INL vs Input Code, Range = 255 μA, Sink Mode**



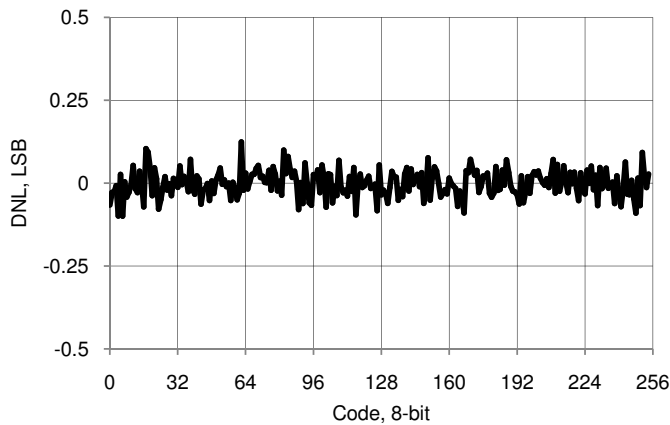**Figure 11-33. IDAC DNL vs Input Code, Range = 255 μA, Source Mode**



**Figure 11-34. IDAC DNL vs Input Code, Range = 255 μA, Sink Mode**
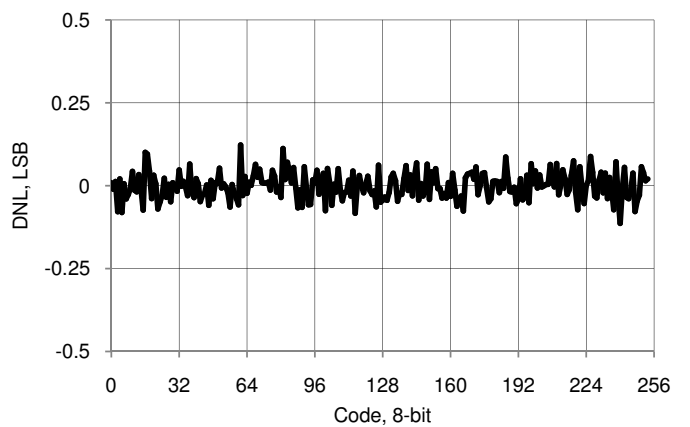

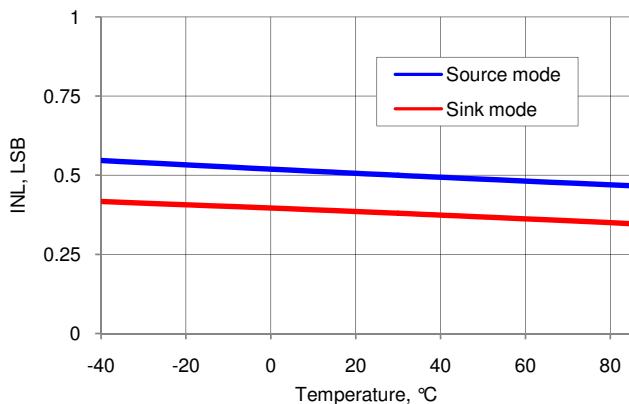
**Figure 11-35. IDAC INL vs Temperature, Range = 255 μA, Fast Mode**



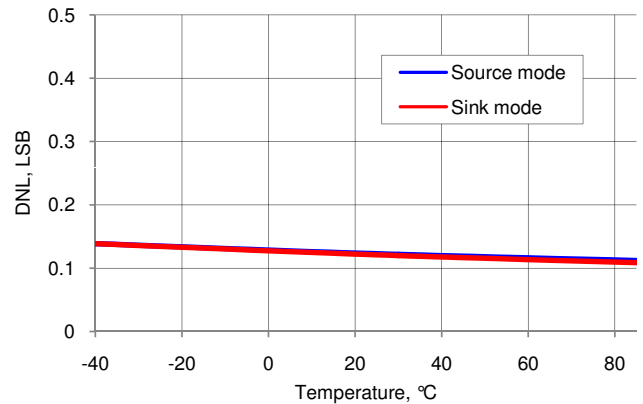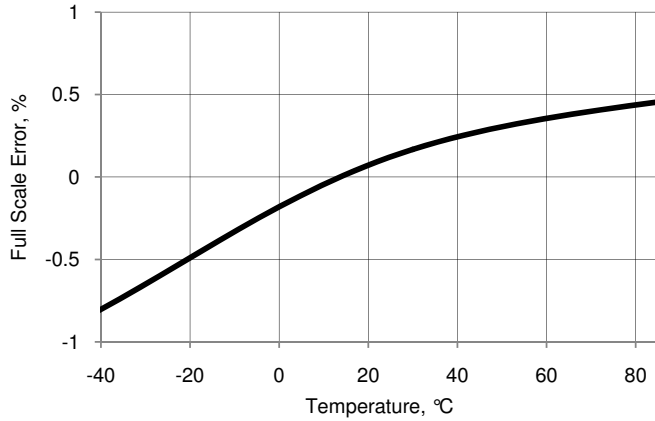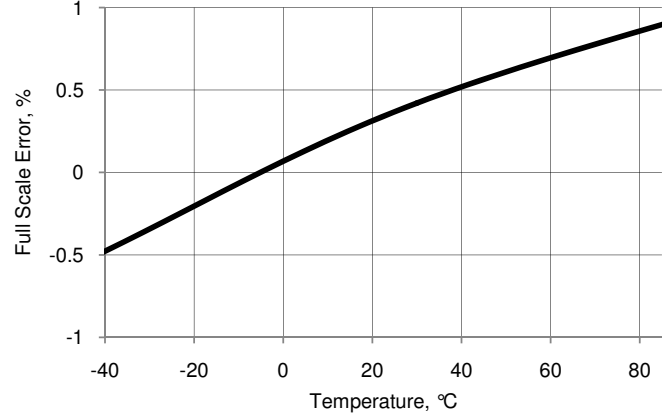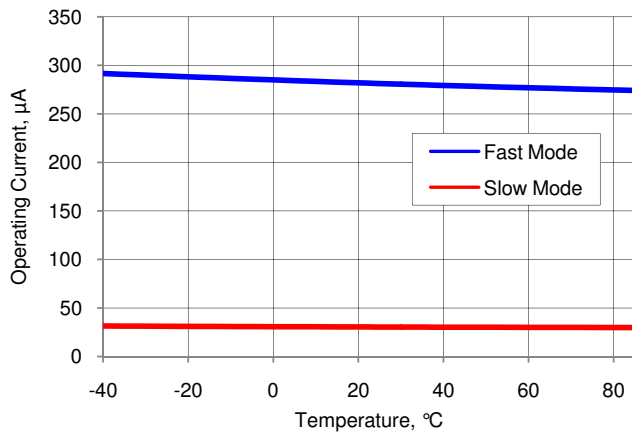**Figure 11-36. IDAC DNL vs Temperature, Range = 255 μA, Fast Mode**

**Figure 11-37. IDAC Full Scale Error vs Temperature, Range = 255 µA, Source Mode**



**Figure 11-38. IDAC Full Scale Error vs Temperature, Range = 255 µA, Sink Mode**



**Figure 11-39. IDAC Operating Current vs Temperature, Range = 255 µA, Code = 0, Source Mode**



**Figure 11-40. IDAC Operating Current vs Temperature, Range = 255 µA, Code = 0, Sink Mode**
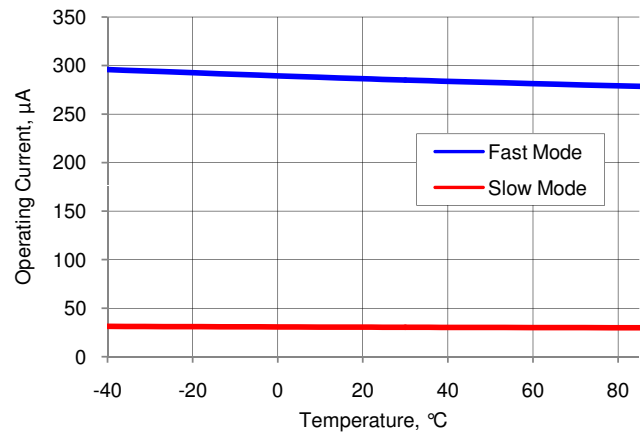


**Table 11-26.  IDAC AC Specifications[51]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{DAC}$ | Update rate | | – | – | 8 | Msps |
| $T_{SETTLE}$ | Settling time to 0.5 LSB | Range = 31.875 µA, full scale transition, fast mode, 600 Ω 15-pF load | – | – | 125 | ns |
| | | Range = 255 µA, full scale transition, fast mode, 600 Ω 15-pF load | – | – | 125 | ns |
| | Current noise | Range = 255 µA, source mode, fast mode, $V_{DDA}$ = 5 V, 10 kHz | – | 340 | – | pA/sqrtHz |

**Note**
51. Based on device characterization (Not production tested).

**Figure 11-41. IDAC Step Response, Codes 0x40 - 0xC0, 255 µA Mode, Source Mode, Fast Mode, V$_{DDA}$ = 5 V**
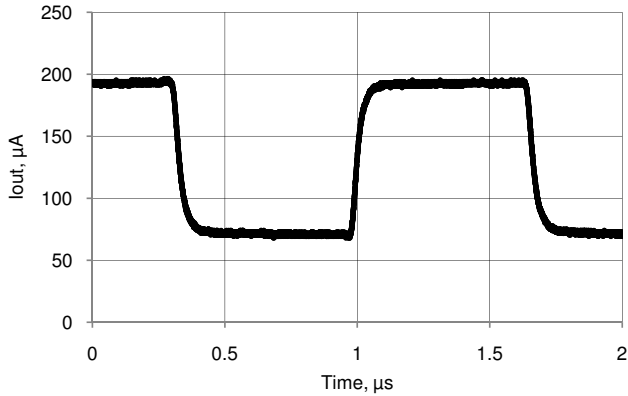


**Figure 11-42. IDAC Glitch Response, Codes 0x7F - 0x80, 255 µA Mode, Source Mode, Fast Mode, V$_{DDA}$ = 5 V**
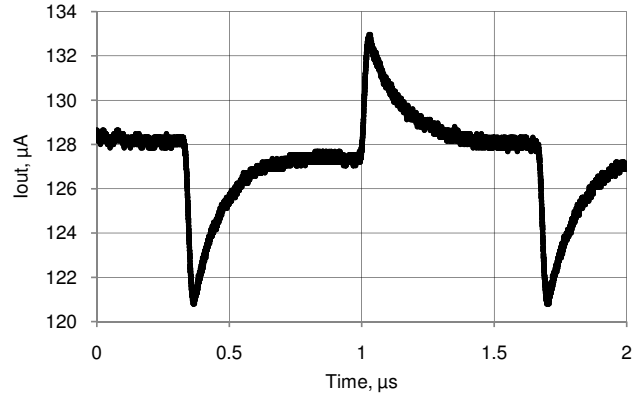


**Figure 11-43. IDAC PSRR vs Frequency**
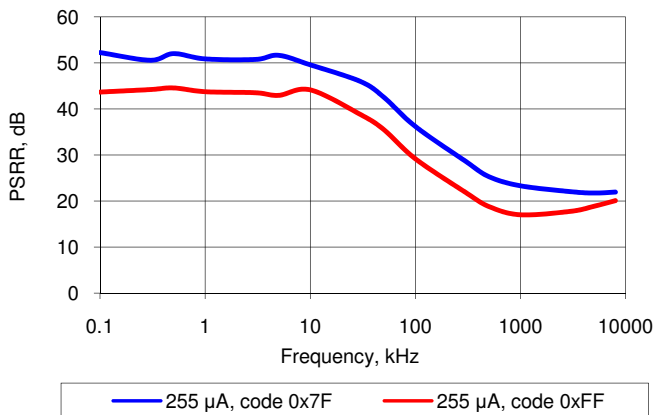


**Figure 11-44. IDAC Current Noise, 255 µA Mode, Source Mode, Fast Mode, V$_{DDA}$ = 5 V**
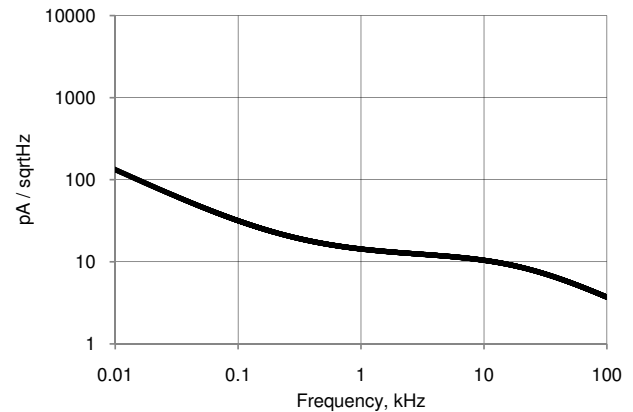


*11.5.6 Voltage Digital to Analog Converter (VDAC)*

See the VDAC component datasheet in PSoC Creator for full electrical specifications and APIs.

