



**PRELIMINARY**

**PSoC<sup>®</sup> 5: CY8C52 Family Datasheet**

**Programmable System-on-Chip (PSoC<sup>®</sup>)**

## General Description

With its unique array of configurable blocks, PSoC<sup>®</sup> 5 is a true system-level solution providing microcontroller unit (MCU), memory, analog, and digital peripheral functions in a single chip. The CY8C52 family offers a modern method of signal acquisition, signal processing, and control with high accuracy, high bandwidth, and high flexibility. Analog capability spans the range from thermocouples (near DC voltages) to ultrasonic signals. The CY8C52 family can handle dozens of data acquisition channels and analog inputs on every GPIO pin. The CY8C52 family is also a high-performance configurable digital system with some part numbers including interfaces such as USB, multimaster I<sup>2</sup>C, and controller area network (CAN), a communications protocol. In addition to communication interfaces, the CY8C52 family has an easy-to-configure logic array, flexible routing to all I/O pins, and a high-performance 32-bit ARM<sup>®</sup> Cortex™-M3 microprocessor core. Designers can easily create system level designs using a rich library of prebuilt components and boolean primitives using PSoC Creator™, a hierarchical schematic design entry tool. The CY8C52 family provides unparalleled opportunities for analog and digital bill of materials integration while easily accommodating last minute design changes through simple firmware updates.

## Features

- 32-bit ARM Cortex-M3 CPU core
  - DC to 40 MHz operation
  - Flash program memory, up to 256 KB, 100,000 write cycles, 20-year retention and multiple security features
  - Up to 64 KB SRAM memory
  - 2-KB electrically erasable programmable read-only memory (EEPROM) memory, 1 million cycles, and 20 years retention
  - 24-channel direct memory access (DMA) with multilayer AMBA high-performance bus (AHB) bus access
    - Programmable chained descriptors and priorities
    - High bandwidth 32-bit transfer support
- Low voltage, ultra low power
  - Operating voltage range: 2.7 V to 5.5 V
  - High efficiency boost regulator from 1.8-V input to 5.0-V output
  - 5 mA at 6 MHz
  - Low power modes including:
    - 3- $\mu$ A sleep mode with real time clock (RTC) and low-voltage detect (LVD) interrupt
    - 1- $\mu$ A hibernate mode with RAM retention
- Versatile I/O system
  - 28 to 72 I/Os (62 GPIOs, eight SIOs, two USBIOs<sup>[1]</sup>)
  - Any GPIO to any digital or analog peripheral routability
  - LCD direct drive from any GPIO, up to 46 x 16 segments
  - CapSense<sup>®</sup> support from any GPIO<sup>[2]</sup>
  - 1.2 V to 5.5 V I/O interface voltages, up to four domains
  - Maskable, independent IRQ on any pin or port
  - Schmitt trigger transistor-transistor logic (TTL) inputs
  - All GPIOs configurable as open drain high/low, pull up/down, High-Z, or strong output
  - 25 mA sink on SIO
- Digital peripherals
  - 20 to 24 programmable logic device (PLD) based universal digital blocks (UDBs)
  - Full CAN 2.0b 16 RX, 8 TX buffers<sup>[1]</sup>
  - Full-Speed (FS) USB 2.0 12 Mbps using internal oscillator<sup>[1]</sup>
  - Four 16-bit configurable timer, counter, and PWM blocks
  - Library of standard peripherals
    - 8-, 16-, 24-, and 32-bit timers, counters, and PWMs
    - SPI, UART, and I<sup>2</sup>C
    - Many others available in catalog
  - Library of advanced peripherals
    - Cyclic redundancy check (CRC)
    - Pseudo random sequence (PRS) generator
    - Local interconnect network (LIN) bus 2.0
    - Quadrature decoder
- Analog peripherals (2.7 V  $\leq$  V<sub>DDA</sub>  $\leq$  5.5 V)
  - 1.024 V  $\pm$ 1% internal voltage reference across -40 °C to +85 °C (128 ppm/°C)
  - Successive approximation register (SAR) analog-to-digital converter (ADC), 12-bit at 1 Msps
  - One 8-bit, 8-Msps current DAC (IDAC) or 1-Msps voltage DAC (VDAC)
  - Two comparators with 95-ns response time
  - CapSense support
- Programming, debug, and trace
  - Serial wire debug (SWD) and single-wire viewer (SWV) interfaces
  - Cortex-M3 flash patch and breakpoint (FPB) block
  - Cortex-M3 data watchpoint and trace (DWT) generates data trace information
  - Cortex-M3 Instrumentation Trace Macrocell (ITM) can be used for printf-style debugging
  - DWT and ITM blocks communicate with off-chip debug and trace systems via the SWV interface
  - Bootloader programming supportable through I<sup>2</sup>C, SPI, UART, USB, and other interfaces
- Precision, programmable clocking
  - 3 to 24 MHz internal oscillator over full temperature and voltage range
  - 4 to 25 MHz crystal oscillator for crystal PPM accuracy
  - Internal PLL clock generation up to 40 MHz
  - 32.768 KHz watch crystal oscillator
  - Low power internal oscillator at 1, 33, and 100 kHz
- Temperature and packaging
  - -40 °C to +85 °C degrees industrial temperature
  - 68-pin QFN and 100-pin TQFP package options

### Notes

1. This feature on select devices only. See [Ordering Information](#) on page 88 for details.
2. GPIOs with opamp outputs are not recommended for use with CapSense.

**Contents**

<b>1. Architectural Overview .....</b>	<b>3</b>	8.6 Temp Sensor .....	48
<b>2. Pinouts .....</b>	<b>5</b>	8.7 DAC .....	48
<b>3. Pin Descriptions .....</b>	<b>8</b>	<b>9. Programming, Debug Interfaces, Resources .....</b>	<b>49</b>
<b>4. CPU .....</b>	<b>9</b>	9.1 Debug Port Acquisition .....	49
4.1 ARM Cortex-M3 CPU .....	9	9.2 SWD Interface .....	49
4.2 Cache Controller .....	11	9.3 Debug Features .....	51
4.3 DMA and PHUB .....	11	9.4 Trace Features .....	51
4.4 Interrupt Controller .....	14	9.5 SWV Interface .....	51
<b>5. Memory .....</b>	<b>16</b>	9.6 Programming Features .....	51
5.1 Static RAM .....	16	9.7 Device Security .....	51
5.2 Flash Program Memory .....	16	<b>10. Development Support .....</b>	<b>52</b>
5.3 Flash Security .....	16	10.1 Documentation .....	52
5.4 EEPROM .....	16	10.2 Online .....	52
5.5 Memory Map .....	17	10.3 Tools .....	52
<b>6. System Integration .....</b>	<b>18</b>	<b>11. Electrical Specifications .....</b>	<b>53</b>
6.1 Clocking System .....	18	11.1 Absolute Maximum Ratings .....	53
6.2 Power System .....	21	11.2 Device Level Specifications .....	54
6.3 Reset .....	24	11.3 Power Regulators .....	56
6.4 I/O System and Routing .....	25	11.4 Inputs and Outputs .....	58
<b>7. Digital Subsystem .....</b>	<b>31</b>	11.5 Analog Peripherals .....	66
7.1 Example Peripherals .....	31	11.6 Digital Peripherals .....	77
7.2 Universal Digital Block .....	35	11.7 Memory .....	80
7.3 UDB Array Description .....	38	11.8 PSoC System Resources .....	82
7.4 DSI Routing Interface Description .....	38	11.9 Clocking .....	84
7.5 CAN .....	40	<b>12. Ordering Information .....</b>	<b>88</b>
7.6 USB .....	41	12.1 Part Numbering Conventions .....	88
7.7 Timers, Counters, and PWMs .....	42	<b>13. Packaging .....</b>	<b>89</b>
7.8 I <sup>2</sup> C .....	42	<b>14. Acronyms .....</b>	<b>91</b>
<b>8. Analog Subsystem .....</b>	<b>43</b>	<b>15. Reference Documents .....</b>	<b>92</b>
8.1 Analog Routing .....	44	<b>16. Document Conventions .....</b>	<b>93</b>
8.2 Successive Approximation ADC .....	46	16.1 Units of Measure .....	93
8.3 Comparators .....	46	<b>17. Revision History .....</b>	<b>94</b>
8.4 LCD Direct Drive .....	47	<b>18. Sales, Solutions, and Legal Information .....</b>	<b>95</b>
8.5 CapSense .....	48		

## 1. Architectural Overview

Introducing the CY8C52 family of ultra low power, flash Programmable System-on-Chip (PSoC) devices, part of a scalable 8-bit PSoC 3 and 32-bit PSoC 5 platform. The CY8C52 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**

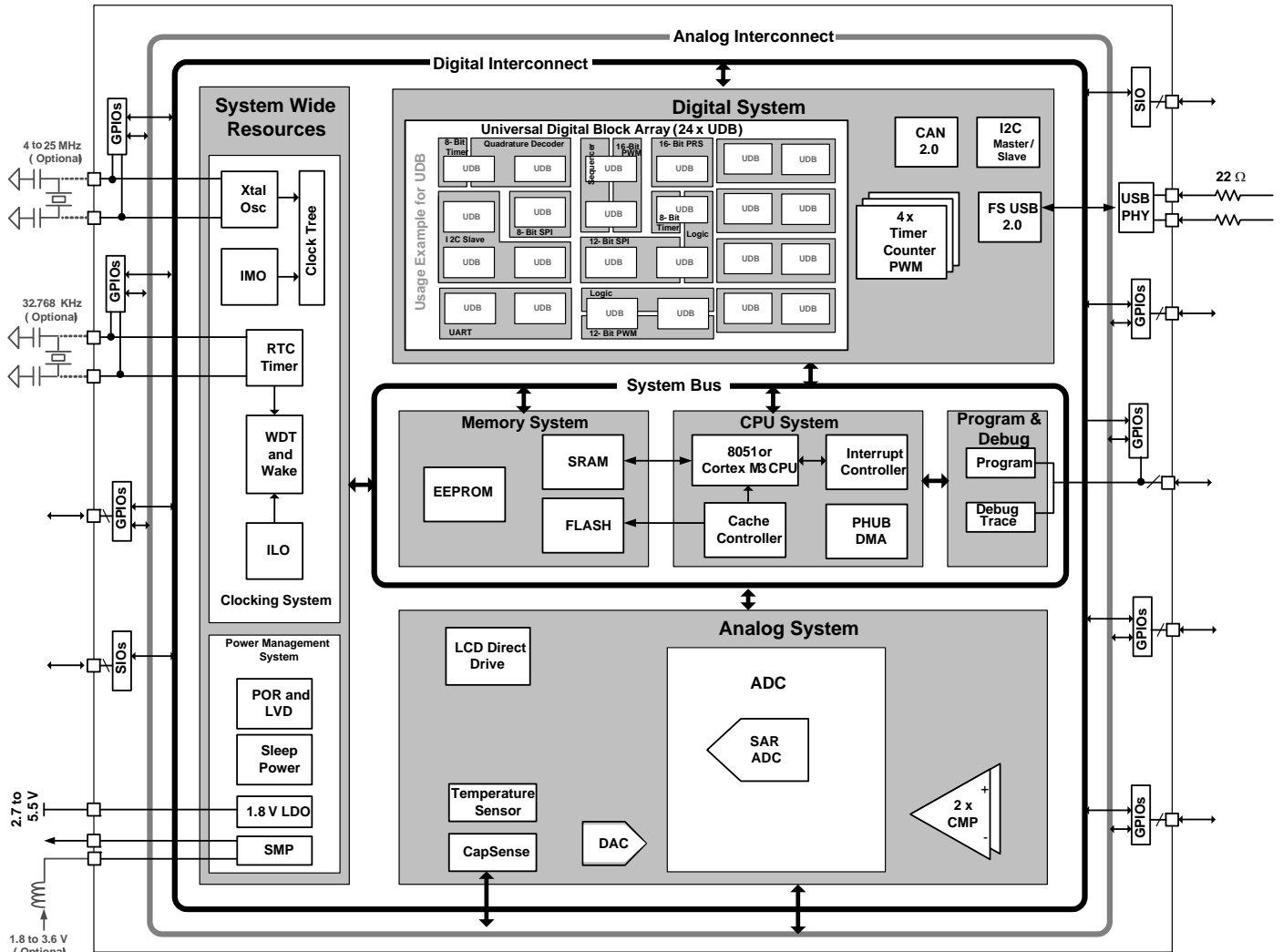


Figure 1-1 on page 3 illustrates the major components of the CY8C52 family. They are:

- 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. The designer 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 CY8C52 family these blocks can include four 16-bit timers, counters, and PWM blocks; I<sup>2</sup>C slave, master, and multimaster; Full-Speed USB; and Full CAN 2.0b.

For more details on the peripherals see the “[Example Peripherals](#)” section on page 31 of this data sheet. For information on UDBs, DSI, and other digital blocks, see the “[Digital Subsystem](#)” section on page 31 of this data sheet.

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 CY8C52 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 43 of this data sheet for more details.

PSoC’s CPU subsystem is built around a 32-bit three-stage pipelined ARM Cortex-M3 processor running at up to 40 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 and byte-writable EEPROM. It provides up to 256 KB of on-chip flash. The CPU can reprogram individual blocks of flash, enabling boot loaders. A powerful and flexible protection model secures the user’s 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.

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 V<sub>DDIO</sub> pins. Every GPIO has analog I/O, LCD drive, flexible interrupt generation, slew rate control, and digital I/O capability. The SIOs on PSoC allow V<sub>OH</sub> to be set independently of V<sub>DDIO</sub> 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 I<sup>2</sup>C 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 25 of this data sheet.

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 1% accuracy at 3 MHz. The IMO can be configured to run from 3 MHz up to 24 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 40 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 CY8C52 family supports a wide supply operating range from 2.7 to 5.5 V. This allows operation from regulated supplies such as 3.3 V ± 10% or 5.0 V ± 10%, or directly from a wide range of battery types. It also provides an integrated high efficiency synchronous boost converter that can power the device from supply voltages as low as 1.8 V. The designer 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<sub>BOOST</sub> 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 1-μA hibernate mode with RAM retention and a 3-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 2 mA when the CPU is running at 6 MHz.

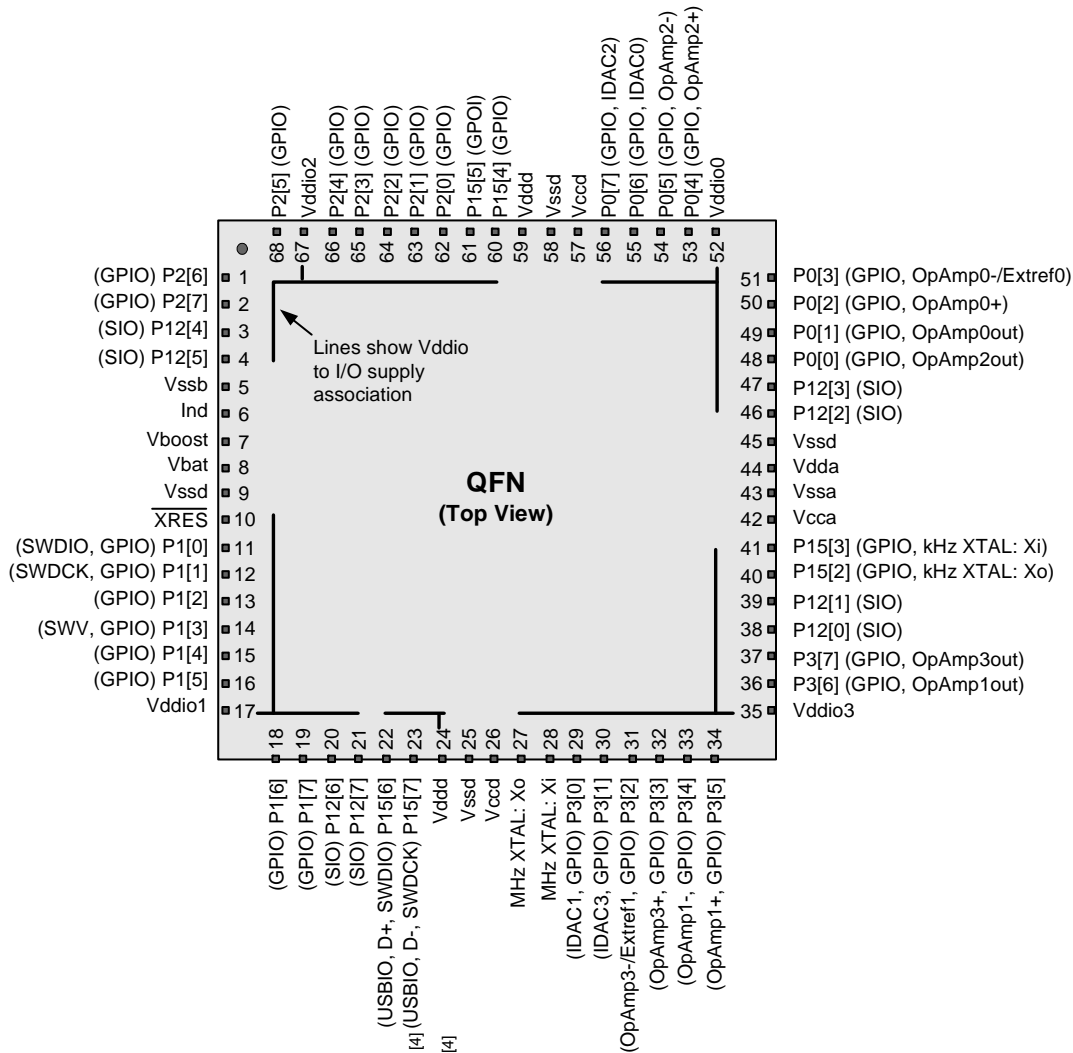
The details of the PSoC power modes are covered in the “[Power System](#)” section on page 21 of this data sheet.

PSoC uses a SWD interface for programming, debug, and test. Using this standard interface enables the designer to 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, 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 49 of this data sheet.

## 2. Pinouts

The Vddio pin that supplies a particular set of pins is indicated by the black lines drawn on the pinout diagrams in [Figure 2-1](#) and [Figure 2-2](#). Using the Vddio pins, a single PSoC can support multiple interface voltage levels, eliminating the need for off-chip level shifters. Each Vddio may sink up to 100 mA total to its associated I/O pins. On the 68-pin and 100-pin devices each set of Vddio associated pins may sink up to 100 mA. The 48 pin device may sink up to 100 mA total for all Vddio0 plus Vddio2 associated I/O pins and 100 mA total for all Vddio1 plus Vddio3 associated I/O pins.