Unless otherwise specified, all charts and graphs show typical values.

**Table 11-27. VDAC DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Resolution | | – | 8 | – | bits |
| INL1 | Integral nonlinearity | 1 V scale | – | ±2.1 | ±2.5 | LSB |
| INL4 | Integral nonlinearity[52] | 4 V scale | – | ±2.1 | ±2.5 | LSB |
| DNL1 | Differential nonlinearity | 1 V scale | – | ±0.3 | ±1 | LSB |
| DNL4 | Differential nonlinearity[52] | 4 V scale | – | ±0.3 | ±1 | LSB |
| Rout | Output resistance | 1 V scale | – | 4 | – | kΩ |
| | | 4 V scale | – | 16 | – | kΩ |
| V$_{OUT}$ | Output voltage range, code = 255 | 1 V scale | – | 1.02 | – | V |
| | | 4 V scale, V$_{DDA}$ = 5 V | – | 4.08 | – | V |

**Note**
52. Based on device characterization (Not production tested).
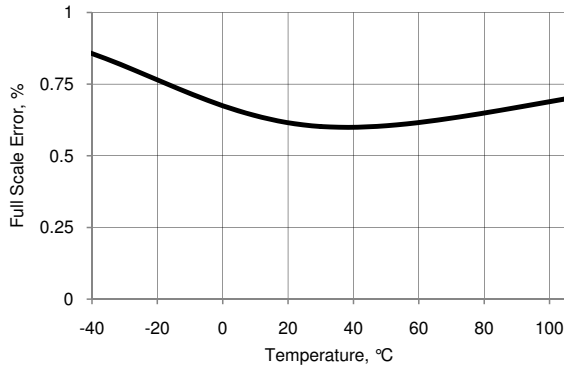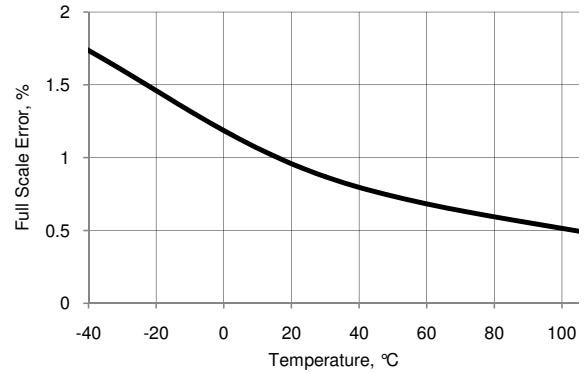
**Table 11-27.  VDAC DC Specifications** (continued)

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Monotonicity | | – | – | Yes | – |
| V<sub>OS</sub> | Zero scale error | | – | 0 | ±0.9 | LSB |
| Eg | Gain error | 1 V scale | – | – | ±2.5 | % |
| | | 4 V scale | – | – | ±2.5 | % |
| TC_Eg | Temperature coefficient, gain error | 1 V scale | – | – | 0.03 | %FSR / °C |
| | | 4 V scale | – | – | 0.03 | %FSR / °C |
| I<sub>DD</sub> | Operating current[53] | Slow mode | – | – | 100 | µA |
| | | Fast mode | – | – | 500 | µA |

**Figure 11-45. VDAC INL vs Input Code, 1 V Mode**
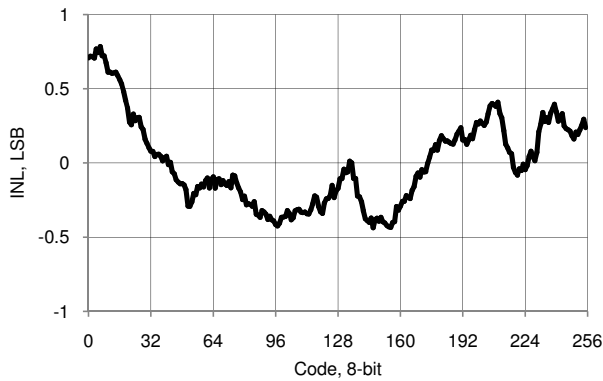


**Figure 11-46. VDAC DNL vs Input Code, 1 V Mode**
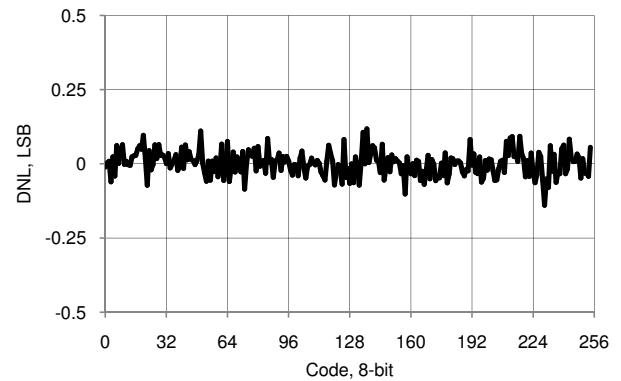


**Figure 11-47. VDAC INL vs Temperature, 1 V Mode**
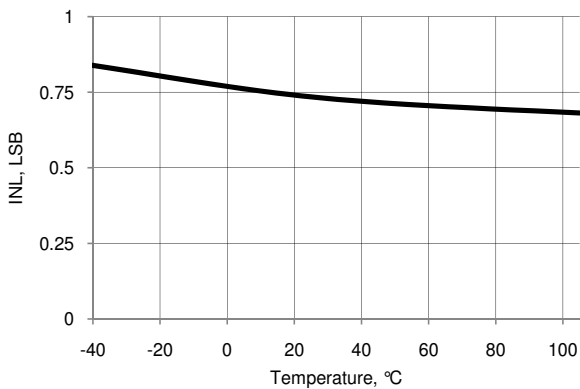


**Figure 11-48. VDAC DNL vs Temperature, 1 V Mode**
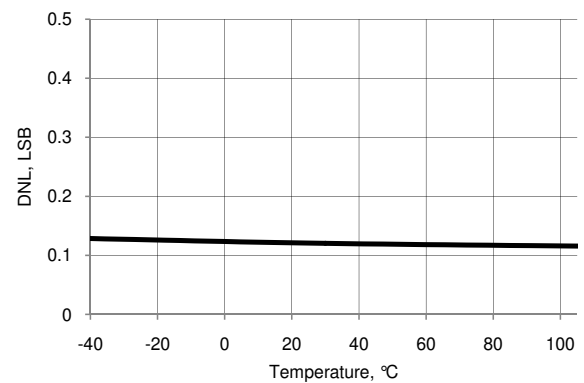


**Note**
53. Based on device characterization (Not production tested).

### Figure 11-49. VDAC Full Scale Error vs Temperature, 1 V Mode



### Figure 11-50. VDAC Full Scale Error vs Temperature, 4 V Mode



### Figure 11-51. VDAC Operating Current vs Temperature, 1V Mode, Slow Mode
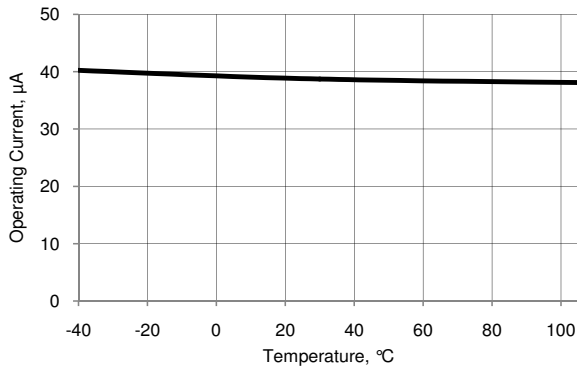


### Figure 11-52. VDAC Operating Current vs Temperature, 1 V Mode, Fast Mode
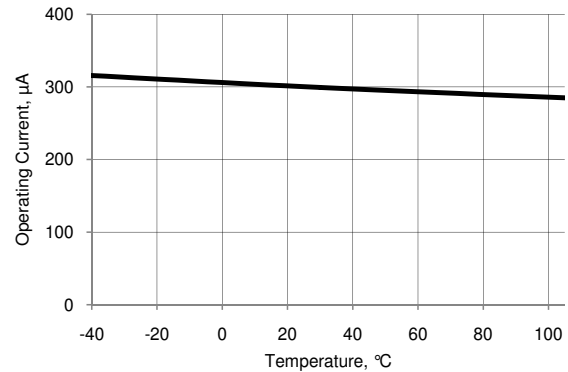


### Table 11-28.  VDAC AC Specifications[54]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{DAC}$ | Update rate | 1 V scale | – | – | 1000 | ksps |
| | | 4 V scale | – | – | 250 | ksps |
| TsettleP | Settling time to 0.1%, step 25% to 75% | 1 V scale, Cload = 15 pF | – | 0.45 | 1 | µs |
| | | 4 V scale, Cload = 15 pF | – | 0.8 | 3.2 | µs |
| TsettleN | Settling time to 0.1%, step 75% to 25% | 1 V scale, Cload = 15 pF | – | 0.45 | 1 | µs |
| | | 4 V scale, Cload = 15 pF | – | 0.7 | 3 | µs |
| | Voltage noise | Range = 1 V, fast mode, $V_{DDA}$ = 5 V, 10 kHz | – | 750 | – | nV/sqrtHz |

**Note**
54. Based on device characterization (Not production tested).

**Figure 11-53. VDAC Step Response, Codes 0x40 - 0xC0, 1 V Mode, Fast Mode, $V_{DDA}$ = 5 V**
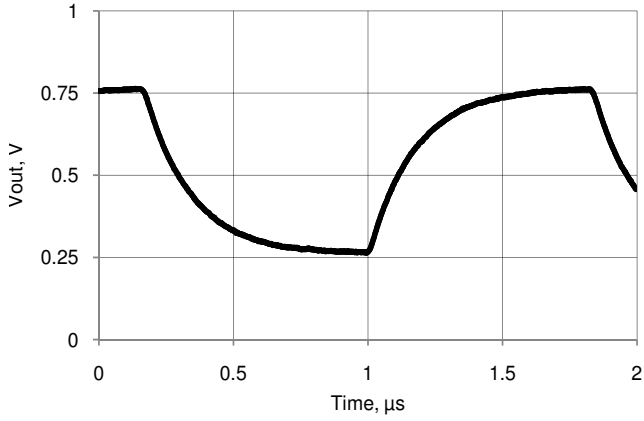


**Figure 11-54. VDAC Glitch Response, Codes 0x7F - 0x80, 1 V Mode, Fast Mode, $V_{DDA}$ = 5 V**
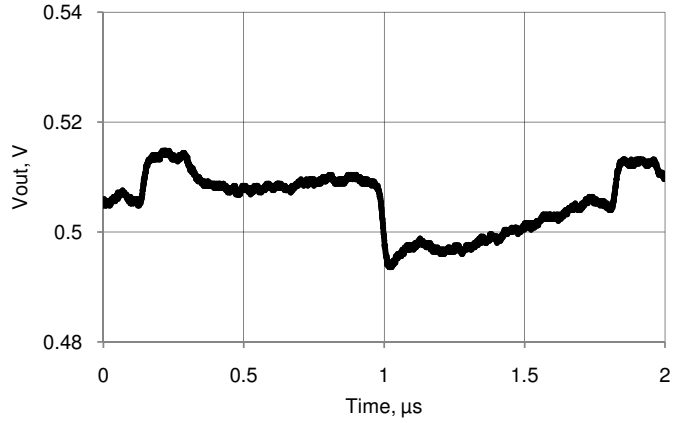


**Figure 11-55. VDAC PSRR vs Frequency**
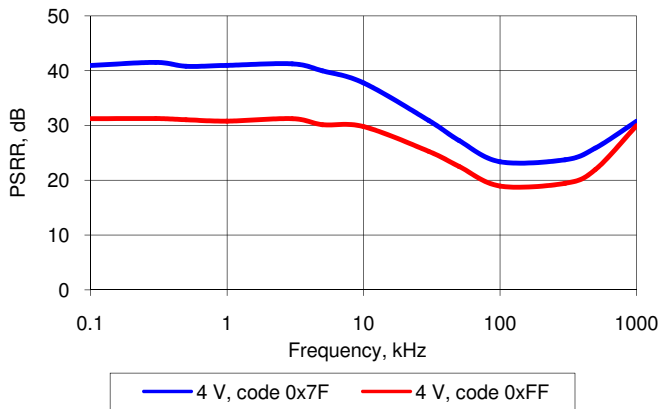


**Figure 11-56. VDAC Voltage Noise, 1 V Mode, Fast Mode, $V_{DDA}$ = 5 V**
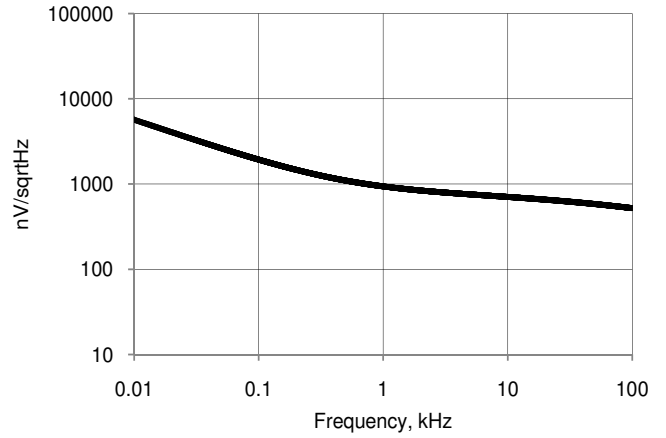
### 11.5.7 Temperature Sensor

**Table 11-29. Temperature Sensor Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Temp sensor accuracy | Range: –40 °C to +85 °C | – | ±5 | – | °C |

### 11.5.8 LCD Direct Drive

**Table 11-30. LCD Direct Drive DC Specifications**[55]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | LCD Block (no glass) | Device sleep mode with wakeup at 400Hz rate to refresh LCD, bus, clock = 3MHz, Vddio = Vdda = 3V, 8 commons, 16 segments, 1/5 duty cycle, 40 Hz frame rate, no glass connected | – | 81 | – | µA |
| $I_{CC\_SEG}$ | Current per segment driver | Strong drive mode | – | 260 | – | µA |
| $V_{BIAS}$ | LCD bias range ($V_{BIAS}$ refers to the main output voltage(V0) of LCD DAC) | $V_{DDA} \geq 3$ V and $V_{DDA} \geq V_{BIAS}$ | 2 | – | 5 | V |
| | LCD bias step size | $V_{DDA} \geq 3$ V and $V_{DDA} \geq V_{BIAS}$ | – | $9.1 \times V_{DDA}$ | – | mV |
| | LCD capacitance per segment/common driver | Drivers may be combined | – | 500 | 5000 | pF |
| | Maximum segment DC offset | Vdda ≥ 3V and Vdda ≥ Vbias | – | – | 20 | mV |
| $I_{OUT}$ | Output drive current per segment driver) | $V_{DDIO}$ = 5.5V, strong drive mode | 355 | – | 710 | µA |

**Table 11-31. LCD Direct Drive AC Specifications**[55]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{LCD}$ | LCD frame rate | | 10 | 50 | 150 | Hz |

**Note**
55. Based on device characterization (Not production tested).

## 11.6 Digital Peripherals

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted.

### 11.6.1 Timer

The following specifications apply to the Timer/Counter/PWM peripheral in timer mode. Timers can also be implemented in UDBs; for more information, see the Timer component datasheet in PSoC Creator.

**Table 11-32. Timer DC Specifications[56]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Block current consumption | 16-bit timer, at listed input clock frequency | – | – | – | µA |
| | 3 MHz | | – | 15 | – | µA |
| | 12 MHz | | – | 60 | – | µA |
| | 48 MHz | | – | 260 | – | µA |
| | 80 MHz | | – | 360 | – | µA |

**Table 11-33. Timer AC Specifications[56]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Operating frequency | | DC | – | 80.01 | MHz |
| | Capture pulse width (Internal)[57] | | 15 | – | – | ns |
| | Capture pulse width (external) | | 30 | – | – | ns |
| | Timer resolution[57] | | 15 | – | – | ns |
| | Enable pulse width[57] | | 15 | – | – | ns |
| | Enable pulse width (external) | | 30 | – | – | ns |
| | Reset pulse width[57] | | 15 | – | – | ns |
| | Reset pulse width (external) | | 30 | – | – | ns |

### 11.6.2 Counter

The following specifications apply to the Timer/Counter/PWM peripheral, in counter mode. Counters can also be implemented in UDBs; for more information, see the Counter component datasheet in PSoC Creator.

**Table 11-34. Counter DC Specifications[56]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Block current consumption | 16-bit counter, at listed input clock frequency | – | – | – | µA |
| | 3 MHz | | – | 15 | – | µA |
| | 12 MHz | | – | 60 | – | µA |
| | 48 MHz | | – | 260 | – | µA |
| | 80 MHz | | – | 360 | – | µA |

**Table 11-35. Counter AC Specifications[56]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Operating frequency | | DC | – | 80.01 | MHz |
| | Capture pulse[57] | | 15 | – | – | ns |
| | Resolution[57] | | 15 | – | – | ns |
| | Pulse width[57] | | 15 | – | – | ns |
| | Pulse width (external) | | 30 | | | ns |

**Notes**
56. Based on device characterization (Not production tested).
57. For correct operation, the minimum Timer/Counter/PWM input pulse width is the period of bus clock.

**Table 11-35.  Counter AC Specifications**[56] (continued)

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Enable pulse width[58] | | 15 | – | – | ns |
| | Enable pulse width (external) | | 30 | – | – | ns |
| | Reset pulse width[58] | | 15 | – | – | ns |
| | Reset pulse width (external) | | 30 | – | – | ns |

### 11.6.3  Pulse Width Modulation

The following specifications apply to the Timer/Counter/PWM peripheral, in PWM mode. PWM components can also be implemented in UDBs; for more information, see the PWM component datasheet in PSoC Creator.

**Table 11-36.  PWM DC Specifications**[59]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Block current consumption | 16-bit PWM, at listed input clock frequency | – | – | – | µA |
| | 3 MHz | | – | 15 | – | µA |
| | 12 MHz | | – | 60 | – | µA |
| | 48 MHz | | – | 260 | – | µA |
| | 80 MHz | | – | 360 | – | µA |

**Table 11-37.  PWM AC Specifications**[59]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Operating frequency | | DC | – | 80.01 | MHz |
| | Pulse width[58] | | 15 | – | – | ns |
| | Pulse width (external) | | 30 | – | – | ns |
| | Kill pulse width[58] | | 15 | – | – | ns |
| | Kill pulse width (external) | | 30 | – | – | ns |
| | Enable pulse width[58] | | 15 | – | – | ns |
| | Enable pulse width (external) | | 30 | – | – | ns |
| | Reset pulse width[58] | | 15 | – | – | ns |
| | Reset pulse width (external) | | 30 | – | – | ns |

### 11.6.4  $I^2C$

**Table 11-38.  Fixed $I^2C$ DC Specifications**[59]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Block current consumption | Enabled, configured for 100 kbps | – | – | 250 | µA |
| | | Enabled, configured for 400 kbps | – | – | 260 | µA |

**Table 11-39.  Fixed $I^2C$ AC Specifications**[59]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Bit rate | | – | – | 1 | Mbps |

**Notes**
58. For correct operation, the minimum Timer/Counter/PWM input pulse width is the period of bus clock.
59. Based on device characterization (Not production tested).

*11.6.5  USB*

**Table 11-40.  USB DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{USB\_5}$ | Device supply ($V_{DDD}$) for USB operation | USB configured, USB regulator enabled | 4.35 | – | 5.25 | V |
| $V_{USB\_3.3}$ | | USB configured, USB regulator bypassed | 3.15 | – | 3.6 | V |
| $V_{USB\_3}$ | | USB configured, USB regulator bypassed[60] | 2.85 | – | 3.6 | V |
| $I_{USB\_Configured}$ | Device supply current in device active mode, bus clock and IMO = 24 MHz | $V_{DDD}$ = 5 V, $F_{CPU}$ = 1.5 MHz | – | 10 | – | mA |
| | | $V_{DDD}$ = 3.3 V, $F_{CPU}$ = 1.5 MHz | – | 8 | – | mA |
| $I_{USB\_Suspended}$ | Device supply current in device sleep mode | $V_{DDD}$ = 5 V, connected to USB host, PICU configured to wake on USB resume signal | – | 0.5 | – | mA |
| | | $V_{DDD}$ = 5 V, disconnected from USB host | – | 0.3 | – | mA |
| | | $V_{DDD}$ = 3.3 V, connected to USB host, PICU configured to wake on USB resume signal | – | 0.5 | – | mA |
| | | $V_{DDD}$ = 3.3 V, disconnected from USB host | – | 0.3 | – | mA |

**Note**
60. Rise/fall time matching (TR) not guaranteed, see Table 11-16 on page 75.

### 11.6.6 Universal Digital Blocks (UDBs)

PSoC Creator provides a library of pre-built and tested standard digital peripherals (UART, SPI, LIN, PRS, CRC, timer, counter, PWM, AND, OR, and so on) that are mapped to the UDB array. See the component datasheets in PSoC Creator for full AC/DC specifications, APIs, and example code.

**Table 11-41.  UDB AC Specifications[61]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Datapath Performance | | | | | | |
| $F_{MAX\_TIMER}$ | Maximum frequency of 16-bit timer in a UDB pair | | – | – | 67.01 | MHz |
| $F_{MAX\_ADDER}$ | Maximum frequency of 16-bit adder in a UDB pair | | – | – | 67.01 | MHz |
| $F_{MAX\_CRC}$ | Maximum frequency of 16-bit CRC/PRS in a UDB pair | | – | – | 67.01 | MHz |
| PLD Performance | | | | | | |
| $F_{MAX\_PLD}$ | Maximum frequency of a two-pass PLD function in a UDB pair | | – | – | 67.01 | MHz |
| Clock to Output Performance | | | | | | |
| $t_{CLK\_OUT}$ | Propagation delay for clock in to data out, see Figure 11-57. | 25 °C, $V_{DDD} \geq 2.7$ V | – | 20 | 25 | ns |
| $t_{CLK\_OUT}$ | Propagation delay for clock in to data out, see Figure 11-57. | Worst-case placement, routing, and pin selection | – | – | 55 | ns |

**Figure 11-57. Clock to Output Performance**
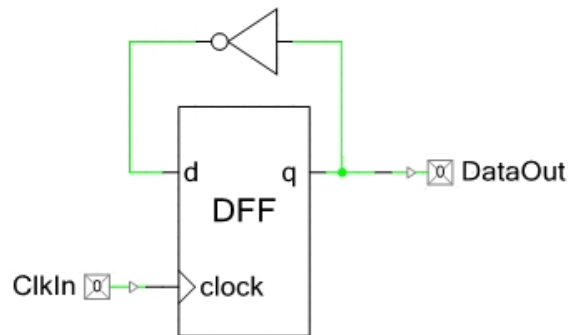


**Note**
61. Based on device characterization (Not production tested).

## 11.7 Memory

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted.

### 11.7.1 Flash

**Table 11-42. Flash DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Erase and program voltage | $V_{DDD}$ pin | 1.71 | – | 5.5 | V |

**Table 11-43. Flash AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{WRITE}$ | Row write time (erase + program) | | – | 15 | 20 | ms |
| $T_{ERASE}$ | Row erase time | | – | 10 | 13 | ms |
| | Row program time | | – | 5 | 7 | ms |
| $T_{BULK}$ | Bulk erase time (256 KB) | | – | – | 140 | ms |
| | Sector erase time (16 KB) | | – | – | 15 | ms |
| $T_{PROG}$ | Total device programming time | No overhead[62] | – | 5 | 7.5 | seconds |
| | Flash data retention time, retention period measured from last erase cycle | Average ambient temp. $T_A$ ≤ 55 °C, 100 K erase/program cycles | 20 | – | – | years |
| | | Average ambient temp. $T_A$ ≤ 85 °C, 10 K erase/program cycles | 10 | – | – | |

### 11.7.2 EEPROM

**Table 11-44. EEPROM DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Erase and program voltage | | 1.71 | – | 5.5 | V |

**Table 11-45. EEPROM AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{WRITE}$ | Single row erase/write cycle time | | – | 10 | 20 | ms |
| | EEPROM data retention time, retention period measured from last erase cycle | Average ambient temp, $T_A$ ≤ 25 °C, 1M erase/program cycles | 20 | – | – | years |
| | | Average ambient temp, $T_A$ ≤ 55 °C, 100 K erase/program cycles | 20 | – | – | |
| | | Average ambient temp. $T_A$ ≤ 85 °C, 10 K erase/program cycles | 10 | – | – | |

**Note**
62. See PSoC 5 Device Programming Specifications for a description of a low-overhead method of programming PSoC 5 flash.