**Figure 2-1. 68-pin QFN Part Pinout<sup>[3]</sup>**



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

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

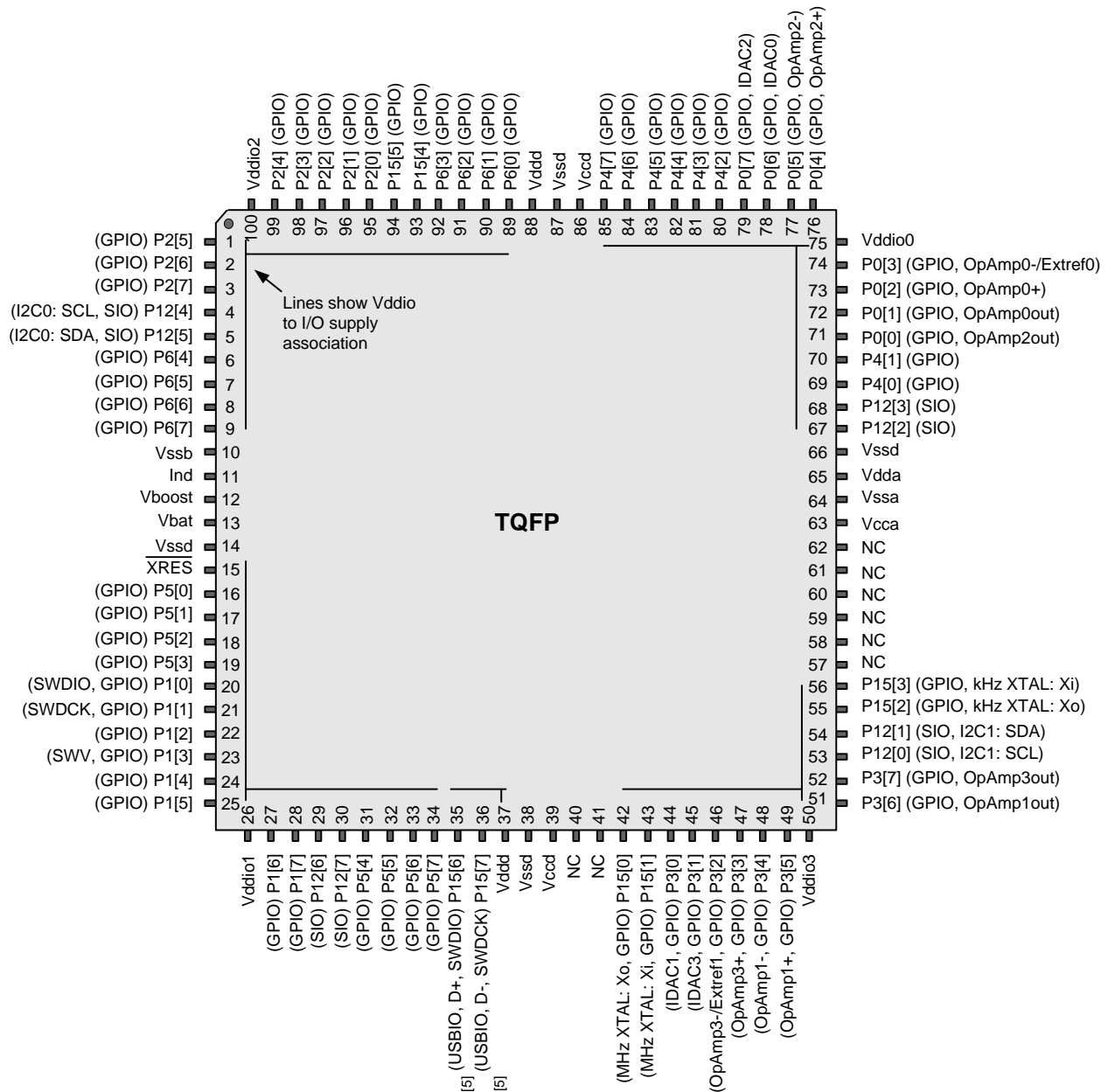


Figure 2-3 and Figure 2-4 show an example schematic and an example PCB layout, for the 100-pin TQFP part, for optimal analog performance on a 2-layer board.

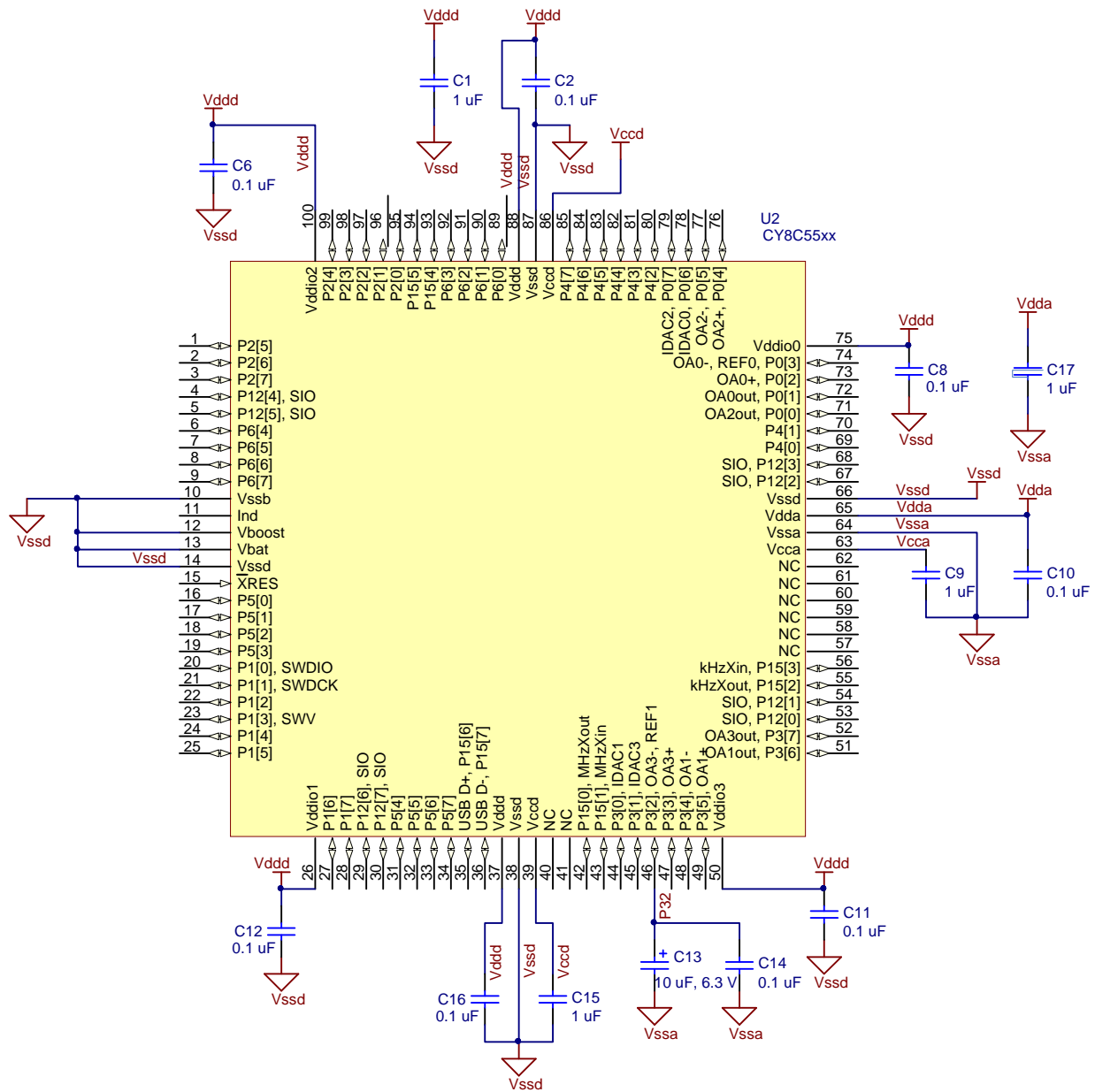
- The two pins labeled Vddd must be connected together.
- The two pins labeled Vccd must be connected together, with capacitance added, as shown in Figure 2-3 and Power System on page 21. The trace between the two Vccd pins should be as short as possible.
- The two pins labeled Vssd must be connected 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**

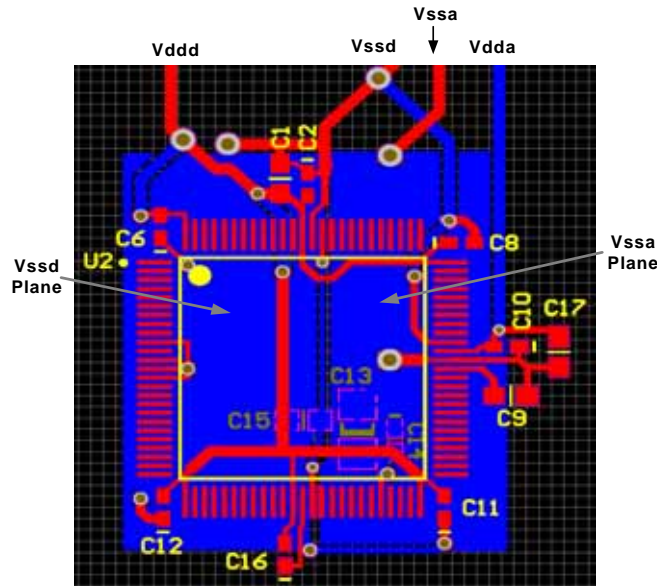
5. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.

Figure 2-3. Example Schematic for 100-pin TQFP Part with Power Connections



**Note** The two Vcccd pins must be connected together with as short a trace as possible. A trace under the device is recommended, as shown in Figure 2-4.

**Figure 2-4. 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.

**GPIO.** General purpose I/O pin provides interfaces to the CPU, digital peripherals, analog peripherals, interrupts, LCD segment drive, and CapSense<sup>[6]</sup>.

**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. If a crystal is not used, then Xi must be shorted to ground and Xo must be left floating.

**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.

**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 V<sub>DDD</sub> instead of from a V<sub>DDIO</sub>. 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<sub>DDD</sub> instead

of from a V<sub>DDIO</sub>. Pins are Do Not Use (DNU) on devices without USB.

**V<sub>BOOST</sub>.** Power sense connection to boost pump.

**V<sub>BAT</sub>.** Battery supply to boost pump.

**V<sub>CCA</sub>.** Output of analog core regulator and input to analog core. Requires a 1 μF capacitor to V<sub>SSA</sub>. Regulator output not for external use.

**V<sub>CCD</sub>.** Output of digital core regulator and input to digital core. The two V<sub>CCD</sub> pins must be shorted together, with the trace between them as short as possible, and a 1-μF capacitor to V<sub>SSD</sub>; see [Power System](#) on page 21. Regulator output not for external use.

**V<sub>DDA</sub>.** Supply for all analog peripherals and analog core regulator. V<sub>DDA</sub> must be the highest voltage present on the device. All other supply pins must be less than or equal to V<sub>DDA</sub>.

**V<sub>DDD</sub>.** Supply for all digital peripherals and digital core regulator. V<sub>DDD</sub> must be less than or equal to V<sub>DDA</sub>.

**V<sub>SSA</sub>.** Ground for all analog peripherals.

**V<sub>SSB</sub>.** Ground connection for boost pump.

**V<sub>SSD</sub>.** Ground for all digital logic and I/O pins.

**V<sub>DDIO0</sub>, V<sub>DDIO1</sub>, V<sub>DDIO2</sub>, V<sub>DDIO3</sub>.** Supply for I/O pins. Each V<sub>DDIO</sub> must be tied to a valid operating voltage (2.7 V to 5.5 V), and must be less than or equal to V<sub>DDA</sub>.

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

**Note**

6. GPIOs with opamp outputs are not recommended for use with CapSense.

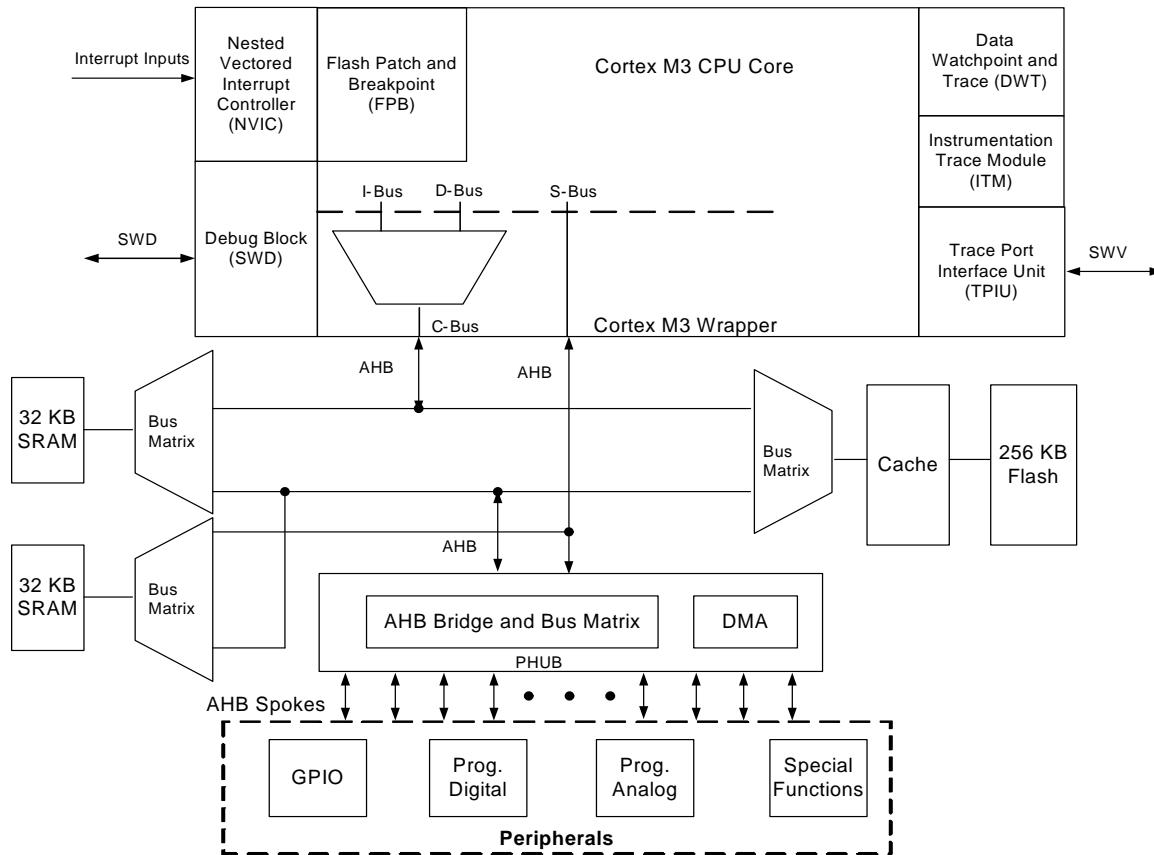


## 4. CPU

### 4.1 ARM Cortex-M3 CPU

The CY8C52 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 module, 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

#### 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<sup>®</sup>-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. Atomic bit-level write and read operations.
- 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. <ul style="list-style-type: none"> <li>■ Low Registers: Registers R0-R7 are accessible by all instructions that specify a general purpose register.</li> <li>■ 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.</li> </ul>
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.
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: <ul style="list-style-type: none"> <li>■ Application program status register (APSR) holds program execution status bits such as zero, carry, negative, in bits[27:31].</li> <li>■ Interrupt program status register (IPSR) holds the current exception number in bits[0:8].</li> <li>■ 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.</li> </ul>
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.

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

Register	Description
CONTROL	A 2-bit register for controlling the operating mode. Bit 0: 0 = privileged level in thread mode, 1 = user level in thread mode. 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 CY8C52 family has a 1 KB 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
2	PHUB local configuration, Power manager, Clocks, IC, EEPROM, Flash programming interface
3	Analog interface and trim, Decimator
4	USB, CAN, I <sup>2</sup> C, 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 127 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.

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:

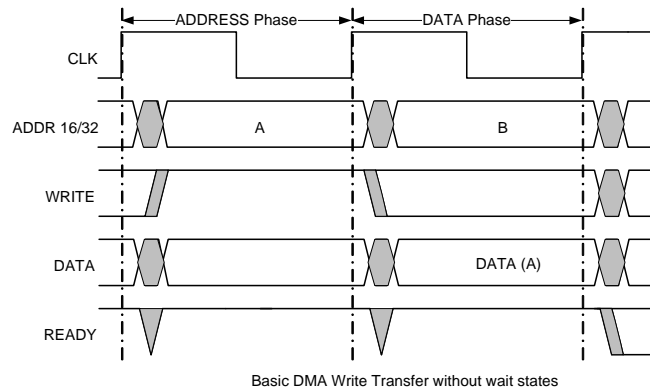
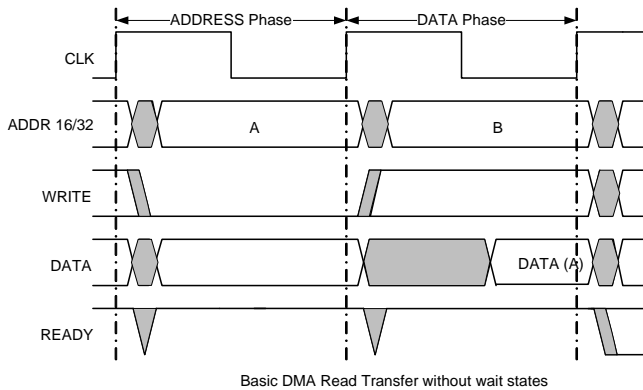
**Table 4-4. Priority Levels**

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

**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**



**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 I<sup>2</sup>C 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 38.

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.
- Configurable number of priority levels: from 3 to 8.
- Dynamic reprioritization of interrupts.
- 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	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]

**Table 4-6. Interrupt Vector Table (continued)**

Interrupt #	Cortex-M3 Exception #	Fixed Function	DMA	UDB
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 <sup>2</sup> C	phub_termout0[15]	udb_intr[15]
16	32	CAN	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	Reserved	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	eeeprom_fault_int	phub_termout1[15]	udb_intr[31]

## 5. Memory

### 5.1 Static RAM

CY8C52 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 and bulk data storage. 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 storing device configuration data and bulk user data. User code may not be run out of this flash memory section. The flash output is 9 bytes wide with 8 bytes of data and 1 additional byte.