### 11.7.3 Nonvolatile Latches (NVL)

**Table 11-46. NVL DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Erase and program voltage | $V_{DDD}$ pin | 1.71 | – | 5.5 | V |

**Table 11-47. NVL AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | NVL endurance | Programmed at 25 °C | 1K | – | – | program/ erase cycles |
| | | Programmed at 0 °C to 70 °C | 100 | – | – | program/ erase cycles |
| | NVL data retention time | Average ambient temp. $T_A \leq 55$ °C | 20 | – | – | years |
| | | Average ambient temp. $T_A \leq 85$ °C | 10 | – | – | years |

### 11.7.4 SRAM

**Table 11-48. SRAM DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{SRAM}$ | SRAM retention voltage[63] | | 1.2 | – | – | V |

**Table 11-49. SRAM AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{SRAM}$ | SRAM operating frequency | | DC | – | 80.01 | MHz |

**Note**
63. Based on device characterization (Not production tested).

### 11.7.5 External Memory Interface

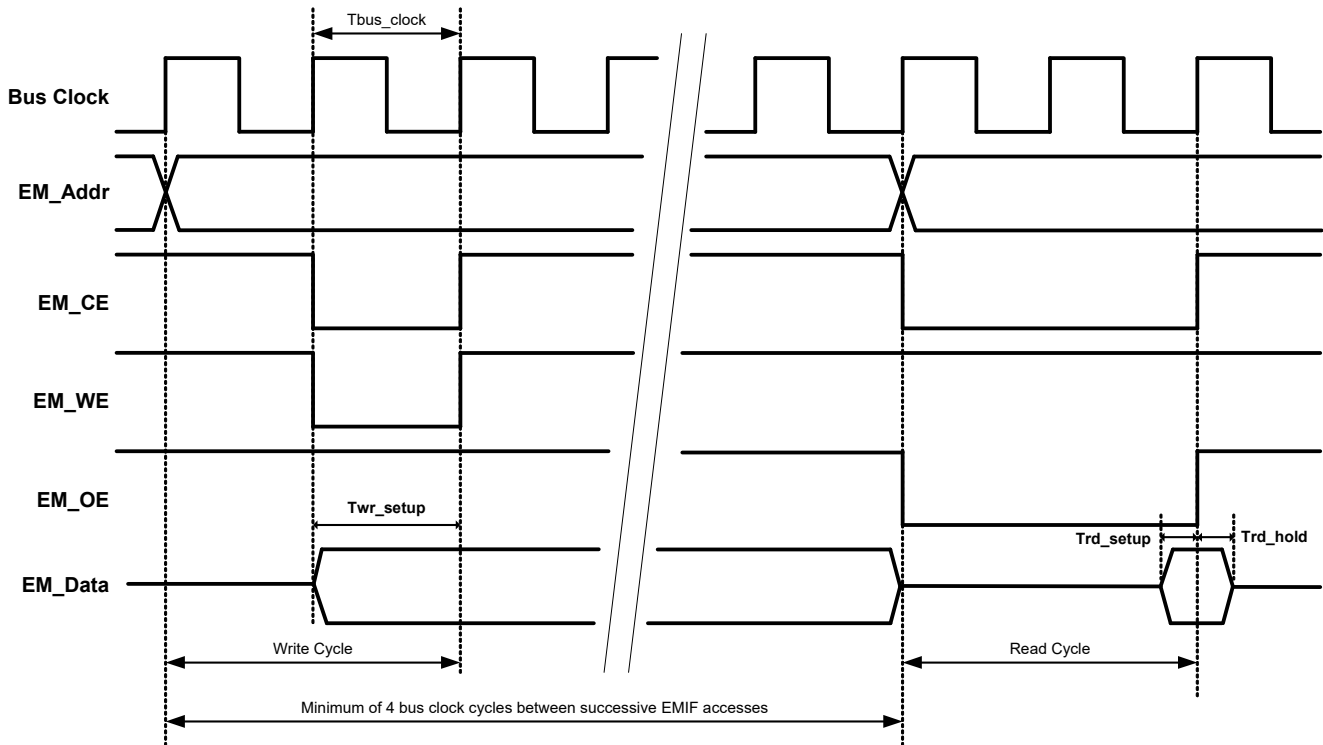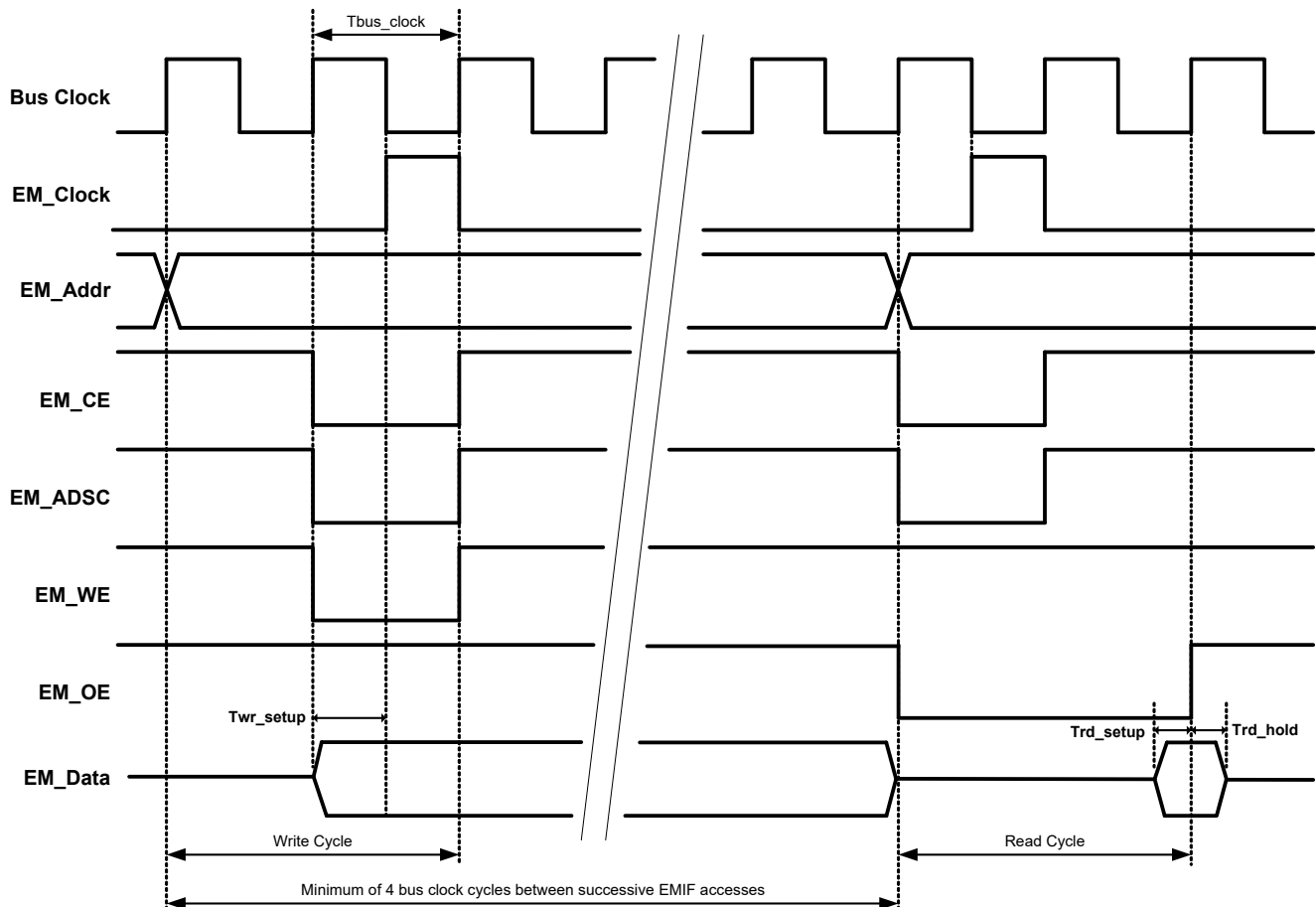**Figure 11-58. Asynchronous Write and Read Cycle Timing, No Wait States**



**Table 11-50. Asynchronous Write and Read Timing Specifications[64]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|-----------|-------------|------------|-----|-----|-----|-------|
| Fbus_clock | Bus clock frequency[65] | | – | – | 33 | MHz |
| Tbus_clock | Bus clock period[66] | | 30.3 | – | – | ns |
| Twr_Setup | Time from EM_data valid to rising edge of EM_WE and EM_CE | | Tbus_clock – 10 | – | – | ns |
| Trd_setup | Time that EM_data must be valid before rising edge of EM_OE | | 5 | – | – | ns |
| Trd_hold | Time that EM_data must be valid after rising edge of EM_OE | | 5 | – | – | ns |

**Notes**
64. Based on device characterization (Not production tested).
65. EMIF signal timings are limited by GPIO frequency limitations. See "GPIO" section on page 68.
66. EMIF output signals are generally synchronized to bus clock, so EMIF signal timings are dependent on bus clock frequency.

### Figure 11-59. Synchronous Write and Read Cycle Timing, No Wait States



### Table 11-51. Synchronous Write and Read Timing Specifications[67]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Fbus_clock | Bus clock frequency[68] | | – | – | 33 | MHz |
| Tbus_clock | Bus clock period[69] | | 30.3 | – | – | ns |
| Twr_Setup | Time from EM_data valid to rising edge of EM_Clock | | Tbus_clock – 10 | – | – | ns |
| Trd_setup | Time that EM_data must be valid before rising edge of EM_OE | | 5 | – | – | ns |
| Trd_hold | Time that EM_data must be valid after rising edge of EM_OE | | 5 | – | – | ns |

**Notes**
67. Based on device characterization (Not production tested).
68. EMIF signal timings are limited by GPIO frequency limitations. See "GPIO" section on page 68.
69. EMIF output signals are generally synchronized to bus clock, so EMIF signal timings are dependent on bus clock frequency.

## 11.8 PSoC System Resources

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted.

### 11.8.1 POR with Brown Out

For brown out detect in regulated mode, $V_{DDD}$ and $V_{DDA}$ must be ≥ 2.0 V. Brown out detect is not available in externally regulated mode.

**Table 11-52. Precise Low-Voltage Reset (PRES) with Brown Out DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|-----------|-------------|------------|-----|-----|-----|-------|
| PRESR | Rising trip voltage | Factory trim | 1.64 | – | 1.68 | V |
| PRESF | Falling trip voltage | | 1.62 | – | 1.66 | V |

**Table 11-53. Power On Reset (POR) with Brown Out AC Specifications[70]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|-----------|-------------|------------|-----|-----|-----|-------|
| PRES_TR[71] | Response time | | – | – | 0.5 | µs |
| | $V_{DDD}$/$V_{DDA}$ droop rate | Sleep mode | – | 5 | – | V/sec |

### 11.8.2 Voltage Monitors

**Table 11-54. Voltage Monitors DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|-----------|-------------|------------|-----|-----|-----|-------|
| LVI | Trip voltage | | | | | |
| | LVI_A/D_SEL[3:0] = 0000b | | 1.68 | 1.73 | 1.77 | V |
| | LVI_A/D_SEL[3:0] = 0001b | | 1.89 | 1.95 | 2.01 | V |
| | LVI_A/D_SEL[3:0] = 0010b | | 2.14 | 2.20 | 2.27 | V |
| | LVI_A/D_SEL[3:0] = 0011b | | 2.38 | 2.45 | 2.53 | V |
| | LVI_A/D_SEL[3:0] = 0100b | | 2.62 | 2.71 | 2.79 | V |
| | LVI_A/D_SEL[3:0] = 0101b | | 2.87 | 2.95 | 3.04 | V |
| | LVI_A/D_SEL[3:0] = 0110b | | 3.11 | 3.21 | 3.31 | V |
| | LVI_A/D_SEL[3:0] = 0111b | | 3.35 | 3.46 | 3.56 | V |
| | LVI_A/D_SEL[3:0] = 1000b | | 3.59 | 3.70 | 3.81 | V |
| | LVI_A/D_SEL[3:0] = 1001b | | 3.84 | 3.95 | 4.07 | V |
| | LVI_A/D_SEL[3:0] = 1010b | | 4.08 | 4.20 | 4.33 | V |
| | LVI_A/D_SEL[3:0] = 1011b | | 4.32 | 4.45 | 4.59 | V |
| | LVI_A/D_SEL[3:0] = 1100b | | 4.56 | 4.70 | 4.84 | V |
| | LVI_A/D_SEL[3:0] = 1101b | | 4.83 | 4.98 | 5.13 | V |
| | LVI_A/D_SEL[3:0] = 1110b | | 5.05 | 5.21 | 5.37 | V |
| | LVI_A/D_SEL[3:0] = 1111b | | 5.30 | 5.47 | 5.63 | V |
| HVI | Trip voltage | | 5.57 | 5.75 | 5.92 | V |