The CPU or DMA controller read both user code and bulk data located in flash through the cache controller. This provides higher CPU performance. Flash programming is performed through a special interface and preempts code execution out of flash. Code execution out of cache may continue during flash programming as long as that code is contained inside the cache.

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 the SWD interface. In-system programming, typically used for bootloaders, is also possible using serial interfaces such as I<sup>2</sup>C, 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 configuration or general-purpose 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 51). 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 data sheets. 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 CY8C52 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 two sections, each containing 64 rows of 16 bytes each.

The CPU cannot execute out of EEPROM.



**5.5 Memory Map**

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

**5.5.1 Address Map**

The 4-GB address space is divided into the ranges shown in [Table 5-2](#):

**Table 5-2. 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. This includes a 1 MByte bit-band region starting at 0x40000000 and a 32 Mbyte bit-band alias region starting at 0x42000000.
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 5-3. 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

**Table 5-3. Peripheral Data Address Map (continued)**

Address Range	Purpose
0x40004500 – 0x400045FF	Ports interrupt control
0x40004700 – 0x400047FF	Flash programming interface
0x40004800 – 0x400048FF	Cache controller
0x40004900 – 0x400049FF	I <sup>2</sup> C controller
0x40004E00 – 0x40004EFF	Decimator
0x40004F00 – 0x40004FFF	Fixed timer/counter/PWMs
0x40005000 – 0x400051FF	I/O ports control
0x40005800 – 0x40005FFF	Analog Subsystem Interface
0x40006000 – 0x400060FF	USB Controller
0x40006400 – 0x40006FFF	UDB Configuration
0x40007000 – 0x40007FFF	PHUB Configuration
0x40008000 – 0x400087FF	EEPROM
0x4000A000 – 0x4000A400	CAN
0x40010000 – 0x4001FFFF	Digital Interconnect Configuration
0xE0000000 – 0xE00FFFFFFF	Cortex-M3 PPB Registers, including NVIC, debug, and trace

The bit-band feature allows individual bits in words in the bit-band region 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.

**5.5.2 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 40 MHz clock, accurate to  $\pm 4\%$  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. The designer 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 PSoC.

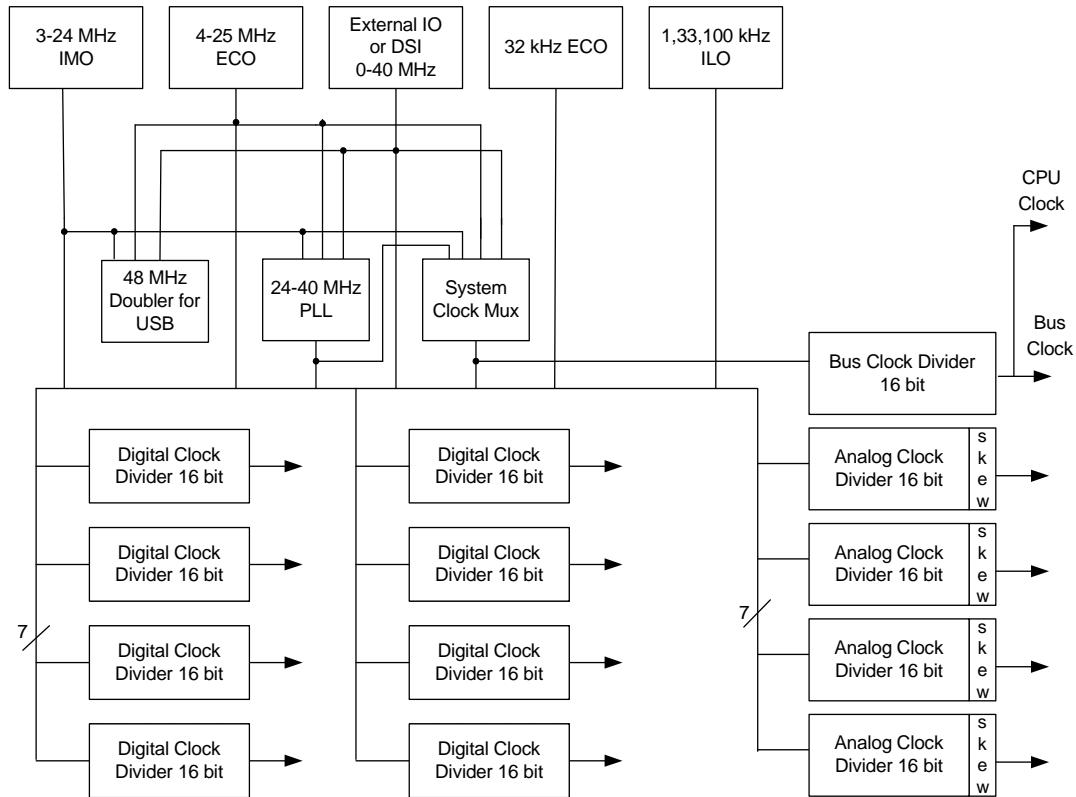
Key features of the clocking system include:

- Seven general purpose clock sources
  - 3 to 24 MHz IMO,  $\pm 4\%$  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 21.
  - DSI signal from an external I/O pin or other logic
  - 24 to 40 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
- Independently sourced clock dividers in all clocks
- 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	$\pm 4\%$ over voltage and temperature	24 MHz	$\pm 10\%$	10 $\mu$ s max
MHzECO	4 MHz	Crystal dependent	25 MHz	Crystal dependent	5 ms typ, max is crystal dependent
DSI	0 MHz	Input dependent	40 MHz	Input dependent	Input dependent
PLL	24 MHz	Input dependent	40 MHz	Input dependent	250 $\mu$ s max
Doubler	48 MHz	Input dependent	48 MHz	Input dependent	1 $\mu$ s 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

#### 6.1.1.1 Internal Main Oscillator

In most designs the IMO is the only clock source required, due to its  $\pm 4\%$  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  $\pm 4\%$  at 3 MHz, up to  $\pm 10\%$  at 24 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, and 24 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 40 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  $\mu$ s (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) works as a low-power system clock to run the CPU. It can also generate time intervals such as fast sleep intervals using the fast timewheel. The fast timewheel is a 100 KHz, 5-bit counter clocked by the ILO that can also be used to wake the system. The fast timewheel settings are programmable, and the counter automatically resets when the terminal count is reached. This enables flexible, periodic wakeups of the CPU at a higher rate than is allowed using the central timewheel. The fast timewheel can generate an optional interrupt each time the terminal count is reached.

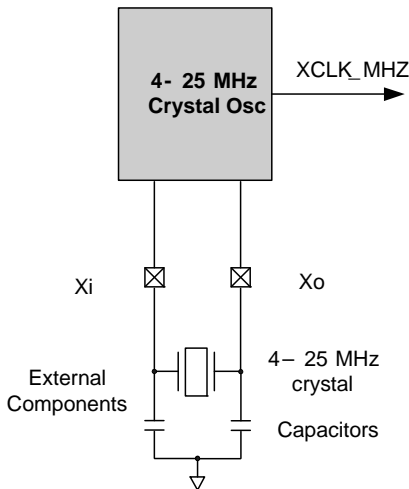
The 33 KHz clock (CLK33K) comes from a divide-by-3 operation on CLK100K. This output can be used as a reduced accuracy version of the 32.768 KHz ECO clock with no need for a crystal.

6.1.2 External Oscillators

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. If a crystal is not used then Xi must be shorted to ground and Xo must be left floating. MHzECO accuracy depends on the crystal chosen.

Figure 6-2. MHzECO Block Diagram

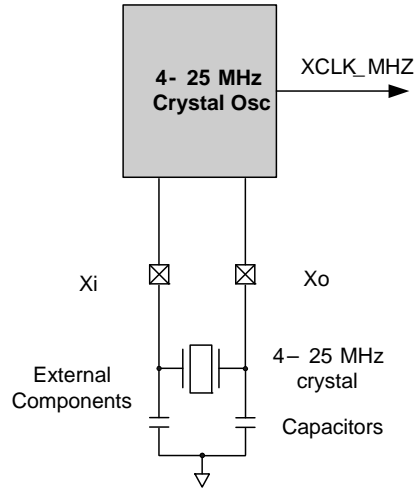


6.1.2.2 32.768 kHz ECO

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



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

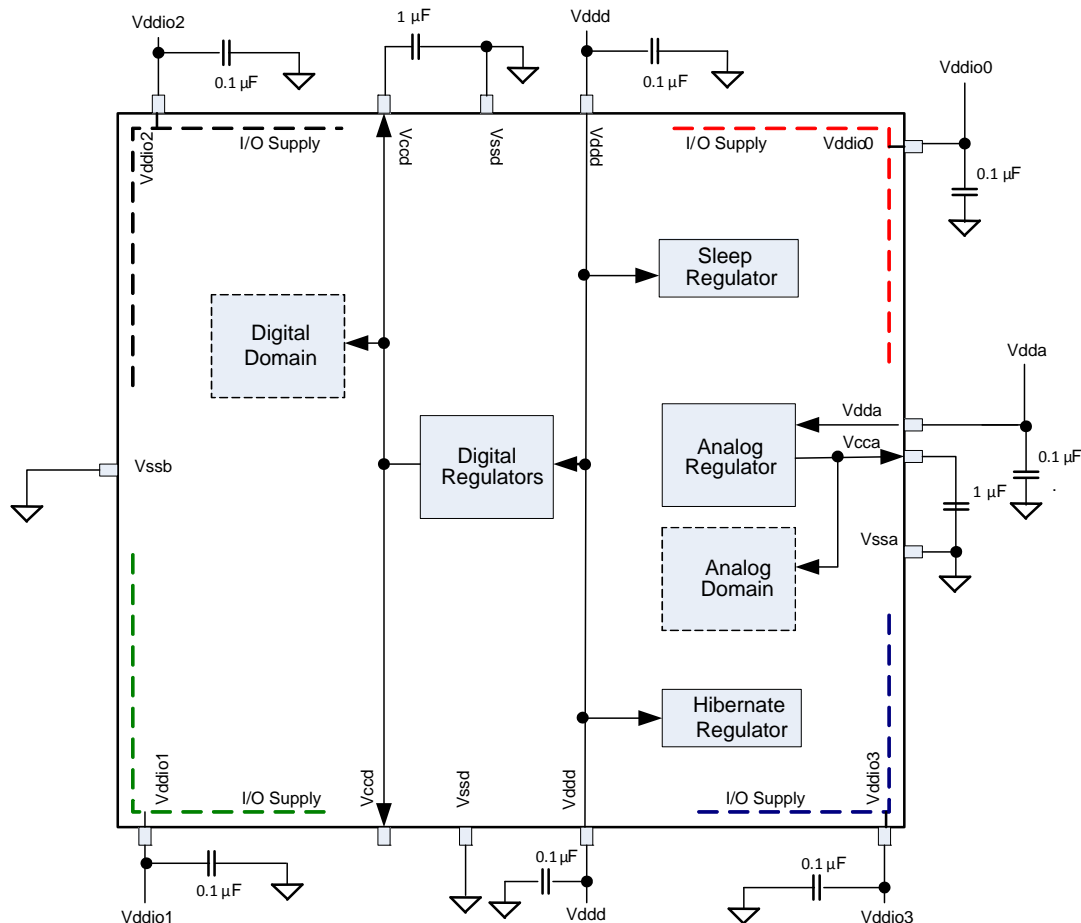
The USB clock domain is unique in that it operates largely asynchronously from the main clock network. The USB logic contains a synchronous bus interface to the chip, while running on an asynchronous clock to process USB data. The USB logic

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  $V_{DDA}$ ,  $V_{DDD}$ , and  $V_{DDIOX}$ , respectively. It also includes two internal 1.8 V regulators that provide the digital ( $V_{CCD}$ ) and analog ( $V_{CCA}$ ) supplies for the internal core logic. The output pins of the regulators ( $V_{CCD}$  and  $V_{CCA}$ ) and the  $V_{DDIO}$  pins must have capacitors connected as shown in Figure 6-4. The two  $V_{CCD}$  pins must be shorted together, with as short a trace as possible, and connected to a  $1\ \mu\text{F} \pm 10\%$  X5R capacitor. The power system also contains a sleep regulator and a hibernate regulator.

Figure 6-4. PSoC Power System



**Note** 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 2-4.

**6.2.1 Power Modes**

PSoC 5 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 5 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](#) on page 23 illustrates the allowable transitions between power modes. Sleep and hibernate modes should not be entered until all V<sub>DDIO</sub> 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 (programmable)	Wakeup, reset, manual register entry	Any interrupt	Any (programmable)	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 (programmable)	All regulators available. Digital and analog regulators can be disabled if external regulation used.
Sleep	All subsystems automatically disabled	Manual register entry	PICU, 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.

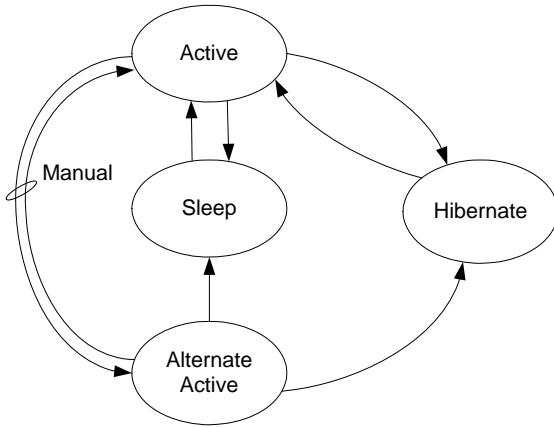
**Table 6-3. Power Modes Wakeup Time and Power Consumption**

Sleep Modes	Wakeup Time	Current (Typ)	Code Execution	Digital Resources	Analog Resources	Clock Sources Available	Wakeup Sources	Reset Sources
Active	–	5 mA <sup>[7]</sup>	Yes	All	All	All	–	All
Alternate Active	–	–	User defined	All	All	All	–	All
Sleep	20 μs typ	3 μA	No	None	None	ILO/kHzECO	PICU, RTC, CTW, LVD	XRES, LVD, WDR
Hibernate	<100 μs	1 μA	No	None	None	None	PICU	XRES

**Note**

7. Bus clock off. Execute from CPU instruction buffer at 6 MHz. See [Table 11-2 on page 54](#)

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  $\mu$ s 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  $\mu$ s.

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. Interrupt sources include internally generated interrupts,

power supervisor, central timewheel, and I/O interrupts. 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 I/O pin (XRES) and WDT.

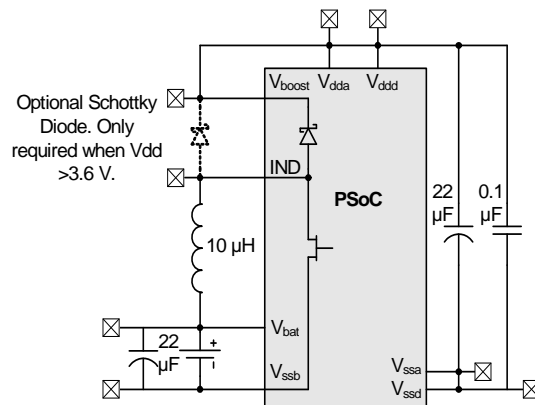
6.2.2 Boost Converter

Applications that use a supply voltage of less than 2.7 V, such as solar or single cell battery supplies, may use the on-chip boost converter. The boost converter may also be used in any system that requires a higher operating voltage than the supply provides. For instance, this includes driving 5.0 V LCD glass in a 3.3 V system. The boost converter accepts an input voltage as low as 1.8 V. With one low cost inductor 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 1.8 V to 3.6 V, and can start up with  $V_{BAT}$  as low as 1.8 V. The converter provides a user configurable output voltage of 3.3 to 5.0 V ( $V_{BOOST}$ ).  $V_{BAT}$  is typically less than  $V_{BOOST}$ ; if  $V_{BAT}$  is greater than or equal to  $V_{BOOST}$ , then  $V_{BOOST}$  will be the same as  $V_{BAT}$ . The block can deliver up to 50 mA ( $I_{BOOST}$ ) depending on configuration.

Four pins are associated with the boost converter:  $V_{BAT}$ ,  $V_{SSB}$ ,  $V_{BOOST}$ , and  $I_{ND}$ . The boosted output voltage is sensed at the  $V_{BOOST}$  pin and must be connected directly to the chip's supply inputs. An inductor is connected between the  $V_{BAT}$  and  $I_{ND}$  pins. The designer can optimize the inductor value to increase the boost converter efficiency based on input voltage, output voltage, current and switching frequency. The External Schottky diode shown in Figure 6-6 is required only in cases when  $V_{BOOST} > 3.6$  V.

Figure 6-6. Application for Boost Converter



The switching frequency can be set to 100 kHz, 400 kHz, 2 MHz, or 32 kHz to optimize efficiency and component cost. The 100 kHz, 400 kHz, and 2 MHz switching frequencies are generated using oscillators internal to the boost converter block. When the 32 kHz switching frequency is selected, the clock is derived from a 32 kHz external crystal oscillator. The 32 kHz external clock is primarily intended for boost standby mode.

At 2 MHz the  $V_{boost}$  output is limited to  $2 \times V_{bat}$ , and at 400 kHz  $V_{boost}$  is limited to  $4 \times V_{bat}$ .

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. The converter can be configured to provide low power, low current regulation in the standby mode. The external 32 kHz crystal can be used to generate inductor boost pulses on the rising and falling edge of the clock when the output voltage is less than the programmed value. This is called automatic thump mode (ATM).