**Table 11-55. Voltage Monitors AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|-----------|-------------|------------|-----|-----|-----|-------|
| LVI_tr[71] | Response time | | – | – | 1 | µs |

**Notes**
70. Based on device characterization (Not production tested).
71. This value is calculated, not measured.

### 11.8.3 Interrupt Controller

**Table 11-56. Interrupt Controller AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Delay from interrupt signal input to ISR code execution from main line code[72] | | – | – | 12 | Tcy CPU |
| | Delay from interrupt signal input to ISR code execution from ISR code (tail-chaining)[72] | | – | – | 6 | Tcy CPU |

### 11.8.4 JTAG Interface

**Figure 11-60. JTAG Interface Timing**
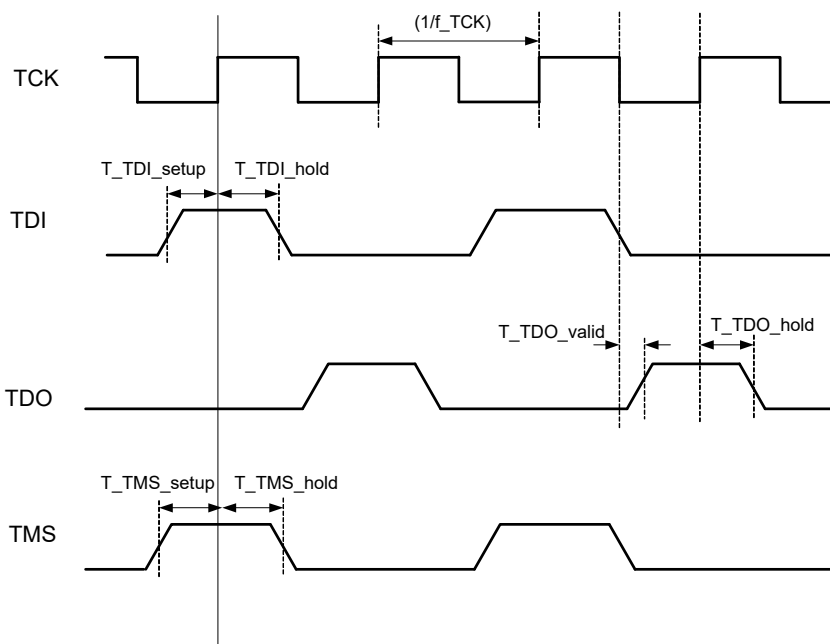


**Table 11-57. JTAG Interface AC Specifications[73]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| f_TCK | TCK frequency | 3.3 V ≤ V$_{DDD}$ ≤ 5 V | – | – | 12[74] | MHz |
| | | 1.71 V ≤ V$_{DDD}$ < 3.3 V | – | – | 7[74] | MHz |
| T_TDI_setup | TDI setup before TCK high | | (T/10) – 5 | – | – | ns |
| T_TMS_setup | TMS setup before TCK high | | T/4 | – | – | |
| T_TDI_hold | TDI, TMS hold after TCK high | T = 1/f_TCK max | T/4 | – | – | |
| T_TDO_valid | TCK low to TDO valid | T = 1/f_TCK max | – | – | 2T/5 | |
| T_TDO_hold | TDO hold after TCK high | T = 1/f_TCK max | T/4 | – | – | |
| T_nTRST | Minimum nTRST pulse width | f_TCK = 2 MHz | 8 | – | – | ns |

**Notes**
72. Arm Cortex-M3 NVIC spec. Visit www.arm.com for detailed documentation about the Cortex-M3 CPU.
73. Based on device characterization (Not production tested).
74. f_TCK must also be no more than 1/3 CPU clock frequency.

## 11.8.5 SWD Interface

**Figure 11-61. SWD Interface Timing**
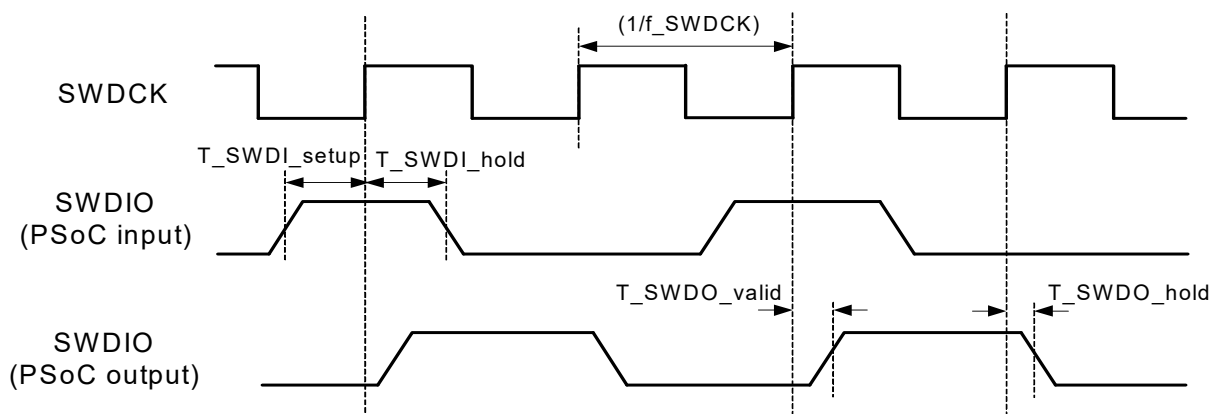


**Table 11-58. SWD Interface AC Specifications[75]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| f_SWDCK | SWDCLK frequency | 3.3 V ≤ $V_{DDD}$ ≤ 5 V | – | – | 12[76] | MHz |
| | | 1.71 V ≤ $V_{DDD}$ < 3.3 V | – | – | 7[76] | MHz |
| | | 1.71 V ≤ $V_{DDD}$ < 3.3 V, SWD over USBIO pins | – | – | 5.5[76] | MHz |
| T_SWDI_setup | SWDIO input setup before SWDCK high | T = 1/f_SWDCK max | T/4 | – | – | |
| T_SWDI_hold | SWDIO input hold after SWDCK high | T = 1/f_SWDCK max | T/4 | – | – | |
| T_SWDO_valid | SWDCK high to SWDIO output | T = 1/f_SWDCK max | – | – | T/2 | |
| T_SWDO_hold | SWDIO output hold after SWDCK high | T = 1/f_SWDCK max | 1 | – | – | ns |

## 11.8.6 TPIU Interface

**Table 11-59. TPIU Interface AC Specifications[75]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | TRACEPORT (TRACECLK) frequency | | – | – | 33[77] | MHz |
| | SWV bit rate | | – | – | 33[77] | Mbit |

**Notes**
75. Based on device characterization (Not production tested).
76. f_SWDCK must also be no more than 1/3 CPU clock frequency.
77. TRACEPORT signal frequency and bit rate are limited by GPIO output frequency, see Table 11-9 on page 69.

## 11.9  Clocking

Specifications are valid for –40 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted. Unless otherwise specified, all charts and graphs show typical values.

### 11.9.1  Internal Main Oscillator

**Table 11-60.  IMO DC Specifications[78]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Icc_imo | Supply current | | | | | |
| | 74.7 MHz | | – | – | 730 | µA |
| | 62.6 MHz | | – | – | 600 | µA |
| | 48 MHz | | – | – | 500 | µA |
| | 24 MHz – USB mode | With oscillator locking to USB bus | – | – | 500 | µA |
| | 24 MHz – non-USB mode | | – | – | 300 | µA |
| | 12 MHz | | – | – | 200 | µA |
| | 6 MHz | | – | – | 180 | µA |
| | 3 MHz | | – | – | 150 | µA |

**Figure 11-62. IMO Current vs. Frequency**
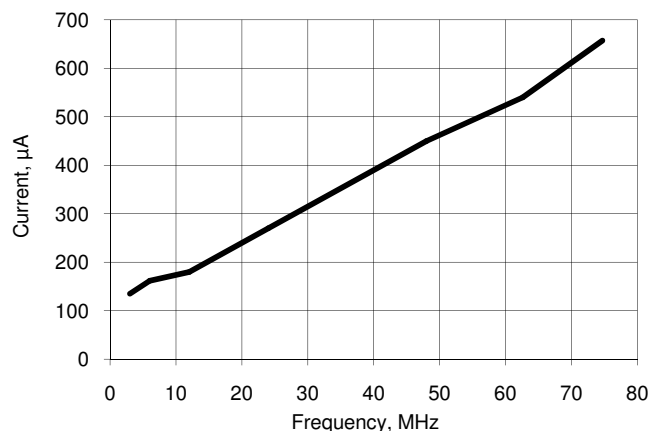


**Table 11-61.  IMO AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{IMO}$ | IMO frequency stability (with factory trim) | | | | | |
| | 74.7 MHz | | –7 | – | 7 | % |
| | 62.6 MHz | | –7 | – | 7 | % |
| | 48 MHz | | –5 | – | 5 | % |
| | 24 MHz – non-USB mode | | –4 | – | 4 | % |
| | 24 MHz – USB mode | With oscillator locking to USB bus | –0.25 | – | 0.25 | % |
| | 12 MHz | | –3 | – | 3 | % |
| | 6 MHz | | –2 | – | 2 | % |
| | 3 MHz | | –2 | – | 2 | % |
| Tstart_imo | Startup time[78] | From enable (during normal system operation) | – | – | 13 | µs |

**Note**
78. Based on device characterization (Not production tested).

**Table 11-61. IMO AC Specifications** (continued)

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Jp-p | Jitter (peak to peak)[79] | | | | | |
| | F = 24 MHz | | – | 0.9 | – | ns |
| | F = 3 MHz | | – | 1.6 | – | ns |
| Jperiod | Jitter (long term)[79] | | | | | |
| | F = 24 MHz | | – | 0.9 | – | ns |
| | F = 3 MHz | | – | 12 | – | ns |

**Figure 11-63. IMO Frequency Variation vs. Temperature**
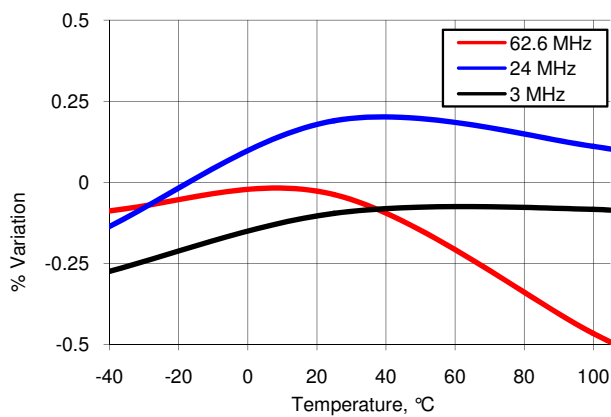


**Figure 11-64. IMO Frequency Variation vs. V$_{CC}$**
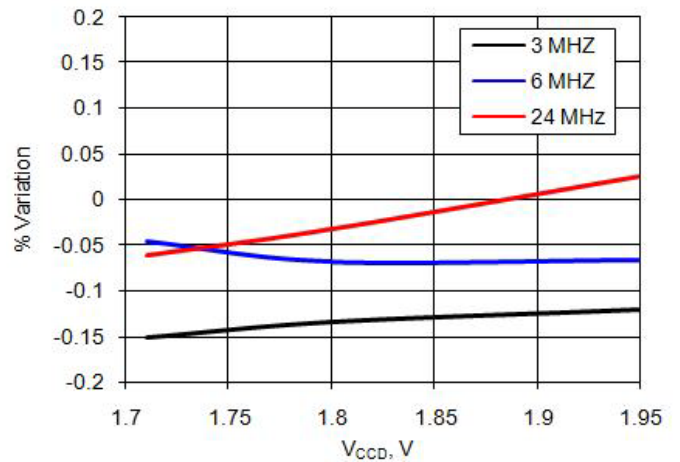


**Note**
79. Based on device characterization (Not production tested).

### 11.9.2 Internal Low-Speed Oscillator

**Table 11-62. ILO DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Operating current[80] | $F_{OUT}$ = 1 kHz | – | – | 1.7 | µA |
| | | $F_{OUT}$ = 33 kHz | – | – | 2.6 | µA |
| | | $F_{OUT}$ = 100 kHz | – | – | 2.6 | µA |
| | Leakage current[80] | Power down mode | – | – | 15 | nA |