The boost typically draws 200 µA in active mode and 12 µA in standby mode. The boost operating modes must be used in conjunction with chip power modes to minimize the total chip 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 mode	Boost can be operated in either active or standby mode.
Chip-sleep mode	Boost can be operated in either active or standby mode. However, it is recommended to operate boost in standby mode for low power consumption
Chip-hibernate mode	Boost can only be operated in active mode. However, it is recommended not to use boost in chip hibernate mode due to high current consumption in boost active mode

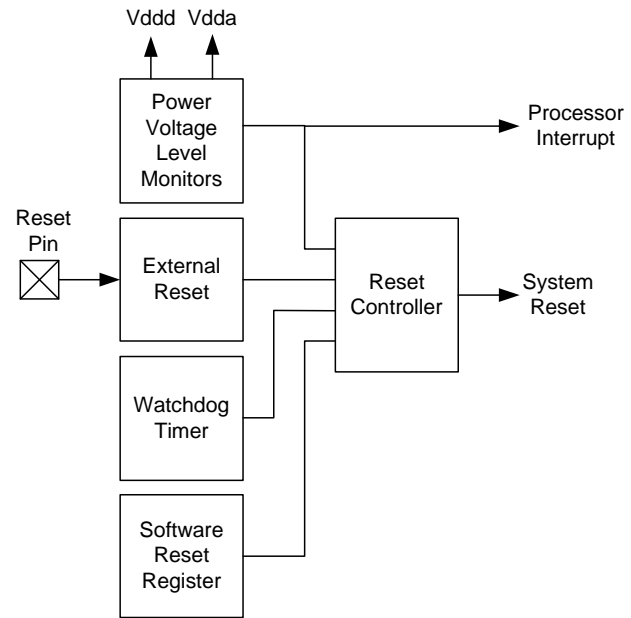
If the boost converter is not used in a given application, tie the V<sub>BAT</sub>, V<sub>SSB</sub>, and V<sub>BOOST</sub> pins to ground and leave the Ind pin unconnected.

### 6.3 Reset

CY8C52 has multiple internal and external reset sources available. The reset sources are:

- Power source monitoring: The analog and digital power voltages, V<sub>DDA</sub>, V<sub>DDD</sub>, V<sub>CCA</sub>, and V<sub>CCD</sub> 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 V<sub>DDIO1</sub>. V<sub>DDD</sub>, V<sub>DDA</sub>, and V<sub>DDIO1</sub> 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. The watchdog timer should not be used during sleep and hibernate modes.
- Software: The device can be reset under program control.

**Figure 6-7. Resets**



The term **system reset** indicates that the processor as well as analog and digital peripherals and registers are reset. A reset status register holds the source of the most recent reset or power voltage monitoring interrupt. The program may examine this register to detect and report exception conditions. This register is cleared after a power on reset.

#### 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<sub>DDD</sub> and V<sub>DDA</sub>, both directly at the pins and at the outputs of the corresponding internal regulators. The trip level is not precise. It is set to approximately 1 volt, which 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 100 ns wide. It may be much wider if one or more of the voltages ramps up slowly.

To save power the IPOR circuit is disabled when the internal digital supply is stable. When the voltage is high enough, the IMO starts.

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

Interrupt circuits are available to detect when V<sub>DDA</sub> and V<sub>DDD</sub> go outside a voltage range. For AHVI, V<sub>DDA</sub> is compared to a fixed trip level. For ALVI and DLVI, V<sub>DDA</sub> and V<sub>DDD</sub> 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	Accuracy
DLVI	V <sub>DDD</sub>	2.7 V-5.5 V	2.71 V-5.45 V in 250 mV increments	±2%
ALVI	V <sub>DDA</sub>	2.7 V-5.5 V	2.71 V-5.45 V in 250 mV increments	±2%
AHVI	V <sub>DDA</sub>	2.7 V-5.5 V	5.75 V	±2%

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.

**6.3.1.2 Other Reset Sources**

■ **XRES - External Reset**

CY8C52 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.

■ **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. The watchdog timer should not be used if the device is to be put into sleep or hibernate mode.

**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 V<sub>DDIO</sub> pins.

There are two types of I/O pins on every device; those with USB provide a third type. Both General Purpose I/O (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<sup>[8]</sup>, and LCD segment drive, while SIO pins are used for voltages in excess of V<sub>DDA</sub> and for programmable output voltages.

■ **Features supported by both GPIO and SIO:**

- 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 on CapSense equipped devices<sup>[8]</sup>
- Analog input and output capability
- Continuous 100 µA clamp current capability
- Standard drive strength down to 2.7 V

■ **Additional features only provided on SIO pins:**

- Higher drive strength than GPIO
- Hot swap capability (5 V tolerance at any operating V<sub>DD</sub>)
- Programmable and regulated high input and output drive levels down to 1.2 V
- No analog input or LCD capability
- Over voltage tolerance up to 5.5 V
- SIO can act as a general purpose analog comparator

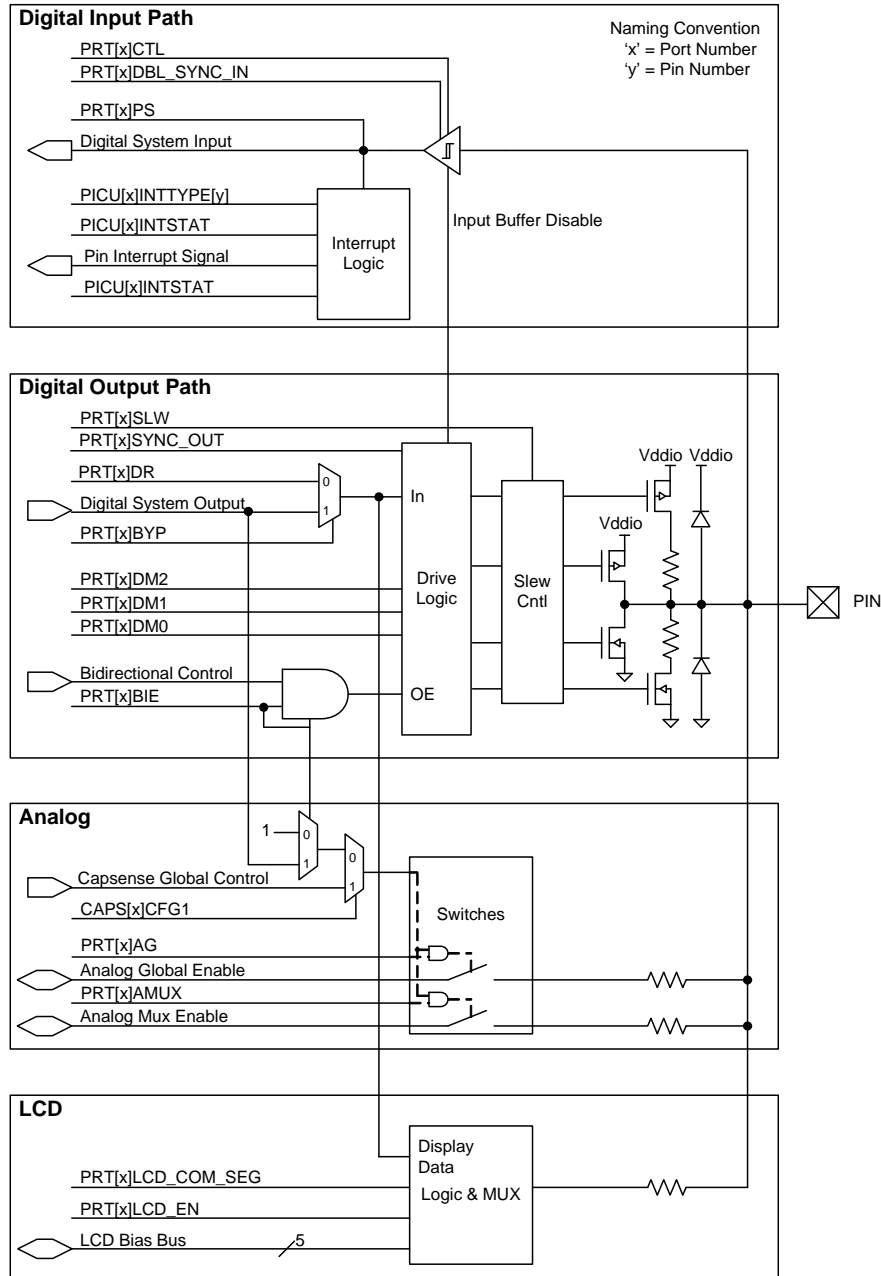
■ **USBIO features:**

- 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

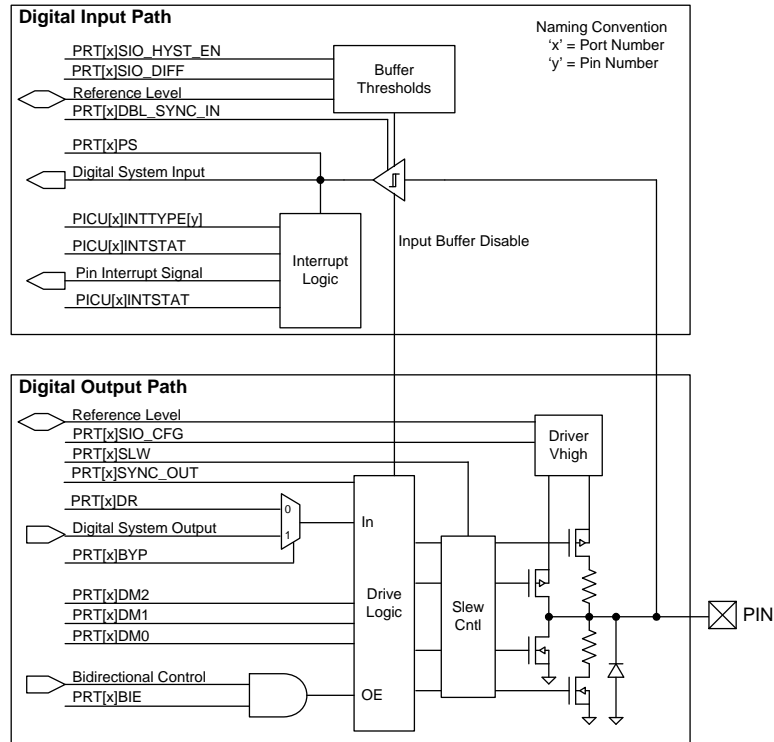
**Note**

8. GPIOs with opamp outputs are not recommended for use with CapSense.

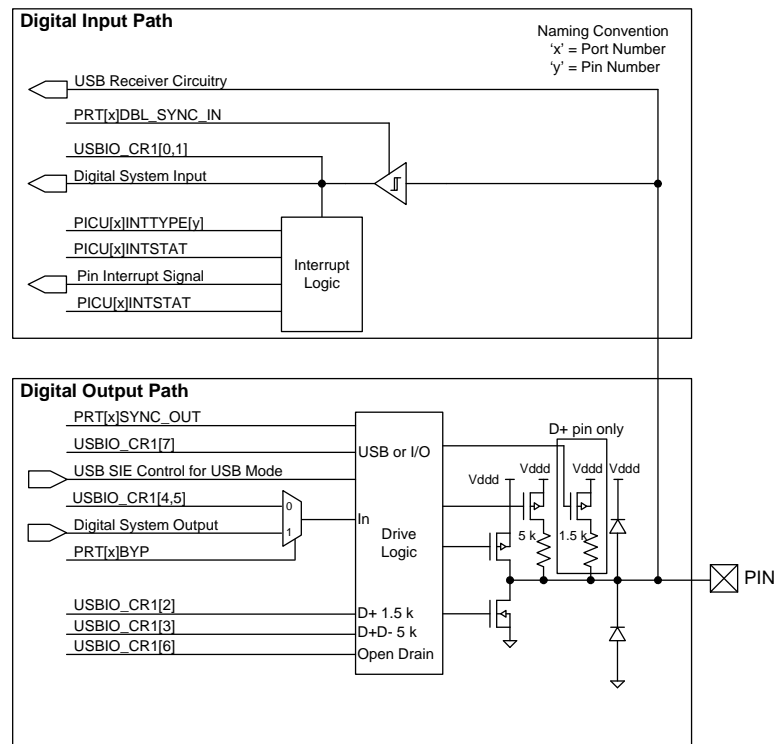
**Figure 6-8. GPIO Block Diagram**



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



**Figure 6-10. USBIO Block Diagram**

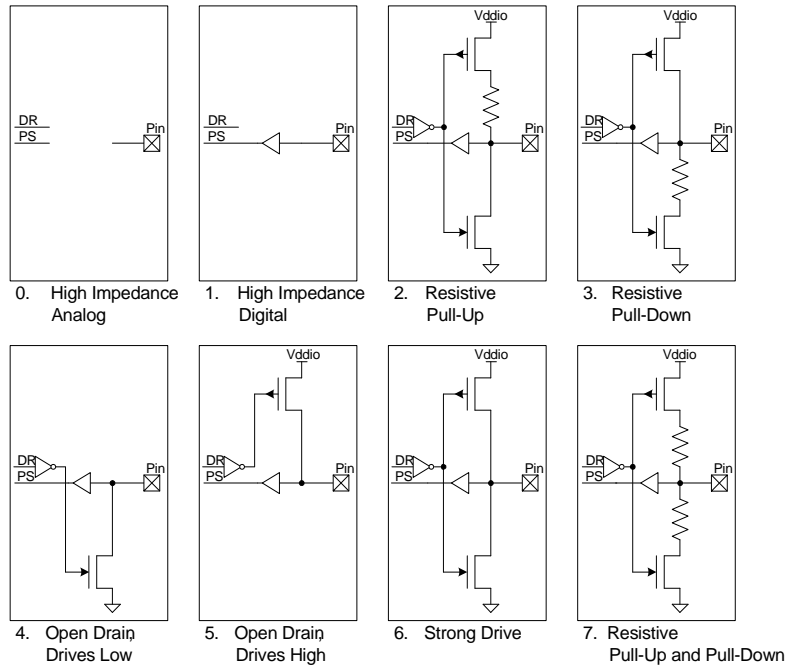


**6.4.1 Drive Modes**

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 6-11 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 6-11. Drive Mode**



**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 <sup>[9]</sup>	0	1	0	Res High (5K)	Strong Low
3	Resistive pull-down <sup>[9]</sup>	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 <sup>[9]</sup>	1	1	1	Res High (5K)	Res Low (5K)

■ 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.

**Note**

9. Resistive pull up and pull down are not available with SIO in regulated output mode.

#### ■ 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<sup>2</sup>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 LVTTTL 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 ( $V_{DDA}$ ) 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  $V_{DDIO}$  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  $V_{DDIO}$  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<sup>[10]</sup>. See the “CapSense” section on page 48 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 47 for details.

#### Note

10. GPIOs with opamp outputs are not recommended for use with CapSense.

**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  $V_{DDIO}$ . SIO pins are individually configurable to output either the standard  $V_{DDIO}$  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 6-12](#)). The DAC on page 48 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.

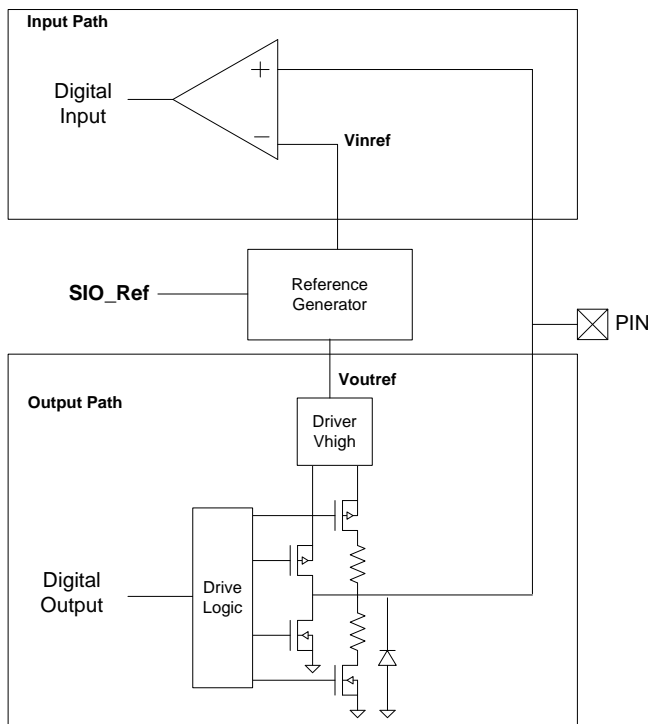
**6.4.12 Adjustable Input Level**

This section applies only to SIO pins. SIO pins by default support the standard CMOS and LVTTTL 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  $V_{DDIO}$ . The reference sets the pins voltage threshold for a high logic level (see [Figure 6-12](#)). Available input thresholds are:

- $0.5 \times V_{DDIO}$
- $0.4 \times V_{DDIO}$
- $0.5 \times V_{REF}$
- $V_{REF}$

Typically the voltage DAC (VDAC) generates the  $V_{REF}$  reference. The DAC on page 48 has more details on VDAC use and reference routing to the SIO pins.

**Figure 6-12. 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 6-9](#) on page 27 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 GPIO pin's protection diode.

**6.4.15 Over Voltage Tolerance**

All I/O pins provide an over voltage ( $V_{DDIO} < V_{IN} < V_{DDA}$ ) tolerance feature at any operating  $V_{DD}$ .

- 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  $\mu$ A using a current limiting resistor. GPIO pins clamp the pin voltage to approximately one diode above the  $V_{DDIO}$  supply.
- In case of a GPIO pin configured for analog input/output, the analog voltage on the pin must not exceed the  $V_{DDIO}$  supply voltage to which the GPIO belongs.

A common application for this feature is connection to a bus such as I<sup>2</sup>C where different devices are running from different supply voltages. In the I<sup>2</sup>C 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 I<sup>2</sup>C bus voltage above the PSoC pin supply. For example, the PSoC chip could operate at 2.7 V, and an external device could run from 5 V. Note that the SIO pin's  $V_{IH}$  and  $V_{IL}$  levels are determined by the associated  $V_{DDIO}$  supply pin.

The I/O pin must be configured into a high impedance drive mode, open drain low drive mode, or pull down drive mode, for over voltage tolerance to work properly. Absolute maximum ratings for the device must be observed for all I/O pins.

**6.4.16 Reset Configuration**

At reset, all I/Os are reset to the High Impedance Analog state.

**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 5. The special features are:

- Digital
  - 4 to 25 MHz crystal oscillator
  - 32.768 KHz crystal oscillator
  - SWD and SWV interface pins
  - External reset
- Analog
  - High current IDAC output
  - External reference inputs

**7. Digital Subsystem**

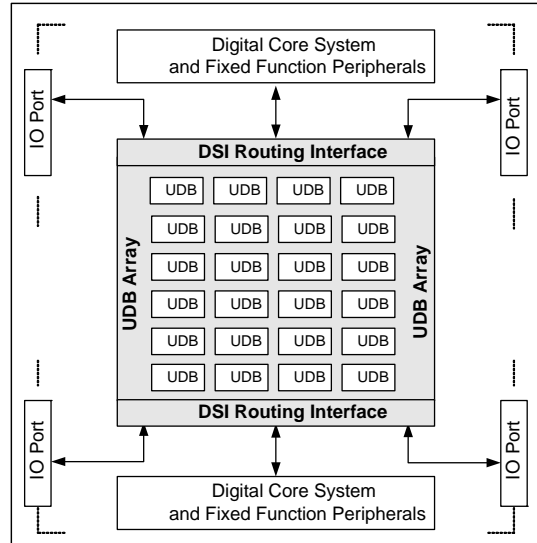
The digital programmable system creates application specific combinations of both standard and advanced digital peripherals and custom logic functions. These peripherals and logic are then interconnected to each other and to any pin on the device, providing a high level of design flexibility and IP security.

The features of the digital programmable system are outlined here to provide an overview of capabilities and architecture. Designers 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. CY8C52 Digital Programmable Architecture**



**7.1 Example Peripherals**

The flexibility of the CY8C52 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 data sheet, and the list is always growing. An example of a component available for use in CY8C52 family, but, not explicitly called out in this data sheet 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 CY8C52 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<sup>2</sup>C (1 to 3 UDBs)
  - UART (1 to 3 UDBs)
- Functions
  - PWM (1 to 2 UDBs)
- Logic (x CPLD product terms per logic function)
  - 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 CY8C52 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 CY8C52 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.

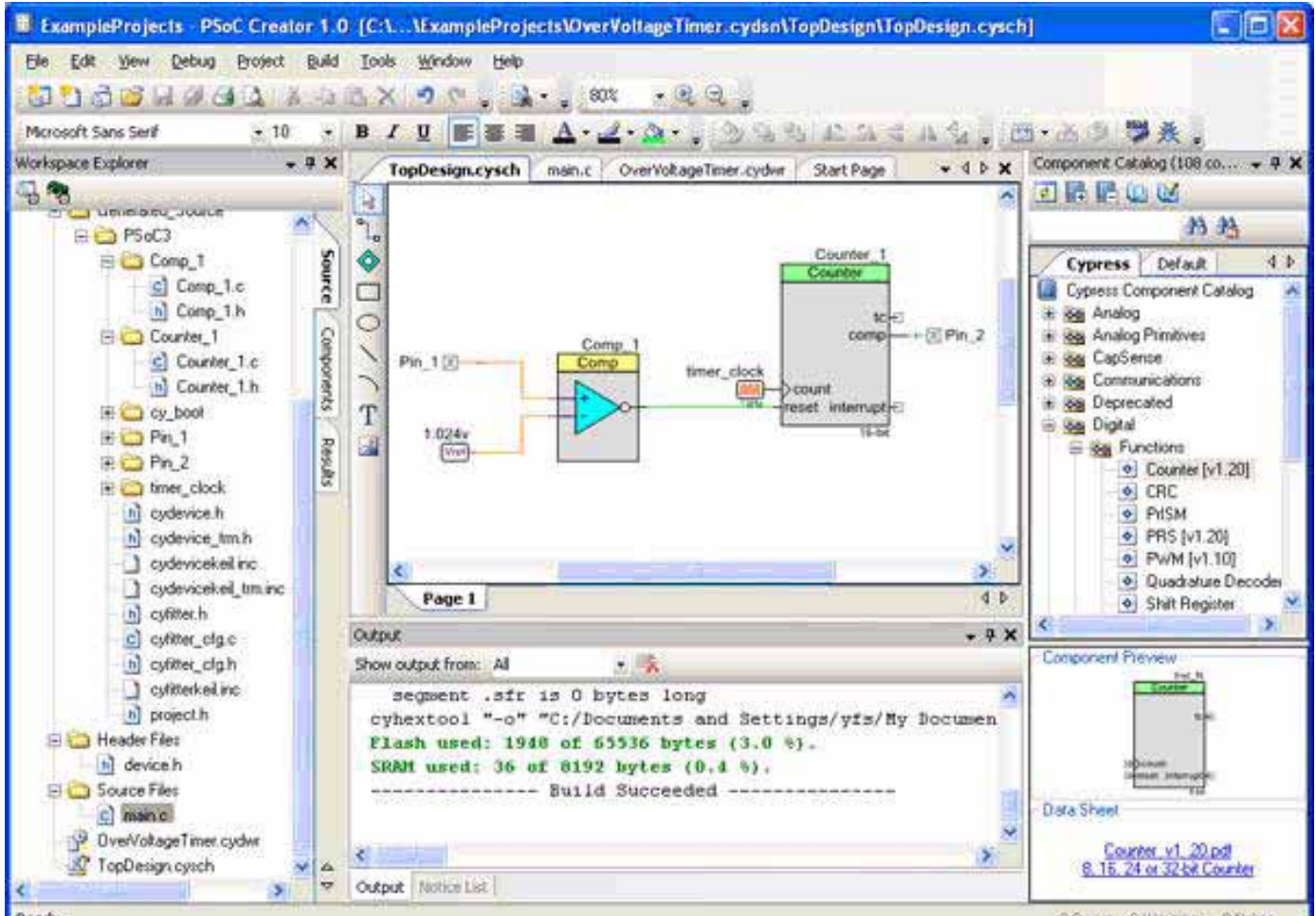
PSoC Creator automatically configures clocks and routes the I/O to the selected pins and then generates APIs to give the application complete control over the hardware. Changing the PSoC device configuration is as simple as adding a new component, setting its parameters, and rebuilding the project.

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.



Figure 7-2. PSoc Creator Framework



7.1.4.2 Component Catalog

**Figure 7-3. Component Catalog**



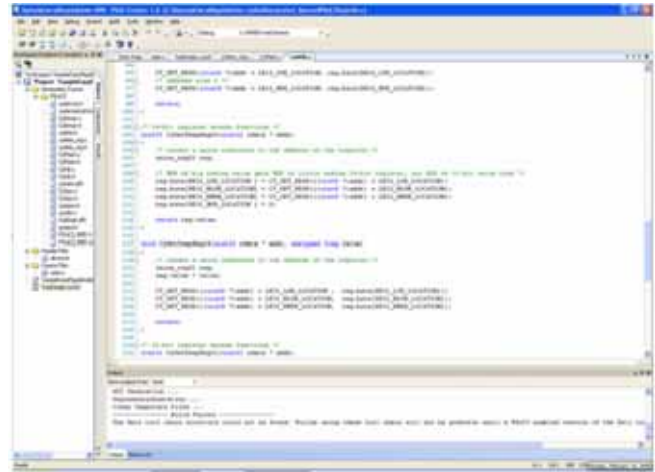
The component catalog is a repository of reusable design elements that select device functionality and customize your PSoC device. It is populated with an impressive selection of content; from simple primitives such as logic gates and device registers, through the digital timers, counters and PWMs, plus analog components such as ADC and DAC, and communication protocols such as I<sup>2</sup>C, USB and CAN. See “[Example Peripherals](#)” section on page 31 for more details about available peripherals. All content is fully characterized and carefully documented in data sheets 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

**Figure 7-4. Code Editor**

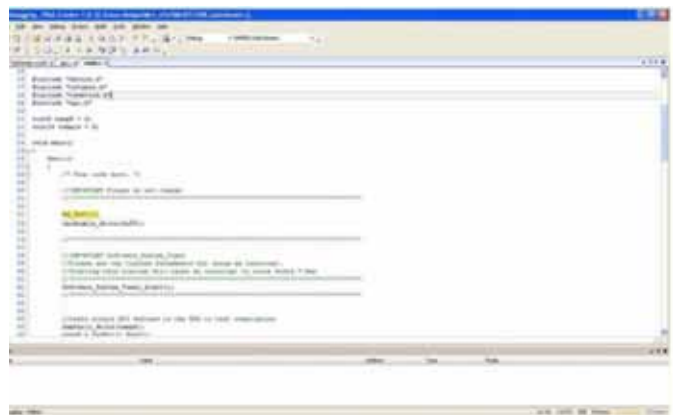


Anchoring the tool is a modern, highly customizable user interface. It includes project management and integrated editors for C and assembler source code, as well the design entry tools.

Project build control leverages compiler technology from top commercial vendors such as ARM<sup>®</sup> Limited, Keil<sup>™</sup>, 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<sup>™</sup> compiler.

7.1.4.5 Nonintrusive Debugging

**Figure 7-5. PSoC Creator Debugger**



With SWD 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.

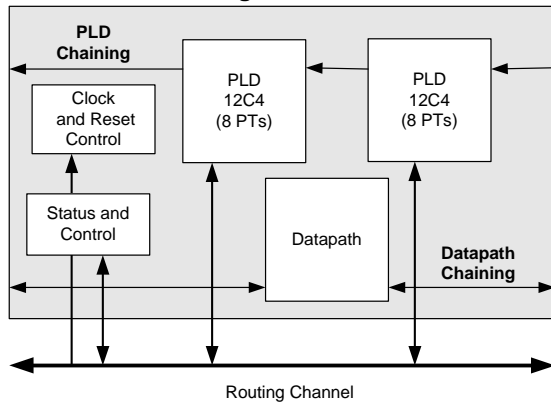
PSoC Creator contains all the tools necessary to complete a design, and then to maintain and extend that design for years to come. All steps of the design flow are carefully integrated and optimized for ease-of-use and to maximize productivity.

**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 I<sup>2</sup>C. Also, the PLD blocks and connectivity provide full featured general purpose programmable logic within the limits of the available resources.

**Figure 7-6. UDB Block Diagram**



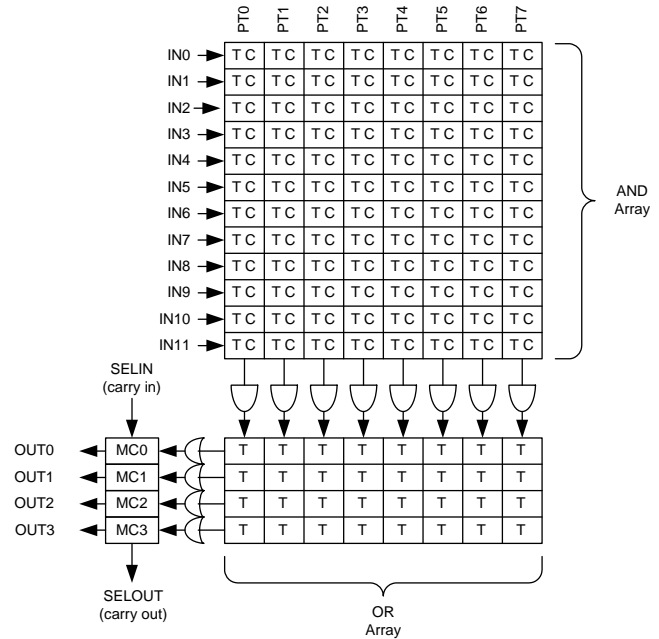
The main component blocks of the UDB are:

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

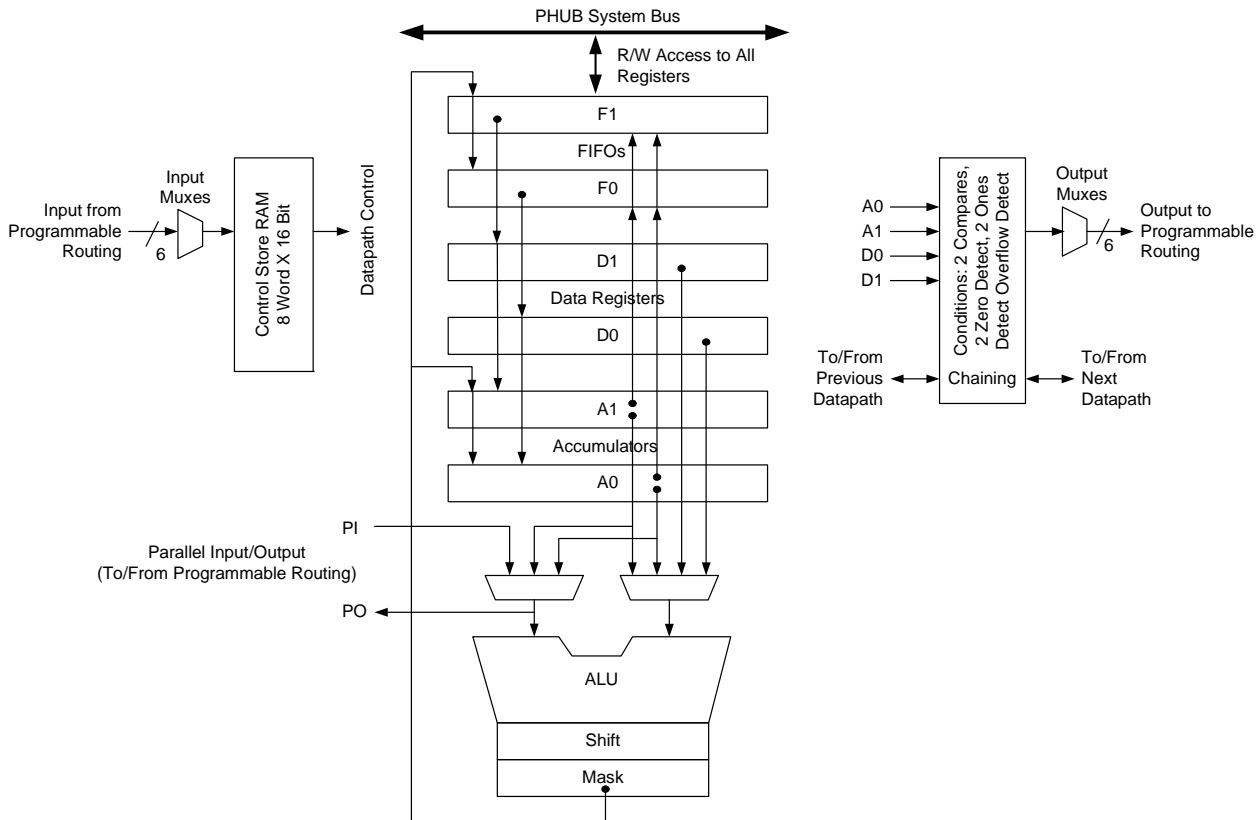


One 12C4 PLD block is shown in Figure 7-7. This PLD has 12 inputs, which feed across eight product terms. Each product term (AND function) can be from 1 to 12 inputs wide, and in a given product term, the true (T) or complement (C) of each input can be selected. The product terms are summed (OR function) 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 7-8. Datapath Top Level**



**7.2.2.6 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.

**7.2.2.7 Dynamic Datapath 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 x 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

**7.2.2.8 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.

**7.2.2.9 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.

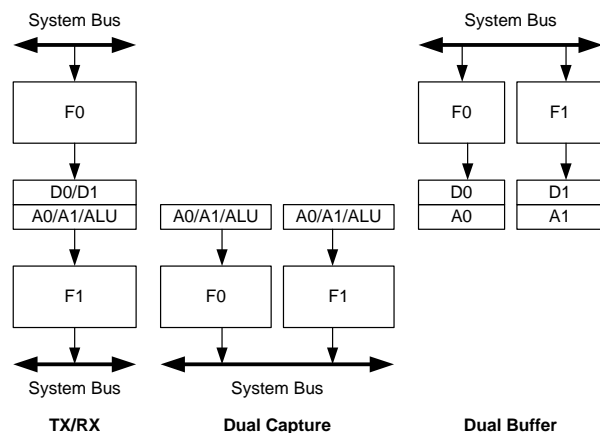
**7.2.2.10 Built in CRC/PRS**

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 used to extend the function into neighboring UDBs.

**7.2.2.11 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 7-9. Example FIFO Configurations**



**7.2.2.12 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.

**7.2.2.13 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.

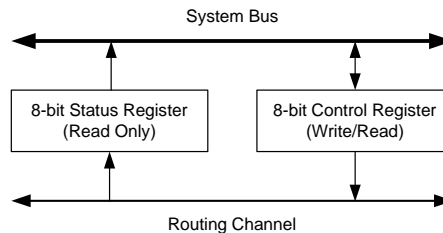
**7.2.2.14 Datapath I/O**

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 7-10. 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.

**7.2.3.15 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.

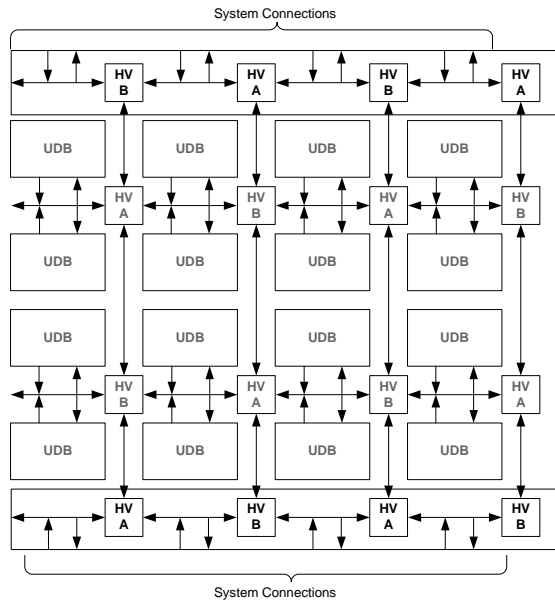
**7.2.3.16 Clock Generation**

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 7-11 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 7-11. Digital System Interface Structure**



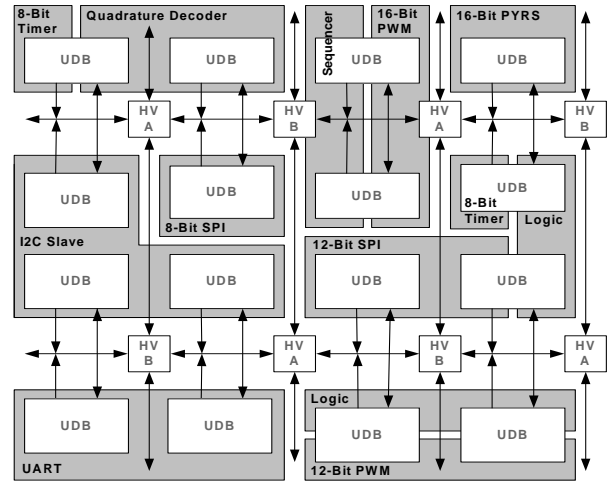
#### 7.3.1 UDB Array Programmable Resources

Figure 7-12 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 7-12. Function Mapping Example in a Bank of UDBs**



### 7.4 DSI Routing Interface Description

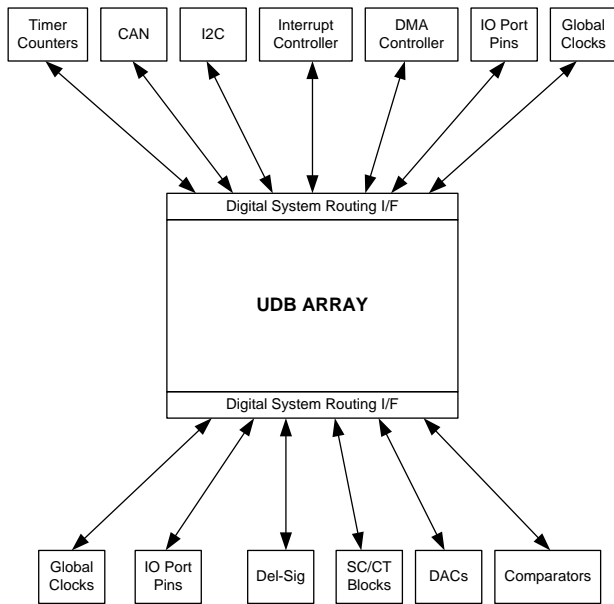
The DSI routing interface is a continuation of the horizontal and vertical routing channels at the top and bottom of the UDB array core. It provides general purpose programmable routing between device peripherals, including UDBs, I/Os, analog peripherals, interrupts, DMA and fixed function peripherals.