**Table 11-63. ILO AC Specifications[81]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Tstart_ilo | Startup time, all frequencies | Turbo mode | – | – | 2 | ms |
| $F_{ILO}$ | ILO frequencies | | | | | |
| | 100 kHz | | 45 | 100 | 200 | kHz |
| | 1 kHz | | 0.5 | 1 | 2 | kHz |

**Figure 11-65. ILO Frequency Variation vs. Temperature**
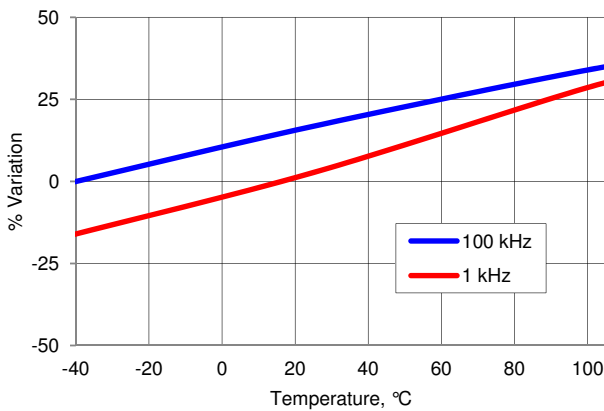


**Figure 11-66. ILO Frequency Variation vs. $V_{DD}$**
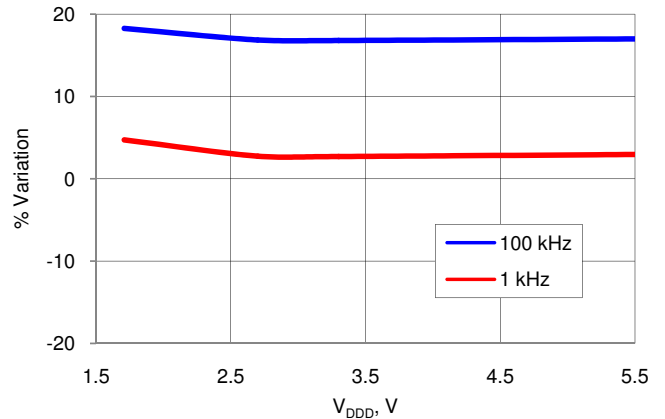


**Notes**
80. This value is calculated, not measured.
81. Based on device characterization (Not production tested).

### 11.9.3  MHz External Crystal Oscillator

For more information on crystal or ceramic resonator selection for the MHzECO, refer to application note AN54439: PSoC 3 and PSoC 5 External Oscillators..

**Table 11-64.  MHzECO DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Operating current[82] | 13.56 MHz crystal | – | 3.8 | – | mA |

**Table 11-65.  MHzECO AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| F | Crystal frequency range | | 4 | – | 25 | MHz |

### 11.9.4  kHz External Crystal Oscillator

**Table 11-66.  kHzECO DC Specifications[82]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Operating current | Low power mode; CL = 6 pF | – | 0.25 | 1.0 | µA |
| DL | Drive level | | – | – | 1 | µW |

**Table 11-67.  kHzECO AC Specifications[82]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| F | Frequency | | – | 32.768 | – | kHz |
| $T_{ON}$ | Startup time | High power mode | – | 1 | – | s |

### 11.9.5  External Clock Reference

**Table 11-68.  External Clock Reference AC Specifications[82]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | External frequency range | | 0 | – | 33 | MHz |
| | Input duty cycle range | Measured at $V_{DDIO}$/2 | 30 | 50 | 70 | % |
| | Input edge rate | $V_{IL}$ to $V_{IH}$ | 0.5 | – | – | V/ns |

### 11.9.6  Phase-Locked Loop

**Table 11-69.  PLL DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | PLL operating current | In = 3 MHz, Out = 80 MHz | – | 650 | – | µA |
| | | In = 3 MHz, Out = 24 MHz | – | 200 | – | µA |
| | | In = 3 MHz, Out = 67 MHz | – | 400 | – | µA |

**Table 11-70.  PLL AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Fpllin | PLL input frequency[83] | | 1 | – | 48 | MHz |
| | PLL intermediate frequency[84] | Output of prescaler | 1 | – | 3 | MHz |
| Fpllout | PLL output frequency[83] | | 24 | – | 80 | MHz |
| | Lock time at startup | | – | – | 250 | µs |
| Jperiod-rms | Jitter (rms)[82] | | – | – | 250 | ps |

**Notes**
82. Based on device characterization (Not production tested).
83. This specification is guaranteed by testing the PLL across the specified range using the IMO as the source for the PLL.
84. PLL input divider, Q, must be set so that the input frequency is divided down to the intermediate frequency range. Value for Q ranges from 1 to 16.

## 12. Ordering Information

In addition to the features listed in Table 12-1, every CY8C52LP device includes: up to 256 KB flash, 64 KB SRAM, 2 KB EEPROM, a precision on-chip voltage reference, precision oscillators, flash, ECC, DMA, a fixed function I$^2$C, JTAG/SWD programming and debug, external memory interface, boost, and more. In addition to these features, the flexible UDBs and analog subsection support a wide range of peripherals. To assist you in selecting the ideal part, PSoC Creator makes a part recommendation after you choose the components required by your application. All CY8C52LP derivatives incorporate device and flash security in user-selectable security levels; see the TRM for details.

**Table 12-1. CY8C52LP Family with Arm Cortex-M3 CPU**

| Part Number | MCU Core | | | | Analog | | | | | | | | Digital | | | I/O[86] | | | | Package | JTAG ID[87] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CPU Speed (MHz) | Flash (KB) | SRAM (KB) | EEPROM (KB) | LCD Segment Drive | ADC | DAC | Comparators | SC/CT Analog Blocks | Opamps | DFB | CapSense | UDBs[85] | 16-bit Timer/PWM | FS USB | Total I/O | GPIO | SIO | USBIO | | |
| CY8C5268LTI-LP030 | 67 | 256 | 64 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | ✔ | 48 | 38 | 8 | 2 | 68-QFN | 0x2E11E069 |
| CY8C5268AXI-LP047 | 67 | 256 | 64 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | ✔ | 72 | 62 | 8 | 2 | 100-TQFP | 0x2E12F069 |
| CY8C5267AXI-LP051 | 67 | 128 | 32 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | ✔ | 72 | 62 | 8 | 2 | 100-TQFP | 0x2E133069 |
| CY8C5267LTI-LP089 | 67 | 128 | 32 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | ✔ | 48 | 38 | 8 | 2 | 68-QFN | 0x2E159069 |
| CY8C5266LTI-LP029 | 67 | 64 | 16 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 20 | 4 | ✔ | 48 | 38 | 8 | 2 | 68-QFN | 0x2E11D069 |
| CY8C5266AXI-LP033 | 67 | 64 | 16 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 20 | 4 | ✔ | 72 | 62 | 8 | 2 | 100-TQFP | 0x2E121069 |
| CY8C5266AXI-LP132 | 67 | 64 | 16 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 20 | 4 | – | 70 | 62 | 8 | 0 | 100-TQFP | 0x2E184069 |
| CY8C5266LTI-LP150 | 67 | 64 | 16 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 20 | 4 | – | 46 | 38 | 8 | 0 | 68-QFN | 0x2E196069 |
| CY8C5266FNI-LP205T | 67 | 64 | 16 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | – | 70 | 62 | 8 | 0 | 99-WLCSP | 0x2E1CD069 |
| CY8C5265LTI-LP050 | 67 | 32 | 8 | 2 | ✔ | 1x12-bit SAR | 1 | 0 | 0 | 0 | – | ✔ | 20 | 4 | ✔ | 48 | 38 | 8 | 2 | 68-QFN | 0x2E132069 |
| CY8C5265AXI-LP056 | 67 | 32 | 8 | 2 | ✔ | 1x12-bit SAR | 1 | 0 | 0 | 0 | – | ✔ | 20 | 4 | ✔ | 72 | 62 | 8 | 2 | 100-TQFP | 0x2E138069 |
| CY8C5265LTI-LP058 | 67 | 32 | 8 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 20 | 4 | ✔ | 48 | 38 | 8 | 2 | 68-QFN | 0x2E13A069 |
| CY8C5265AXI-LP082 | 67 | 32 | 8 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 20 | 4 | ✔ | 72 | 62 | 8 | 2 | 100-TQFP | 0x2E152069 |
| CY8C5287AXI-LP095[88] | 80 | 256 | 64 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | ✔ | 72 | 62 | 8 | 2 | 100-TQFP | 0x2E15F069 |
| CY8C5288LTI-LP090 | 80 | 256 | 64 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | ✔ | 48 | 38 | 8 | 2 | 68-QFN | 0x2E15A069 |
| CY8C5288FNI-LP213 | 80 | 256 | 64 | 2 | ✔ | 1x12-bit SAR | 1 | 2 | 0 | 0 | – | ✔ | 24 | 4 | ✔ | 72 | 62 | 8 | 2 | 99-WLCSP | 0x2E1D5069 |

**Notes**
85. UDBs support a wide variety of functionality including SPI, LIN, UART, timer, counter, PWM, PRS, and others. Individual functions may use a fraction of a UDB or multiple UDBs. Multiple functions can share a single UDB. See Example Peripherals on page 39 for more information on how UDBs can be used.
86. The I/O Count includes all types of digital I/O: GPIO, SIO, and the two USB I/O. See I/O System and Routing on page 32 for details on the functionality of each of these types of I/O.
87. The JTAG ID has three major fields. The most significant nibble (left digit) is the version, followed by a 2 byte part number and a 3 nibble manufacturer ID.
88. This part varies from the numbering conventions described in Table 12-1. It has 256 KB flash, not 128 KB.

## 12.1 Part Numbering Conventions

PSoC 5LP devices follow the part numbering convention described here. All fields are single character alphanumeric (0, 1, 2, …, 9, A, B, …, Z) unless stated otherwise.

CY8Cabcdefg-LPxxx

- **a: Architecture**
  - □ 3: PSoC 3
  - □ 5: PSoC 5
- **b: Family group within architecture**
  - □ 2: CY8C52LP family
  - □ 4: CY8C54LP family
  - □ 6: CY8C56LP family
  - □ 8: CY8C58LP family
- **c: Speed grade**
  - □ 6: 67 MHz
  - □ 8: 80 MHz
- **d: Flash capacity**
  - □ 5: 32 KB
  - □ 6: 64 KB
  - □ 7: 128 KB
  - □ 8: 256 KB

- **ef: Package code**
  - □ Two character alphanumeric
  - □ AX: TQFP
  - □ LT: QFN
  - □ PV: SSOP
  - □ FN: CSP
- **g: Temperature range**
  - □ C: commercial
  - □ I: industrial
  - □ A: automotive
- **xxx: Peripheral set**
  - □ Three character numeric
  - □ No meaning is associated with these three characters

### Examples

CY8C  5  2  8  8 AX/PV I - LPx  x  x

| | |
|---|---|
| | Cypress Prefix |
| **5: PSoC 5** | Architecture |
| **2: CY8C52 Family** | Family Group within Architecture |
| **8: 80 MHz** | Speed Grade |
| **8: 256 KB** | Flash Capacity |
| **AX: TQFP, PV:SSOP** | Package Code |
| **I: Industrial** | Temperature Range |
| | Peripheral Set |
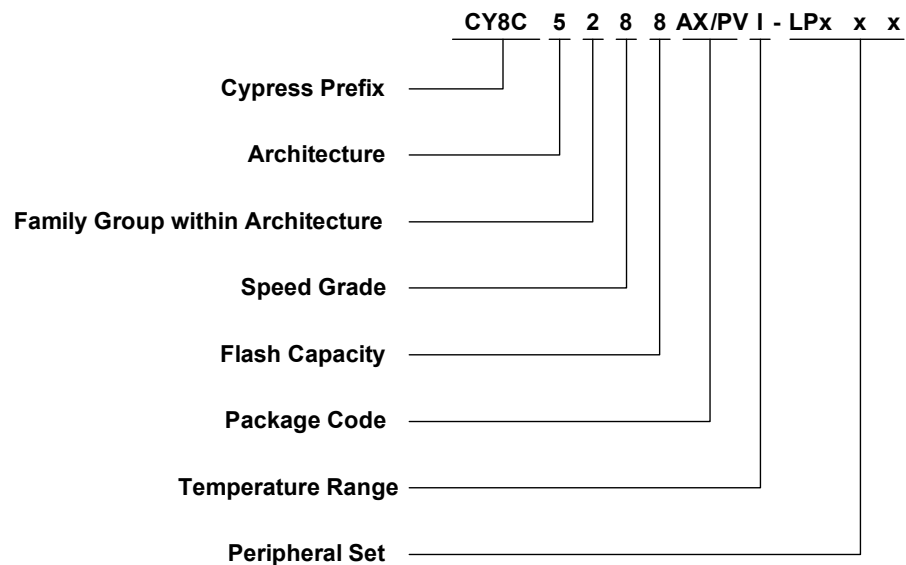
Tape and reel versions of these devices are available and are marked with a "T" at the end of the part number.