Figure 7-13 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:

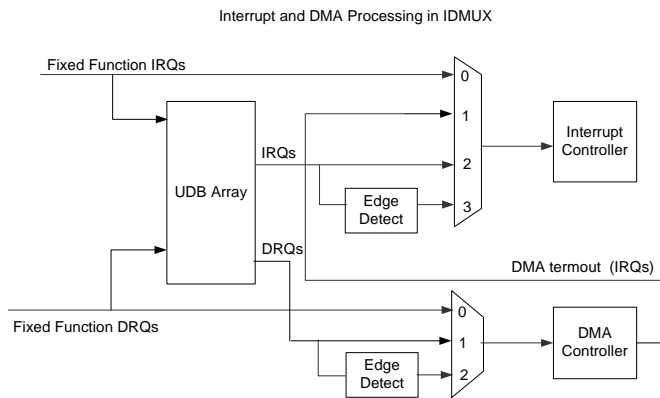
- Interrupt requests from all digital peripherals in the system.
- DMA requests from all digital peripherals in the system.
- Digital peripheral data signals that need flexible routing to I/Os.
- Digital peripheral data signals that need connections to UDBs.
- Connections to the interrupt and DMA controllers.
- Connection to I/O pins.
- Connection to analog system digital signals.

**Figure 7-13. Digital System Interconnect**



Interrupt and DMA routing is very flexible in the CY8C52 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 7-14 shows the structure of the IDMUX (Interrupt/DMA Multiplexer).

**Figure 7-14. Interrupt and DMA Processing in the IDMUX**



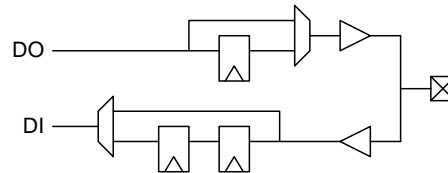
**7.4.1 I/O Port Routing**

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

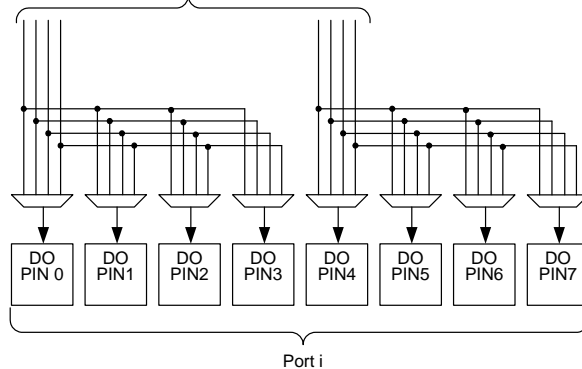
single synchronized (pipelined) and a data input signal has the option to be double synchronized. The synchronization clock is the system clock (see Figure 6-1). Normally all inputs from pins are synchronized as this is required if the CPU interacts with the signal or any signal derived from it. Asynchronous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

**Figure 7-15. I/O Pin Synchronization Routing**



**Figure 7-16. I/O Pin Output Connectivity**

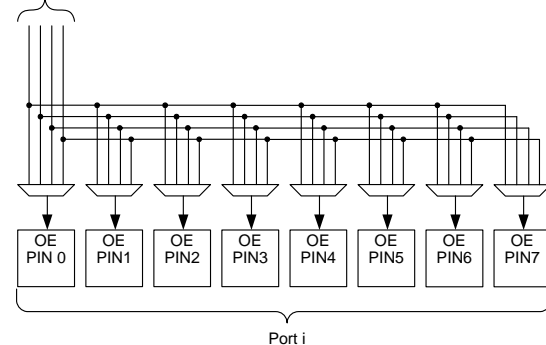
8 IO Data Output Connections from the UDB Array Digital System Interface



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 7-17. I/O Pin Output Enable Connectivity**

4 IO Control Signal Connections from UDB Array Digital System Interface

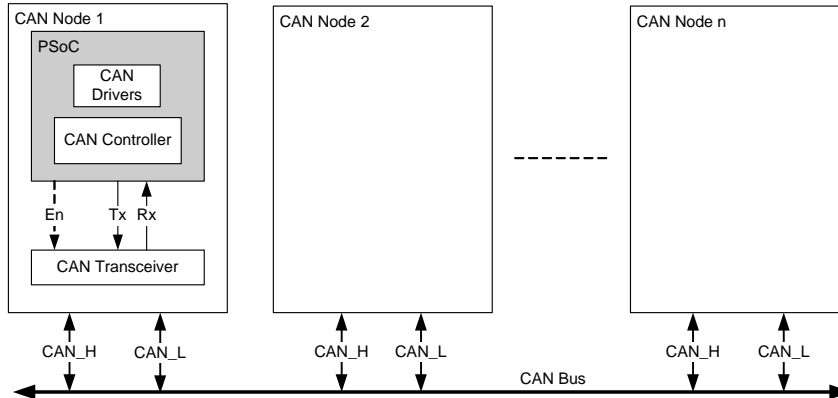


**7.5 CAN**

The CAN peripheral is a fully functional controller area network (CAN) supporting communication baud rates up to 1 Mbps. The CAN controller implements the CAN2.0A and CAN2.0B specifications as defined in the Bosch specification and conforms to the ISO-11898-1 standard. The CAN protocol was originally designed for automotive applications with a focus on a high level of fault detection. This ensures high communication

reliability at a low cost. Because of its success in automotive applications, CAN is used as a standard communication protocol for motion oriented machine control networks (CANOpen) and factory automation applications (DeviceNet). The CAN controller features allow the efficient implementation of higher level protocols without affecting the performance of the microcontroller CPU. Full configuration support is provided in PSoC Creator.

**Figure 7-18. CAN Bus System Implementation**



**7.5.1 CAN Features**

- CAN2.0A/B protocol implementation - ISO 11898 compliant
  - Standard and extended frames with up to 8 bytes of data per frame
  - Message filter capabilities
  - Remote Transmission Request (RTR) support
  - Programmable bit rate up to 1 Mbps
- Listen Only mode
- SW readable error counter and indicator
- Sleep mode: Wake the device from sleep with activity on the Rx pin
- Supports two or three wire interface to external transceiver (Tx, Rx, and Enable). The three-wire interface is compatible with the Philips PHY; the PHY is not included on-chip. The three wires can be routed to any I/O
- Enhanced interrupt controller
  - CAN receive and transmit buffers status
  - CAN controller error status including BusOff

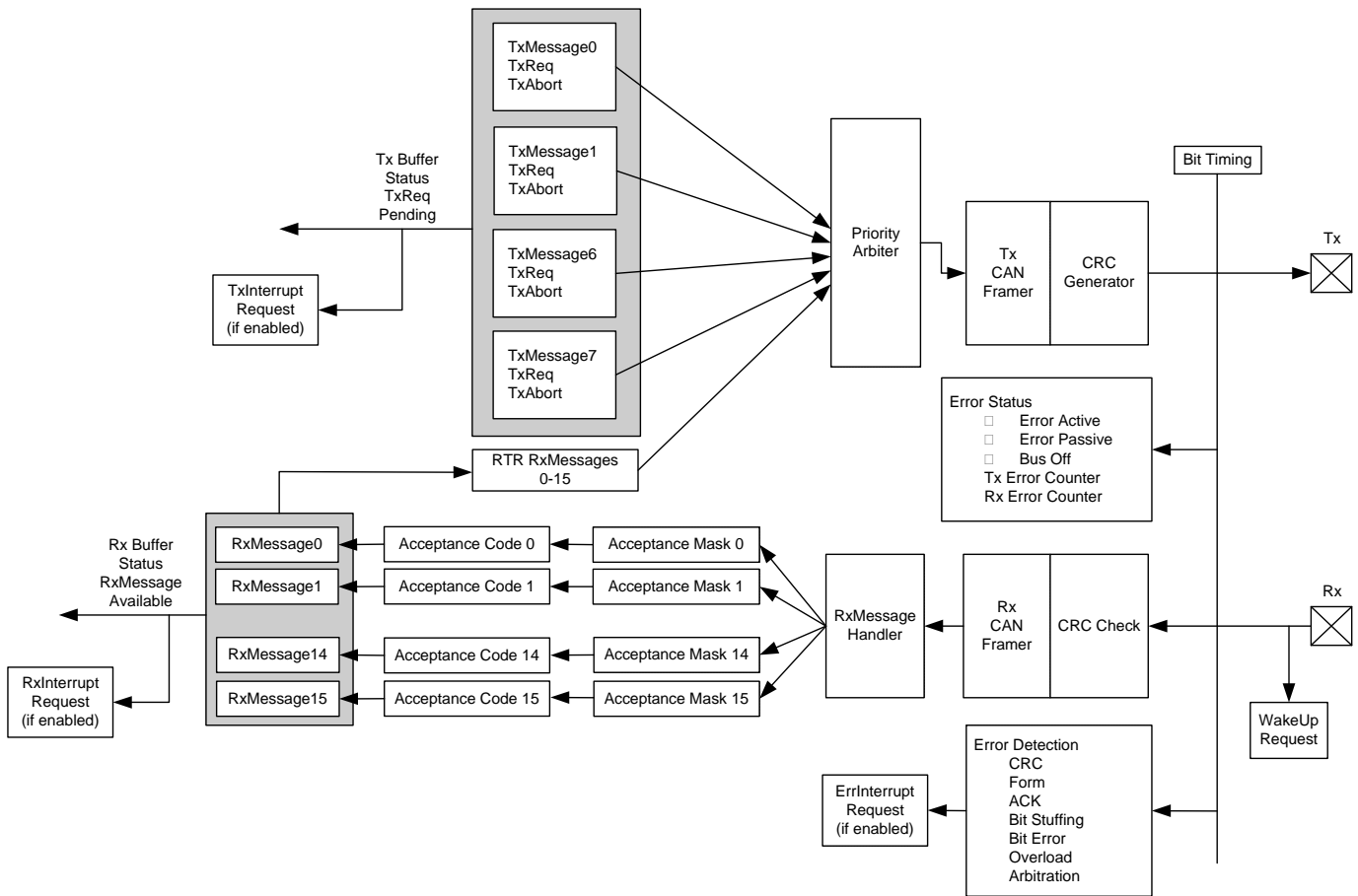
- Receive path
  - 16 receive buffers each with its own message filter
  - Enhanced hardware message filter implementation that covers the ID, IDE and RTR
  - DeviceNet addressing support
  - Multiple receive buffers linkable to build a larger receive message array
  - Automatic transmission request (RTR) response handler
  - Lost received message notification
- Transmit path
  - Eight transmit buffers
  - Programmable transmit priority
  - Round robin
  - Fixed priority
  - Message transmissions abort capability

**7.5.2 Software Tools Support**

- CAN Controller configuration integrated into PSoC Creator:
- CAN Configuration walkthrough with bit timing analyzer
  - Receive filter setup



Figure 7-19. CAN Controller Block Diagram



7.6 USB

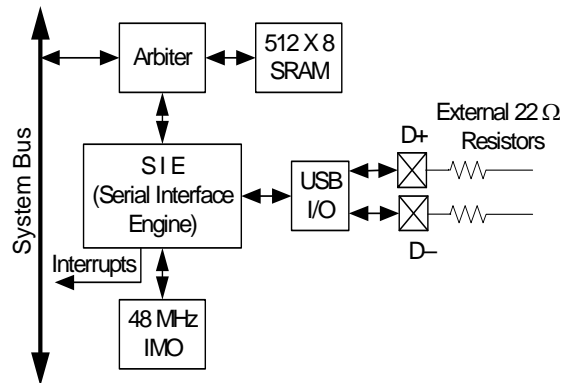
PSoc includes a dedicated Full-Speed (12 Mbps) USB 2.0 transceiver supporting all four USB transfer types: control, interrupt, bulk, and isochronous. PSoc Creator provides full configuration support. USB interfaces to hosts through two dedicated USBIO pins, which are detailed in the “I/O System and Routing” section on page 25.

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
- Two memory modes
  - Manual Memory Management with No DMA Access
  - Manual Memory Management with Manual 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 7-20. USB



**7.7 Timers, Counters, and PWMs**

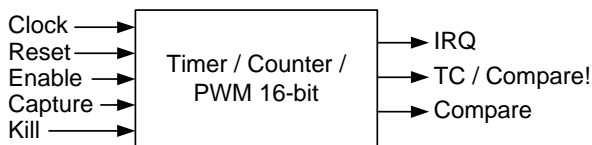
The Timer/Counter/PWM peripheral is a 16-bit dedicated peripheral providing three of the most common embedded peripheral features. As almost all embedded systems use some combination of timers, counters, and PWMs. Four of them have been included on this PSoC device family. Additional and more advanced functionality timers, counters, and PWMs can also be instantiated in Universal Digital Blocks (UDBs) as required. PSoC Creator allows designers 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 7-21. Timer/Counter/PWM**



**7.8 I<sup>2</sup>C**

The I<sup>2</sup>C peripheral provides a synchronous two wire interface designed to interface the PSoC device with a two wire I<sup>2</sup>C serial communication bus. The bus is compliant with Philips 'The I<sup>2</sup>C Specification' version 2.1. Additional I<sup>2</sup>C interfaces can be instantiated using Universal Digital Blocks (UDBs) in PSoC Creator, as required.

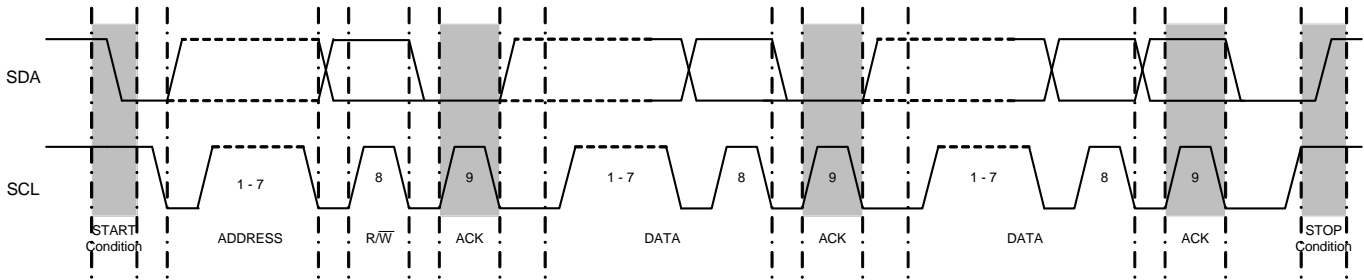
To eliminate the need for excessive CPU intervention and overhead, I<sup>2</sup>C specific support is provided for status detection and generation of framing bits. I<sup>2</sup>C operates as a slave, a master, or multimaster (Slave and Master). In slave mode, the unit always listens for a start condition 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. I<sup>2</sup>C interfaces through the DSI routing and allows direct connections to any GPIO or SIO pins.

I<sup>2</sup>C 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 (3.4 Mbps in UDBs)
- 7 or 10-bit addressing (10-bit addressing requires firmware support)
- SMBus operation (through firmware support - SMBus supported in hardware in UDBs)

Data transfers follow the format shown in [Figure 7-22](#). 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. However, if a master still wishes to communicate on the bus, it can generate a repeated START condition (Sr) and address another slave without first generating a STOP condition. Various combinations of read/write formats are then possible within such a transfer.

**Figure 7-22. I<sup>2</sup>C Complete Transfer Timing**



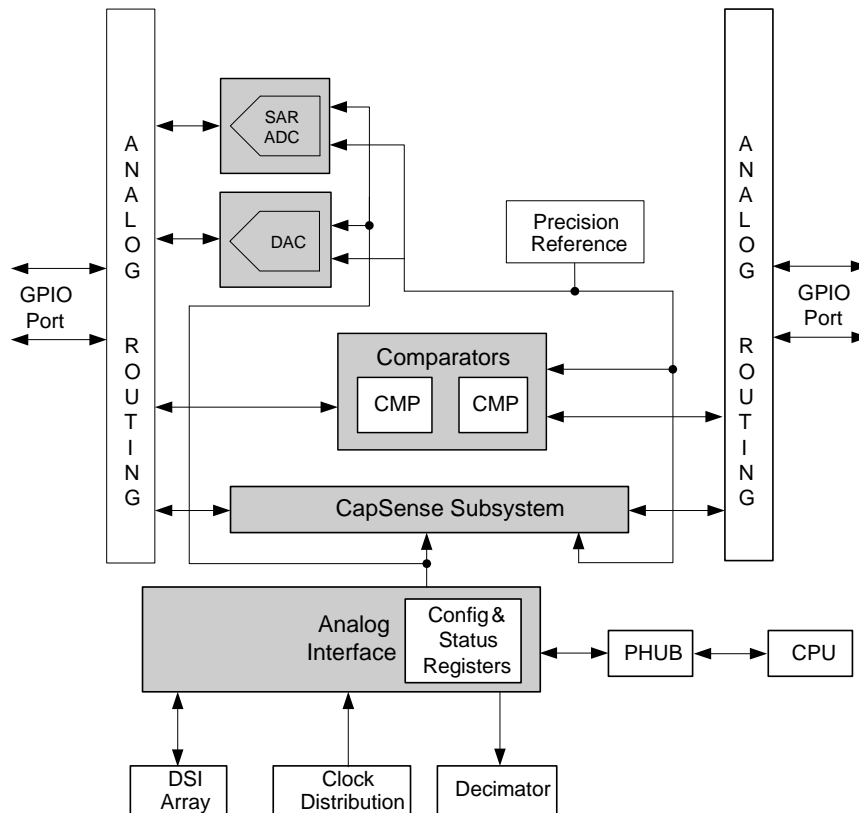
## 8. Analog Subsystem

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 CY8C38 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. All analog routing switches are open when the device is in sleep or hibernate mode.