All devices in the PSoC 5LP CY8C52LP family comply to RoHS-6 specifications, demonstrating the commitment by Cypress to lead-free products. Lead (Pb) is an alloying element in solders that has resulted in environmental concerns due to potential toxicity. Cypress uses nickel-palladium-gold (NiPdAu) technology for the majority of leadframe-based packages.

A high level review of the Cypress Pb-free position is available on our website. Specific package information is also available. Package Material Declaration Datasheets (PMDDs) identify all substances contained within Cypress packages. PMDDs also confirm the absence of many banned substances. The information in the PMDDs will help Cypress customers plan for recycling or other "end of life" requirements.

## 13. Packaging

**Table 13-1. Package Characteristics**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_A$ | Operating ambient temperature | | –40 | 25 | 85 | °C |
| $T_J$ | Operating junction temperature | | –40 | – | 100 | °C |
| $T_{JA}$ | Package $\theta_{JA}$ (68-pin QFN) | | – | 15 | – | °C/Watt |
| $T_{JA}$ | Package $\theta_{JA}$ (100-pin TQFP) | | – | 34 | – | °C/Watt |
| $T_{JC}$ | Package $\theta_{JC}$ (68-pin QFN) | | – | 13 | – | °C/Watt |
| $T_{JC}$ | Package $\theta_{JC}$ (100-pin TQFP) | | – | 10 | - | °C/Watt |
| $T_A$ | Operating ambient temperature | For CSP parts | –40 | 25 | 85 | °C |
| $T_J$ | Operating junction temperature | For CSP parts | –40 | – | 100 | °C |
| $T_{JA}$ | Package $\theta_{JA}$ (99-ball CSP) | | | 16.5 | | °C/Watt |
| $T_{Jc}$ | Package $\theta_{JC}$ (99-ball CSP) | | – | 0.1 | – | °C/Watt |

**Table 13-2. Solder Reflow Peak Temperature**

| Package | Maximum Peak Temperature | Maximum Time at Peak Temperature |
|---|---|---|
| 68-pin QFN | 260 °C | 30 seconds |
| 100-pin TQFP | 260 °C | 30 seconds |
| 99-pin CSP | 255 °C | 30 seconds |

**Table 13-3. Package Moisture Sensitivity Level (MSL), IPC/JEDEC J-STD-2**

| Package | MSL |
|---|---|
| 68-pin QFN | MSL 3 |
| 100-pin TQFP | MSL 3 |
| 99-pin CSP | MSL1 |

## Figure 13-1. 68-pin QFN 8 × 8 with 0.4 mm Pitch Package Outline (Sawn Version)
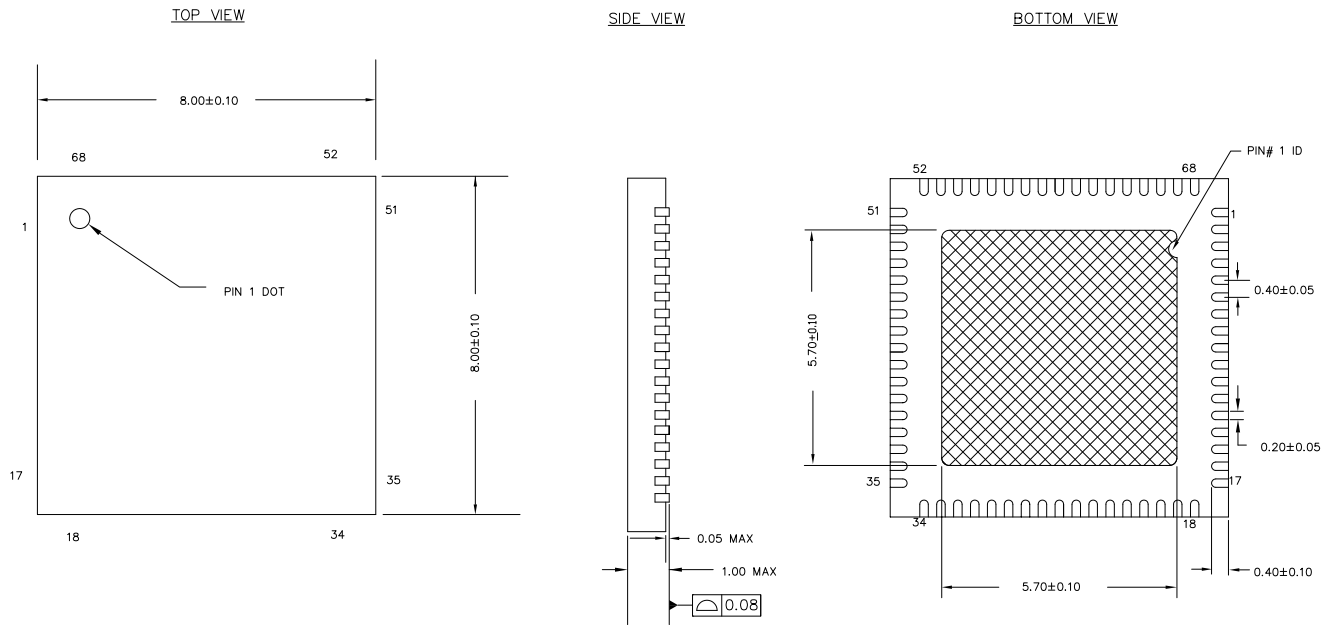


NOTES:

1. ⊠ HATCH AREA IS SOLDERABLE EXPOSED METAL.

2. REFERENCE JEDEC#: MO-220

3. PACKAGE WEIGHT: 17 ± 2mg

4. ALL DIMENSIONS ARE IN MILLIMETERS

001-09618 *E

## Figure 13-2. 100-pin TQFP (14 × 14 × 1.4 mm) Package Outline
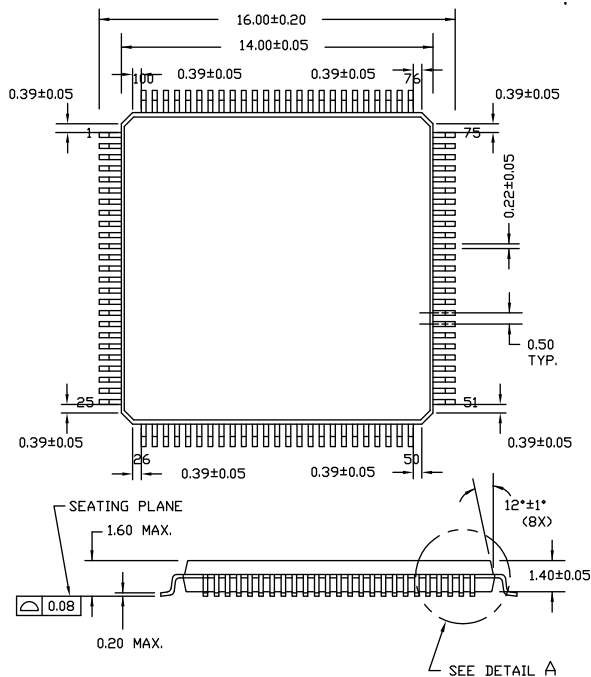


NOTE:

1. JEDEC STD REF MS-026

2. BODY LENGTH DIMENSION DOES NOT INCLUDE MOLD PROTRUSION/END FLASH
MOLD PROTRUSION/END FLASH SHALL NOT EXCEED 0.0098 in (0.25 mm) PER SIDE
BODY LENGTH DIMENSIONS ARE MAX PLASTIC BODY SIZE INCLUDING MOLD MISMATCH

3. DIMENSIONS IN MILLIMETERS

NOTE: PKG. CAN HAVE

TOP LEFT CORNER CHAMFER    OR    4 CORNERS CHAMFER
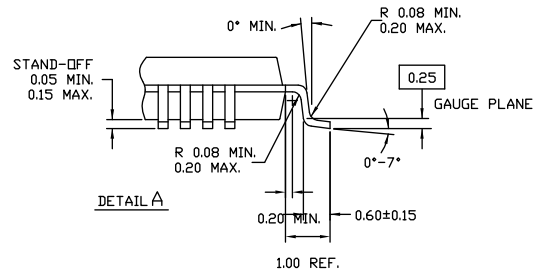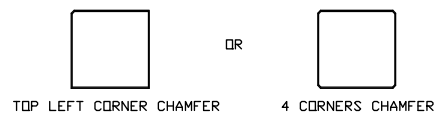
51-85048 *K

**Figure 13-3. WLCSP Package (5.192 × 5.940 × 0.6 mm)**
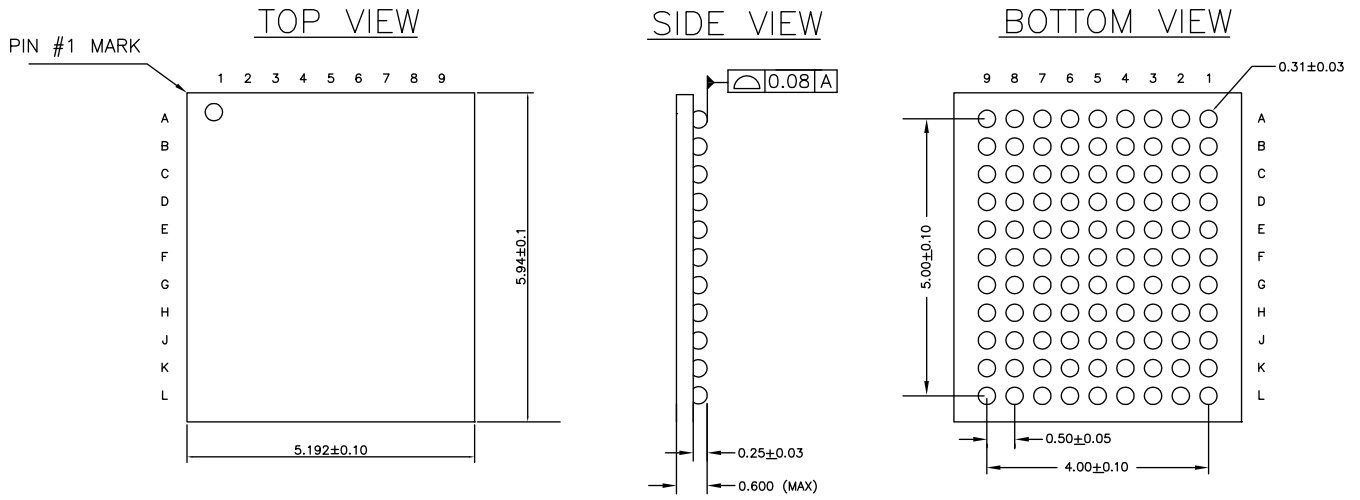


NOTES:
1. REFERENCE JEDEC Publication 95: Design Guide 4.18
2. ALL DIMENSIONS ARE IN MILLIMETERS

001-88034 *B

# 14. Acronyms

**Table 14-1. Acronyms Used in this Document**

| Acronym | Description |
|---|---|
| abus | analog local bus |
| ADC | analog-to-digital converter |
| AG | analog global |
| AHB | AMBA (advanced microcontroller bus architecture) high-performance bus, an Arm data transfer bus |
| ALU | arithmetic logic unit |
| AMUXBUS | analog multiplexer bus |
| API | application programming interface |
| APSR | application program status register |
| Arm® | advanced RISC machine, a CPU architecture |
| ATM | automatic thump mode |
| BW | bandwidth |
| CMRR | common-mode rejection ratio |
| CPU | central processing unit |
| CRC | cyclic redundancy check, an error-checking protocol |
| DAC | digital-to-analog converter, see also IDAC, VDAC |
| DFB | digital filter block |
| DIO | digital input/output, GPIO with only digital capabilities, no analog. See GPIO. |
| DMA | direct memory access, see also TD |
| DNL | differential nonlinearity, see also INL |
| DNU | do not use |
| DR | port write data registers |
| DSI | digital system interconnect |
| DWT | data watchpoint and trace |
| ECC | error correcting code |
| ECO | external crystal oscillator |
| EEPROM | electrically erasable programmable read-only memory |
| EMI | electromagnetic interference |
| EMIF | external memory interface |
| EOC | end of conversion |
| EOF | end of frame |
| EPSR | execution program status register |
| ESD | electrostatic discharge |
| ETM | embedded trace macrocell |
| FIR | finite impulse response, see also IIR |
| FPB | flash patch and breakpoint |