For information on how to make pin selections for optimal analog routing, refer to the application note, [AN58304 - PSoC<sup>®</sup> 3 and PSoC<sup>®</sup> 5 - Pin Selection for Analog Designs](#).

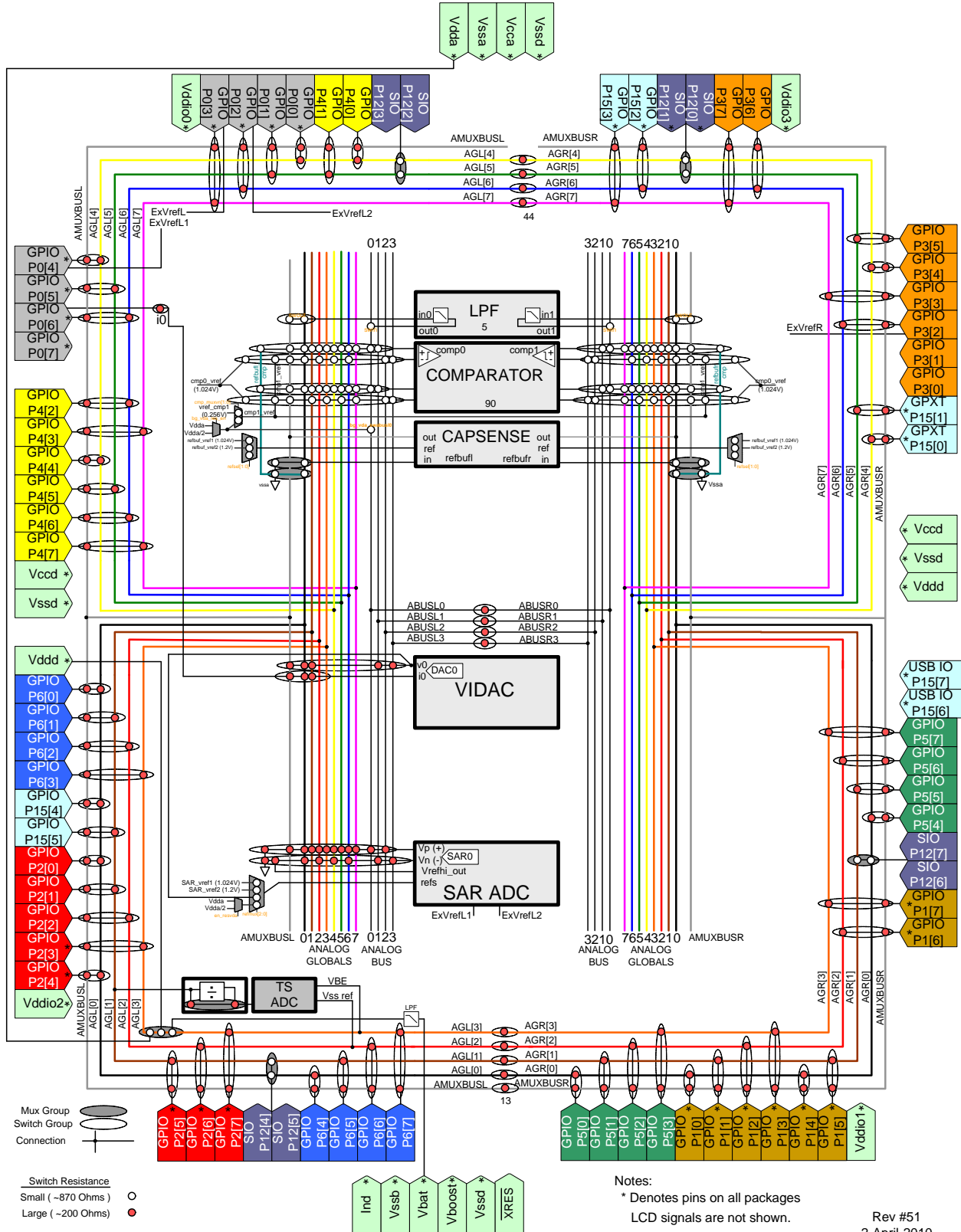
### 8.1.1 Features

- 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 CY8C38 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 CY8C38, one in the left half (AMUXBUSL) and one in the right half (AMUXBUSR), as shown in [Figure 8-2](#).

**Figure 8-2. CY8C52 Analog Interconnect**



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 CY8C38, 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.

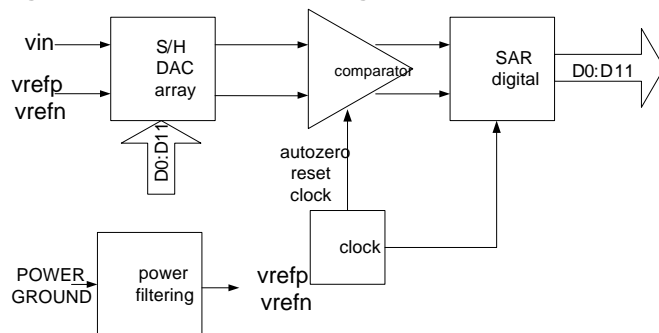
## 8.2 Successive Approximation ADC

The CY8C52 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 16 times the sample rate; the maximum clock rate is 16 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  $\mu$ s 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

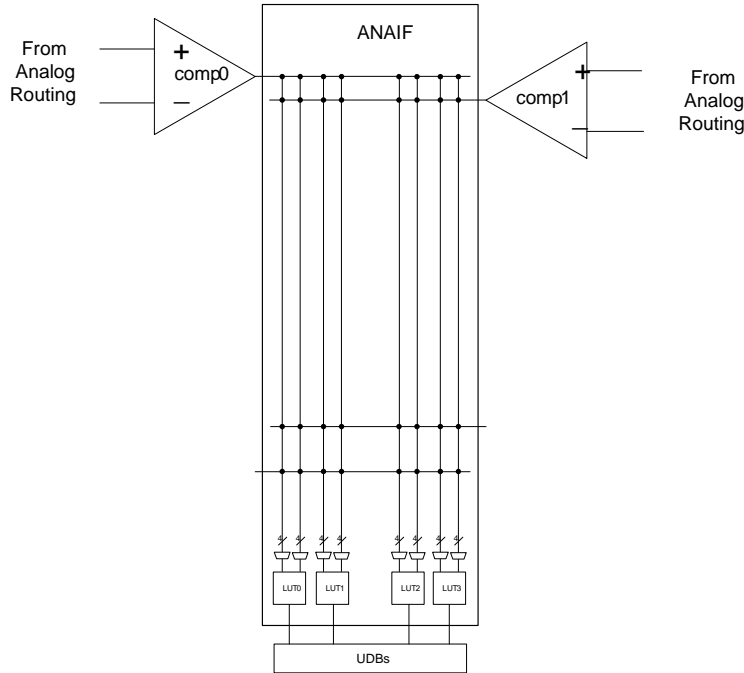
The CY8C52 family of devices contains two comparators in a device. Comparators have these features:

- Input offset factory trimmed to less than 5 mV
- Rail-to-rail common mode input range ( $V_{SSA}$  to  $V_{CCA}$ )
- 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 CY8C52 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	<b>FALSE ('0')</b>
0001b	<b>A AND B</b>
0010b	<b>A AND (NOT B)</b>
0011b	<b>A</b>
0100b	<b>(NOT A) AND B</b>
0101b	<b>B</b>
0110b	<b>A XOR B</b>
0111b	<b>A OR B</b>
1000b	<b>A NOR B</b>
1001b	<b>A XNOR B</b>
1010b	<b>NOT B</b>
1011b	<b>A OR (NOT B)</b>
1100b	<b>NOT A</b>
1101b	<b>(NOT A) OR B</b>
1110b	<b>A NAND B</b>
1111b	<b>TRUE ('1')</b>

**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 CY8C52 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.

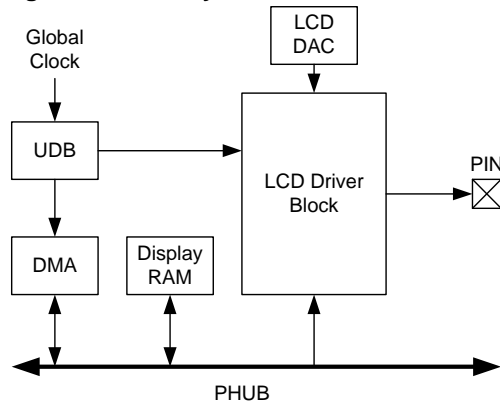
PSoC Creator provides an LCD segment drive component. The component wizard provides easy and flexible configuration of LCD resources. You can specify pins for segments and commons along with other options. The software configures the device to meet the required specifications. This is possible because 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 x 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
- LCD driver configurable to be active when PSoC is in limited active mode

**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.

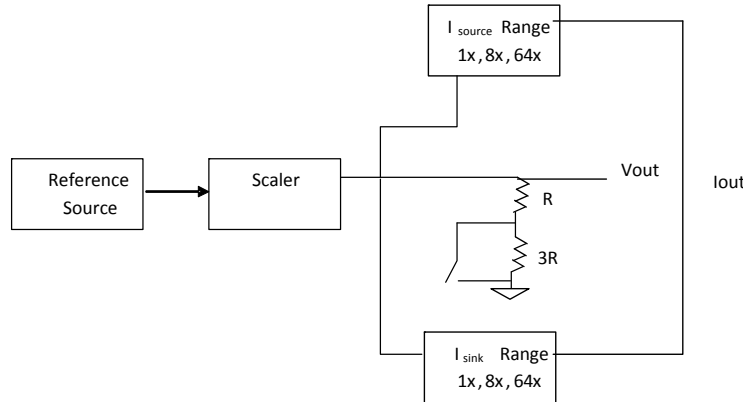
### 8.7 DAC

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  $\pm 25\%$  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 32  $\mu$ A, 0 to 256  $\mu$ A, 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 1.024 V and 0 to 4.096 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:

- SWD access
- FPB block for implementing breakpoints and code patches
- DWT block for implementing watchpoints, trigger resources, and system profiling
- ITM for support of printf-style debugging

PSoC devices include extensive support for programming, testing, debugging, and tracing both hardware and firmware. SWD supports all programming and debug features of the device. The SWV provides trace output from the DWT and ITM.

For more information on PSoC 5 programming, refer to the application note [AN64359 - In-System Programming for PSoC<sup>®</sup> 5](#).

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 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 reen able 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. 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.

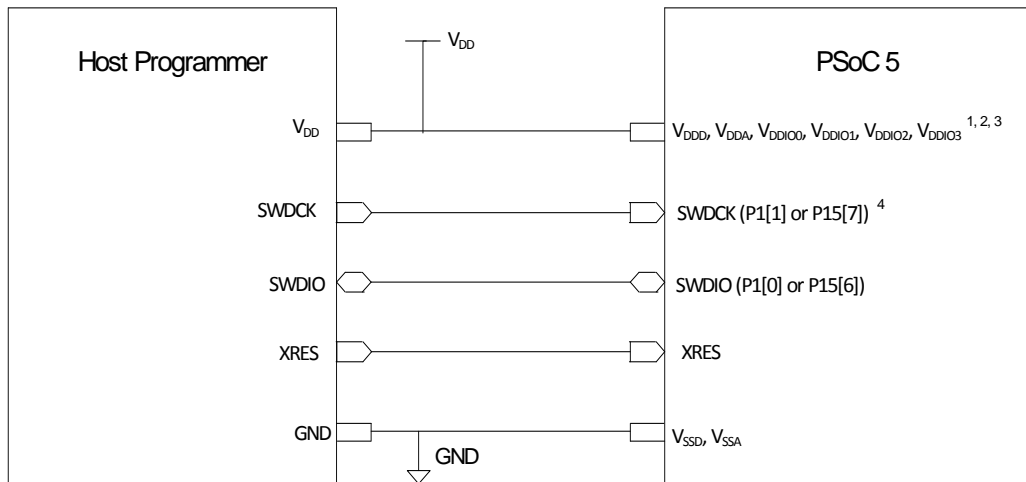
### 9.1 Debug Port Acquisition

Prior to programming or debugging, the debug port must be acquired. There is a time window after reset within which the Port Acquire must be completed. This window is initially 8  $\mu$ s; if eight clocks are detected on the SWDCK line within the 8  $\mu$ s period, the time window will then be extended to 400  $\mu$ s to complete the port acquire operation. The port acquire key must be transmitted over one of the two SWD pin pairs; see [SWD Interface](#) on page 49. For a detailed description of the acquire key sequence, refer to the Technical Reference Manual.

### 9.2 SWD Interface

SWD uses two pins, either two port 1 pins 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. SWD is used for debugging or for programming the flash memory. In addition, the SWD interface supports the SWV trace output. The SWD interface also includes the SWV interface, see ["SWV Interface"](#) on page 51. When using the SWD/SWV pins as standard GPIO, make sure that the GPIO functionality and PCB circuits do not interfere with SWD/SWV use. The SWV trace output is automatically activated whenever the SWD is activated.

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



- <sup>1</sup> The voltage levels of the Host Programmer and the PSoC 5 voltage domains involved in programming should be the same. XRES pin is powered by  $V_{DDIO1}$ . The USB SWD pins are powered by  $V_{DD}$ . So for programming using the USB SWD pins with XRES pin, the  $V_{DD}$ ,  $V_{DDIO1}$  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 ( $V_{DD}$ ,  $V_{DDA}$ ,  $V_{DDIO0}$ ,  $V_{DDIO2}$ ,  $V_{DDIO3}$ ) need not be at the same voltage level as host Programmer.
- <sup>2</sup>  $V_{DDA}$  must be greater than or equal to all other power supplies ( $V_{DD}$ ,  $V_{DDIO}$ 's) in PSoC 5.
- <sup>3</sup> For Power cycle mode Programming, XRES pin is not required. But the Host programmer must have the capability to toggle power ( $V_{DD}$ ,  $V_{DDA}$ , All  $V_{DDIO}$ '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,  $V_{DDA}$  must be greater than or equal to all other supplies.
- <sup>4</sup> When USB SWD pins are used for Programming, the P1[1] SWDCK pin must be externally connected to Ground using external pull-down resistor (around 100 K resistor). This is required for P15[7] SWDCK signal to be seen by PSoC 5's internal logic.

### 9.3 Debug Features

The CY8C52 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

### 9.4 Trace Features

The following trace features are supported:

- Data watchpoint on access to data address, address range, or data value
- Software event monitoring, “printf-style” debugging

### 9.5 SWV Interface

The SWV interface provides trace data to a debug host via the Cypress MiniProg3 or an external trace port analyzer.

### 9.6 Programming Features

The SWD interface provides 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 5 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 WOL must be programmed at  $V_{DD} \leq 3.3$  V.

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 16). 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 data sheets. 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.

## 10. Development Support

The CY8C52 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](http://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 CY8C52 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 data sheets:** The flexibility of PSoC allows the creation of new peripherals (components) long after the device has gone into production. Component data sheets 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](http://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 CY8C52 family is part of a development tool ecosystem. Visit us at [www.cypress.com/go/psoccreator](http://www.cypress.com/go/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  $-40\text{ °C} \leq T_A \leq 85\text{ °C}$  and  $T_J \leq 100\text{ °C}$ , except where noted. Specifications are valid for 2.7 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 data sheets for full AC/DC specifications of individual functions. See the “[Example Peripherals](#)” section on page 31 for further explanation of PSoC Creator components.

### 11.1 Absolute Maximum Ratings

**Table 11-1. Absolute Maximum Ratings DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
$T_{STG}$	Storage temperature	Recommended storage temperature is $+25\text{ °C} \pm 25\text{ °C}$ . Extended duration storage temperatures above $85\text{ °C}$ degrade reliability.	-55	25	100	°C
$V_{DDA}$	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
$V_{GPIO}^{[11]}$	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		-	-	20	mA
LU	Latch up current <sup>[12]</sup>		-140	-	140	mA
$ESD_{HBM}$	Electrostatic discharge voltage	Human body model	500	-	-	V
$ESD_{CDM}$	Electrostatic discharge voltage	Charge device model	500	-	-	V

**Note** Usage above the absolute maximum conditions listed in [Table 11-1](#) may cause permanent damage to the device. Exposure to maximum conditions for extended periods of time may affect device reliability. When used below maximum conditions but above normal operating conditions the device may not operate to specification.

**Notes**

- 11. The  $V_{DDIO}$  supply voltage must be greater than the maximum analog voltage on the associated GPIO pins. Maximum analog voltage on GPIO pin  $\leq V_{DDIO} \leq V_{DDA}$
- 12. Meets or exceeds JEDEC Spec EIA/JESD78 IC Latch-up Test.

**11.2 Device Level Specifications**

Specifications are valid for  $-40\text{ °C} \leq T_A \leq 85\text{ °C}$  and  $T_J \leq 100\text{ °C}$ , except where noted. Specifications are valid for 2.7 V to 5.5 V, except where noted.