**Table 14-1. Acronyms Used in this Document** (continued)

| Acronym | Description |
|---|---|
| FS | full-speed |
| GPIO | general-purpose input/output, applies to a PSoC pin |
| HVI | high-voltage interrupt, see also LVI, LVD |
| IC | integrated circuit |
| IDAC | current DAC, see also DAC, VDAC |
| IDE | integrated development environment |
| I²C, or IIC | Inter-Integrated Circuit, a communications protocol |
| IIR | infinite impulse response, see also FIR |
| ILO | internal low-speed oscillator, see also IMO |
| IMO | internal main oscillator, see also ILO |
| INL | integral nonlinearity, see also DNL |
| I/O | input/output, see also GPIO, DIO, SIO, USBIO |
| IPOR | initial power-on reset |
| IPSR | interrupt program status register |
| IRQ | interrupt request |
| ITM | instrumentation trace macrocell |
| LCD | liquid crystal display |
| LIN | Local Interconnect Network, a communications protocol. |
| LR | link register |
| LUT | lookup table |
| LVD | low-voltage detect, see also LVI |
| LVI | low-voltage interrupt, see also HVI |
| LVTTL | low-voltage transistor-transistor logic |
| MAC | multiply-accumulate |
| MCU | microcontroller unit |
| MISO | master-in slave-out |
| NC | no connect |
| NMI | nonmaskable interrupt |
| NRZ | non-return-to-zero |
| NVIC | nested vectored interrupt controller |
| NVL | nonvolatile latch, see also WOL |
| opamp | operational amplifier |
| PAL | programmable array logic, see also PLD |
| PC | program counter |
| PCB | printed circuit board |
| PGA | programmable gain amplifier |
| PHUB | peripheral hub |
| PHY | physical layer |

**Table 14-1. Acronyms Used in this Document** (continued)

| Acronym | Description |
|---------|-------------|
| PICU | port interrupt control unit |
| PLA | programmable logic array |
| PLD | programmable logic device, see also PAL |
| PLL | phase-locked loop |
| PMDD | package material declaration datasheet |
| POR | power-on reset |
| PRES | precise low-voltage reset |
| PRS | pseudo random sequence |
| PS | port read data register |
| PSoC® | Programmable System-on-Chip™ |
| PSRR | power supply rejection ratio |
| PWM | pulse-width modulator |
| RAM | random-access memory |
| RISC | reduced-instruction-set computing |
| RMS | root-mean-square |
| RTC | real-time clock |
| RTL | register transfer language |
| RTR | remote transmission request |
| RX | receive |
| SAR | successive approximation register |
| SC/CT | switched capacitor/continuous time |
| SCL | I$^2$C serial clock |
| SDA | I$^2$C serial data |
| S/H | sample and hold |
| SIO | special input/output, GPIO with advanced features. See GPIO. |
| SNR | signal-to-noise ratio |
| SOC | start of conversion |
| SOF | start of frame |
| SPI | Serial Peripheral Interface, a communications protocol |
| SR | slew rate |
| SRAM | static random access memory |
| SRES | software reset |
| SWD | serial wire debug, a test protocol |
| SWV | single-wire viewer |
| TD | transaction descriptor, see also DMA |
| THD | total harmonic distortion |
| TIA | transimpedance amplifier |
| TRM | technical reference manual |

**Table 14-1. Acronyms Used in this Document** (continued)

| Acronym | Description |
|---------|-------------|
| TTL | transistor-transistor logic |
| TX | transmit |
| UART | Universal Asynchronous Transmitter Receiver, a communications protocol |
| UDB | universal digital block |
| USB | Universal Serial Bus |
| USBIO | USB input/output, PSoC pins used to connect to a USB port |
| VDAC | voltage DAC, see also DAC, IDAC |
| WDT | watchdog timer |
| WOL | write once latch, see also NVL |
| WRES | watchdog timer reset |
| XRES | external reset pin |
| XTAL | crystal |

# 15. Document Conventions

## 15.1 Units of Measure

**Table 15-1. Units of Measure**

| Symbol | Unit of Measure |
|---|---|
| °C | degrees Celsius |
| dB | decibels |
| fF | femtofarads |
| Hz | hertz |
| KB | 1024 bytes |
| kbps | kilobits per second |
| Khr | kilohours |
| kHz | kilohertz |
| kΩ | kilohms |
| ksps | kilosamples per second |
| LSB | least significant bit |
| Mbps | megabits per second |
| MHz | megahertz |
| MΩ | megaohms |
| Msps | megasamples per second |
| μA | microamperes |

**Table 15-1. Units of Measure** (continued)

| Symbol | Unit of Measure |
|---|---|
| μF | microfarads |
| μH | microhenrys |
| μs | microseconds |
| μV | microvolts |
| μW | microwatts |
| mA | milliamperes |
| ms | milliseconds |
| mV | millivolts |
| nA | nanoamperes |
| ns | nanoseconds |
| nV | nanovolts |
| Ω | ohms |
| pF | picofarads |
| ppm | parts per million |
| ps | picoseconds |
| s | seconds |
| sps | samples per second |
| sqrtHz | square root of hertz |
| V | volts |

## Document History Page

| Description Title: PSoC® 5LP: CY8C52LP Family Datasheet Programmable System-on-Chip (PSoC®)<br>Document Number: 001-84933 | | | |
|---|---|---|---|
| **Revision** | **ECN** | **Submission Date** | **Description of Change** |
| ** | 3825653 | 12/07/2012 | Datasheet for new CY8C52LP family. |
| *A | 3897878 | 02/07/2013 | Removed Preliminary status<br>Updated characterization footnotes in Electrical Specifications.<br>Updated conditions for SAR ADC INL and DNL specifications in Table 11-19<br>Updated Table 11-63 (ILO AC Specifications).<br>Changed "UDB Configuration" to "UDB Working Registers" in Table 5-5.<br>Removed references to CAN.<br>Updated VREF accuracy.<br>Updated INL VIDAC spec.<br>Removed drift specs from Voltage Reference Specifications. |
| *B | 3902085 | 02/12/2013 | Changed Hibernate wakeup time from 125 µs to 200 µs in Table 6-3 and Table 11-3. |
| *C | 4114902 | 09/30/2013 | Added information about 1 KB cache in Features.<br>Added warning on reset devices in the EEPROM section.<br>Added DBGEN field in Table 5-3.<br>Deleted statement about repeat start from the I2C section.<br>Removed $T_{STG}$ spec from Table 11-1 and added a note clarifying the maximum storage temperature range.<br>Updated chip Idd, regulator, SAR ADC, IDAC, and VDAC graphs.<br>Added min and max values for the Regulator Output Capacitor parameter.<br>Updated $C_{IN}$ specs in GPIO DC Specifications and SIO DC Specifications.<br>Updated rise and fall time specs in Fast Strong mode in Table 11-9, and deleted related graphs.<br>Updated Voltage Reference Specifications and IMO AC Specifications.<br>Updated 100-TQFP package diagram.<br>Added Appendix for CSP package (preliminary. |
| *D | 4225729 | 12/20/2013 | Added SIO Comparator Specifications.<br>Changed $T_{HIBERNATE}$ max value from 200 to 150.<br>Updated CSP package and ordering information.<br>Added 80 MHz parts in Table 12-1. |
| *E | 4386988 | 05/22/2014 | Updated General Description and Features.<br>Added More Information and PSoC Creator sections.<br>Updated JTAG IDs in Ordering Information.<br>Updated 100-TQFP package diagram. |
| *F | 4587100 | 12/08/2014 | Added link to AN72845 in Note 3.<br>Updated interrupt priority numbers in Section 4.4.<br>Updated Section 5.4 to clarify the factory default values of EEPROM.<br>Corrected ECCEN settings in Table 5-3.<br>Updated Section 6.1.1 and Section 6.1.2.<br>Added a note below Figure 6-4.<br>Updated Figure 6-12.<br>Changed 'Control Store RAM' to 'Dynamic Configuration RAM' in Figure 7-4 and changed Section 7.2.2.2 heading to 'Dynamic Configuration RAM'.<br>Updated Section 7.7. |

## Document History Page (continued)

| Description Title: PSoC® 5LP: CY8C52LP Family Datasheet Programmable System-on-Chip (PSoC®) |
|---|
| Document Number: 001-84933 |

| Revision | ECN | Submission Date | Description of Change |
|---|---|---|---|
| *G | 4698847 | 03/24/2015 | Updated System Integration:<br>Updated Power System:<br>Updated Boost Converter:<br>Updated entire section.<br><br>Updated Electrical Specifications:<br>Updated Power Regulators:<br>Updated Inductive Boost Regulator:<br>Updated Table 11-6:<br>Updated details of $V_{BAT}$, $I_{OUT}$, $V_{OUT}$, $Reg_{LOAD}$, $Reg_{LINE}$ parameters.<br>Removed $V_{OUT}$: $V_{BAT}$ parameter and its details.<br>Removed Table "Inductive Boost Regulator AC Specifications".<br>Updated Table 11-7:<br>Updated details of $L_{BOOST}$, $C_{BOOST}$ parameters.<br>Added $C_{BAT}$ parameter and its details.<br>Added Figure 11-8, Figure 11-9, Figure 11-10, Figure 11-11, Figure 11-12, Figure 11-13, Figure 11-14.<br>Removed Figure "Efficiency vs $I_{OUT}$ $V_{BOOST}$ = 3.3 V, $L_{BOOST}$ = 10 µH".<br>Removed Figure "Efficiency vs $I_{OUT}$ $V_{BOOST}$ = 3.3 V, $L_{BOOST}$ = 22 µH".<br><br>Updated Appendix: CSP Package Summary:<br>Updated Packaging:<br>spec 001-88034 – Changed revision from ** to *A. |
| *H | 4839323 | 07/15/2015 | Added reference to code examples in More Information.<br>Updated typ value of $T_{WRITE}$ from 2 to 10 in EEPROM AC specs table.<br>Changed "Device supply for USB operation" to "Device supply ($V_{DDD}$) for USB operation" in USB DC Specifications.<br>Clarified power supply sequencing and margin for $V_{DDA}$ and $V_{DDD}$.<br>Updated Serial Wire Debug Interface with limitations of debugging on Port 15.<br>Updated Delta-sigma ADC DC Specifications |
| *I | 5030641 | 11/30/2015 | Added Table 2-1.<br>Removed the configurable XRES information.<br>Updated Section 5.6<br>Updated Section 6.3.1.1.<br>Updated values for DSI Fmax, Fgpioin max, and Fsioin max.<br>Corrected the web link for the PSoC 5 Device Programming Specifications in Section 9.<br>Updated CSP Package Bootloader section.<br>Added MHzECO DC Specifications.<br>Updated 99-WLCSP and 100-pin TQFP package drawings.<br>Added a footnote reference for the "CY8C5287AXI-LP095" part in Table 12-1 clarifying that it has 256 KB flash.<br>Added the CY8C5667AXQ-LP040 part in Table 12-1. |
| *J | 5478402 | 10/25/2016 | Updated More Information.<br>Add Links to CAD Libraries in Section 2.<br>Corrected typos in External Electrical Connections. |
| *K | 5703770 | 04/20/2017 | Updated the Cypress logo and copyright information. |
| *L | 5772009 | 06/13/2017 | Added CY8C5266FNI-LP205 in Ordering Information. |
| *M | 6385319 | 11/15/2018 | Added a footnote on DNU pins for the WLCSP pinout table.<br>Updated the links in Sales, Solutions, and Legal Information. |
| *N | 6739970 | 11/27/2019 | Updated the paragraph in section Inputs and Outputs.<br>Completing Sunset review. |

# Sales, Solutions, and Legal Information

## Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

### Products

| | |
|---|---|
| Arm® Cortex® Microcontrollers | cypress.com/arm |
| Automotive | cypress.com/automotive |
| Clocks & Buffers | cypress.com/clocks |
| Interface | cypress.com/interface |
| Internet of Things | cypress.com/iot |
| Memory | cypress.com/memory |
| Microcontrollers | cypress.com/mcu |
| PSoC | cypress.com/psoc |
| Power Management ICs | cypress.com/pmic |
| Touch Sensing | cypress.com/touch |
| USB Controllers | cypress.com/usb |
| Wireless Connectivity | cypress.com/wireless |

### PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

### Cypress Developer Community

Community | Projects | Video | Blogs | Training | Components

### Technical Support

cypress.com/support