11.2.1 Device Level Specifications

**Table 11-2. DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units	
V <sub>DDA</sub>	Analog supply voltage and input to analog core regulator	Analog core regulator enabled	2.7	–	5.5	V	
V <sub>DDA</sub>	Analog supply voltage, analog regulator bypassed	Analog core regulator disabled	1.71	1.8	1.89	V	
V <sub>DDD</sub>	Digital supply voltage relative to V <sub>SSD</sub>	Digital core regulator enabled	2.7	–	V <sub>DDA</sub> <sup>[13]</sup>	V	
V <sub>DDD</sub>	Digital supply voltage, digital regulator bypassed	Digital core regulator disabled	2.7	1.8	1.89	V	
V <sub>DDIO</sub> <sup>[14]</sup>	I/O supply voltage relative to V <sub>SSIO</sub>		1.71	–	V <sub>DDA</sub> <sup>[13]</sup>	V	
V <sub>CCA</sub>	Direct analog core voltage input (Analog regulator bypass)	Analog core regulator disabled	1.71	1.8	1.89	V	
V <sub>CCD</sub>	Direct digital core voltage input (Digital regulator bypass)	Digital core regulator disabled	1.71	1.8	1.89	V	
I <sub>DD</sub> <sup>[15]</sup>	<b>Active Mode, V<sub>DD</sub> = 2.7 V–5.5 V</b>						
	Execute from flash cache, see <a href="#">Cache Controller</a> on page 11 and <a href="#">Flash Program Memory</a> on page 16	CPU at 6 MHz	T = –40 °C	–	–	–	mA
			T = 25 °C	–	5	–	mA
			T = 85 °C	–	–	–	mA
	<b>Sleep Mode<sup>[16]</sup></b>						
	CPU = OFF RTC = ON (= ECO32K ON, in low power mode) Sleep timer = ON (= ILO ON at 1 kHz) <sup>[16]</sup> WDT = OFF POR = ON Boost = OFF SIO pins in single ended input, unregulated output mode	V <sub>DD</sub> = V <sub>DDIO</sub> = 4.5–5.5 V	T = –40 °C	–	–	–	µA
			T = 25 °C	–	–	–	µA
			T = 85 °C	–	–	–	µA
		V <sub>DD</sub> = V <sub>DDIO</sub> = 2.7–3.6 V	T = –40 °C	–	–	–	µA
			T = 25 °C	–	3	–	µA
			T = 85 °C	–	–	–	µA
	<b>Hibernate Mode</b>						
	Hibernate mode current All regulators and oscillators off. SRAM retention GPIO interrupts are active Boost = OFF SIO pins in single ended input, unregulated output mode	V <sub>DD</sub> = V <sub>DDIO</sub> = 4.5–5.5 V	T = –40 °C	–	–	–	nA
T = 25 °C			–	–	–	nA	
T = 85 °C			–	–	–	nA	
V <sub>DD</sub> = V <sub>DDIO</sub> = 2.7–3.6 V		T = –40 °C	–	–	–	nA	
		T = 25 °C	–	1000	–	nA	
		T = 85 °C	–	–	–	nA	

**Notes**

- 13. The power supplies can be brought up in any sequence however once stable V<sub>DDA</sub> must be greater than or equal to all other supplies.
- 14. The V<sub>DDIO</sub> supply voltage must be greater than the maximum analog voltage on the associated GPIO pins. Maximum analog voltage on GPIO pin  $\leq V_{DDIO} \leq V_{DDA}$ .
- 15. The current consumption of additional peripherals that are implemented only in programmed logic blocks can be found in their respective data sheets, 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 data sheet and component data sheets.
- 16. Sleep timer generates periodic interrupts to wake up the CPU. This specification applies only to those times that the CPU is off.

**Table 11-3. AC Specifications<sup>[17]</sup>**

Parameter	Description	Conditions	Min	Typ	Max	Units
F <sub>CPU</sub>	CPU frequency		DC	–	40.01	MHz
F <sub>BUSCLK</sub>	Bus frequency		DC	–	40.01	MHz
Svdd	V <sub>DD</sub> ramp rate		–	–	1	V/ns
T <sub>IO_INIT</sub>	Time from V <sub>DDD</sub> /V <sub>DDA</sub> /V <sub>CCD</sub> /V <sub>CCA</sub> ≥ IPOR to I/O ports set to their reset states		–	–	10	μs
T <sub>STARTUP</sub>	Time from V <sub>DDD</sub> /V <sub>DDA</sub> /V <sub>CCD</sub> /V <sub>CCA</sub> ≥ min operating voltage to CPU executing code at reset vector	No PLL used, IMO boot mode 12 MHz typ.	–	–	66	μs
T <sub>SLEEP</sub>	Wakeup from limited active mode – Application of non-LVD interrupt to beginning of execution of next CPU instruction		–	20	–	μs
T <sub>HIBERNATE</sub>	Wakeup form hibernate mode – Application of external interrupt to beginning of execution of next CPU instruction		–	–	100	μs

**Notes**

17. IBased on device characterization (Not production tested).

**11.3 Power Regulators**

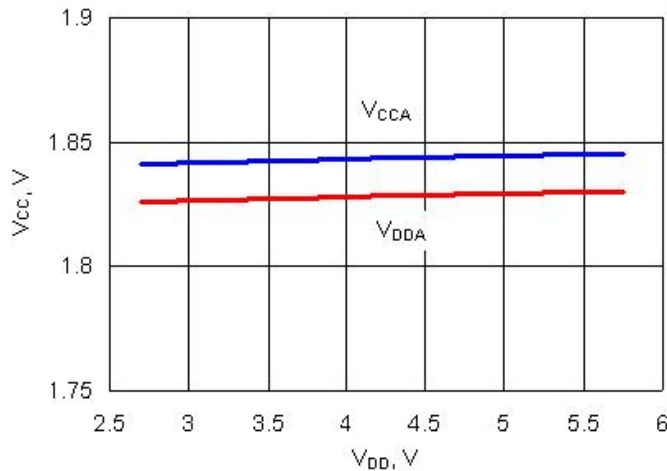
Specifications are valid for  $-40\text{ }^{\circ}\text{C} \leq T_A \leq 85\text{ }^{\circ}\text{C}$  and  $T_J \leq 100\text{ }^{\circ}\text{C}$ , except where noted. Specifications are valid for 2.7 V to 5.5 V, except where noted.

11.3.1 Digital Core Regulator

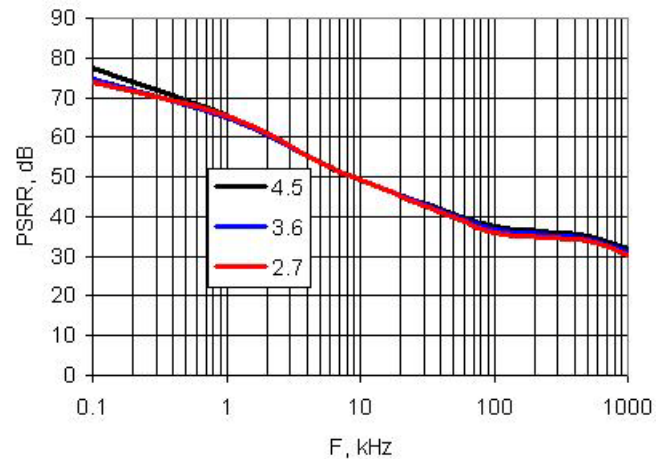
**Table 11-4. Digital Core Regulator DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
V <sub>DDD</sub>	Input voltage		2.7	–	5.5	V
V <sub>CCD</sub>	Output voltage		–	1.80	–	V
	Regulator output capacitor	±10%, X5R ceramic or better. The two V <sub>CCD</sub> pins must be shorted together, with as short a trace as possible, see <a href="#">Power System</a> on page 21	–	1	–	μF

**Figure 11-1. Regulators V<sub>CC</sub> vs V<sub>DD</sub>**



**Figure 11-2. Digital Regulator PSRR vs Frequency and V<sub>DD</sub>**

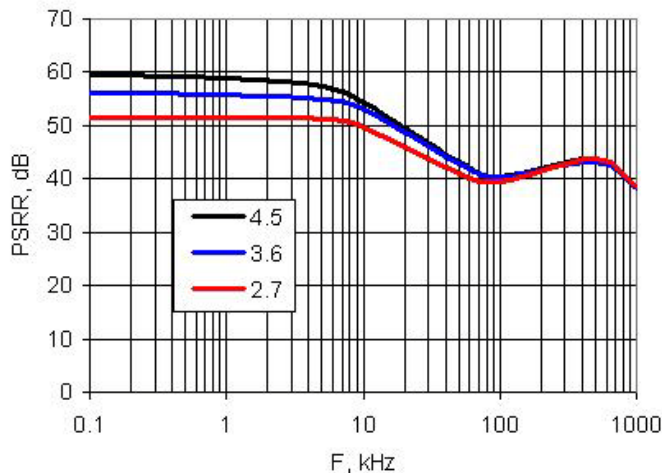


11.3.2 Analog Core Regulator

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

Parameter	Description	Conditions	Min	Typ	Max	Units
V <sub>DDA</sub>	Input voltage		2.7	–	5.5	V
V <sub>CCA</sub>	Output voltage		–	1.80	–	V
	Regulator output capacitor	±10%, X5R ceramic or better	–	1	–	μF

**Figure 11-3. Analog Regulator PSRR vs Frequency and V<sub>DD</sub>**





11.3.3 Inductive Boost Regulator.

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

Unless otherwise specified, operating conditions are:  $V_{BAT} = 2.4\text{ V}$ ,  $V_{OUT} = 3.3\text{ V}$ ,  $I_{OUT} = 40\text{ mA}$ ,  $F_{SW} = 400\text{ kHz}$ ,  $L_{BOOST} = 10\text{ }\mu\text{H}$ ,  $C_{BOOST} = 22\text{ }\mu\text{F} \parallel 0.1\text{ }\mu\text{F}$

Parameter	Description	Conditions	Min	Typ	Max	Units
$V_{BAT}$	Input voltage Includes startup		1.8	–	3.6	V
$I_{OUT}$	Load current <sup>[18, 19]</sup>	$V_{BAT} = 1.8 - 3.6\text{ V}$ , $V_{OUT} = 3.6 - 5.0\text{ V}$ , external diode	–	–	50	mA
		$V_{BAT} = 1.8 - 3.6\text{ V}$ , $V_{OUT} = 2.7 - 3.6\text{ V}$ , internal diode	–	–	75	mA
$I_{LPK}$	Inductor peak current		–	–	700	mA
$I_Q$	Quiescent current	Boost active mode	–	200	–	$\mu\text{A}$
		Boost standby mode, 32 khz external crystal oscillator, $I_{OUT} < 1\text{ }\mu\text{A}$	–	12	–	$\mu\text{A}$
$V_{OUT}$	Boost voltage range <sup>[20, 21]</sup>					
	3.0 V		2.85	3.00	3.15	V
	3.3 V		3.14	3.30	3.47	V
	3.6 V		3.42	3.60	3.78	V
	5.0 V	External diode required	4.75	5.00	5.25	V
$Reg_{LOAD}$	Load regulation		–	–	3.8	%
$Reg_{LINE}$	Line regulation		–	–	4.1	%
$\eta$	Efficiency	$L_{BOOST} = 10\text{ }\mu\text{H}$	70	85	–	%
		$L_{BOOST} = 22\text{ }\mu\text{H}$	82	90	–	%

**Table 11-7. Inductive Boost Regulator AC Specifications**

Unless otherwise specified, operating conditions are:  $V_{BAT} = 2.4\text{ V}$ ,  $V_{OUT} = 3.3\text{ V}$ ,  $I_{OUT} = 40\text{ mA}$ ,  $F_{SW} = 400\text{ kHz}$ ,  $L_{BOOST} = 10\text{ }\mu\text{H}$ ,  $C_{BOOST} = 22\text{ }\mu\text{F} \parallel 0.1\text{ }\mu\text{F}$ .

Parameter	Description	Conditions	Min	Typ	Max	Units
$V_{RIPPLE}$	Ripple voltage (peak-to-peak)	$F_{SW} = 400\text{ kHz}$ , $I_{OUT} = 10\text{ mA}$	–	–	100	mV
$F_{SW}$	Switching frequency		–	0.1, 0.4, or 2	–	MHz

**Notes**

- 18. For output voltages above 3.6 V, an external diode is required.
- 19. Maximum output current applies for output voltages  $\leq 4x$  input voltage.
- 20. Based on device characterization (Not production tested).
- 21. At boost frequency of 2 MHz,  $V_{OUT}$  is limited to  $2 \times V_{BAT}$ . At 400 kHz,  $V_{OUT}$  is limited to  $4 \times V_{BAT}$ .

**Table 11-8. Recommended External Components for Boost Circuit**

Parameter	Description	Conditions	Min	Typ	Max	Units
L <sub>BOOST</sub>	Boost inductor		4.7	10	47	μH
C <sub>BOOST</sub>	Filter capacitor <sup>[22]</sup>		10	22	47	μF
I <sub>F</sub>	External Schottky diode average forward current	External Schottky diode is required for V <sub>OUT</sub> > 3.6 V	1	–	–	A
V <sub>R</sub>			20	–	–	V

**11.4 Inputs and Outputs**

Specifications are valid for –40 °C ≤ T<sub>A</sub> ≤ 85 °C and T<sub>J</sub> ≤ 100 °C, except where noted. Specifications are valid for 2.7 V to 5.5 V, except where noted. Unless otherwise specified, all charts and graphs show typical values.

11.4.1 GPIO

**Table 11-9. GPIO DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
V <sub>IH</sub>	Input voltage high threshold	CMOS Input, PRT[x]CTL = 0	0.7 × V <sub>DDIO</sub>	–	–	V
V <sub>IL</sub>	Input voltage low threshold	CMOS Input, PRT[x]CTL = 0	–	–	0.3 × V <sub>DDIO</sub>	V
V <sub>IH</sub>	Input voltage high threshold	LVTTL Input, PRT[x]CTL = 1	2.0	–	–	V
V <sub>IL</sub>	Input voltage low threshold	LVTTL Input, PRT[x]CTL = 1	–	–	0.8	V
V <sub>OH</sub>	Output voltage high	I <sub>OH</sub> = 4 mA at 3.3 V <sub>DDIO</sub>	V <sub>DDIO</sub> – 0.6	–	–	V
V <sub>OL</sub>	Output voltage low	I <sub>OL</sub> = 8 mA at 3.3 V <sub>DDIO</sub>	–	–	0.6	V
R <sub>pullup</sub>	Pull up resistor		3.5	5.6	8.5	kΩ
R <sub>pulldown</sub>	Pull down resistor		3.5	5.6	8.5	kΩ
I <sub>IL</sub>	Input leakage current (absolute value) <sup>[23]</sup>	25 °C, V <sub>DDIO</sub> = 3.0 V	–	–	2	nA
C <sub>IN</sub>	Input capacitance <sup>[23]</sup>		–	–	7	pF
V <sub>H</sub>	Input voltage hysteresis (Schmitt-Trigger) <sup>[23]</sup>		–	40	–	mV
I <sub>diode</sub>	Current through protection diode to V <sub>DDIO</sub> and V <sub>SSIO</sub>		–	–	100	μA
R <sub>global</sub>	Resistance pin to analog global bus	25 °C, V <sub>DDIO</sub> = 3.0 V	–	320	–	Ω
R <sub>mux</sub>	Resistance pin to analog mux bus	25 °C, V <sub>DDIO</sub> = 3.0 V	–	220	–	Ω

**Note**

22. Based on device characterization (Not production tested).

Figure 11-4. GPIO Output High Voltage and Current

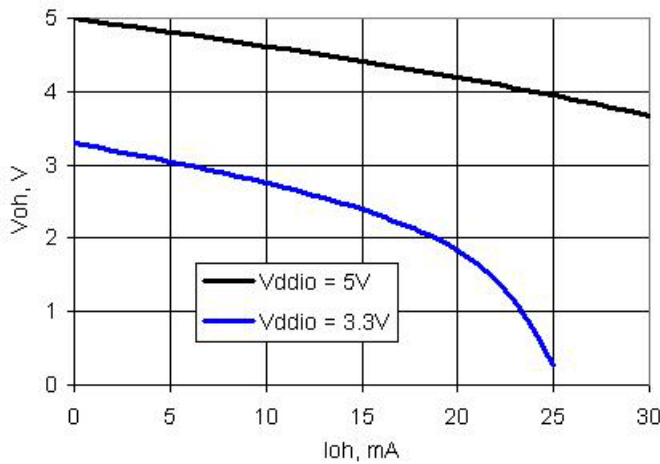


Figure 11-5. GPIO Output Low Voltage and Current

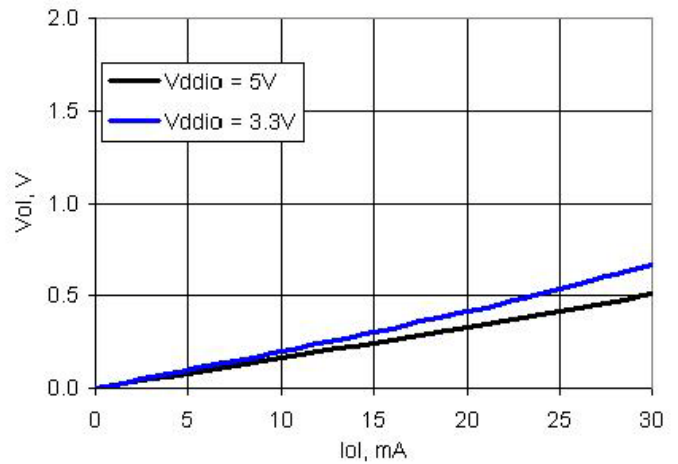


Table 11-10. GPIO AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
TriseF	Rise time in Fast Strong Mode <sup>[23]</sup>	3.3 V V <sub>DDIO</sub> Load = 25 pF	–	–	12	ns
TfallF	Fall time in Fast Strong Mode <sup>[23]</sup>	3.3 V V <sub>DDIO</sub> Load = 25 pF	–	–	12	ns
TriseS	Rise time in Slow Strong Mode <sup>[23]</sup>	3.3 V V <sub>DDIO</sub> Load = 25 pF	–	–	60	ns
TfallS	Fall time in Slow Strong Mode <sup>[23]</sup>	3.3 V V <sub>DDIO</sub> Load = 25 pF	–	–	60	ns
Fgpioout	GPIO output operating frequency					
	Fast strong drive mode	90/10% V <sub>DDIO</sub> into 25 pF	–	–	33	MHz
	3.3 V ≤ V <sub>DDIO</sub> ≤ 5.5 V, slow strong drive mode	90/10% V <sub>DDIO</sub> into 25 pF	–	–	7	MHz
	2.7 V ≤ V <sub>DDIO</sub> < 3.3 V, slow strong drive mode	90/10% V <sub>DDIO</sub> into 25 pF	–	–	3.5	MHz
Fgpioin	GPIO input operating frequency					
	2.7 V ≤ V <sub>DDIO</sub> ≤ 5.5 V	90/10% V <sub>DDIO</sub>	–	–	40	MHz

Figure 11-6. GPIO Output Rise and Fall Times, Fast Strong Mode, V<sub>DDIO</sub> = 3.3 V, 25 pF Load

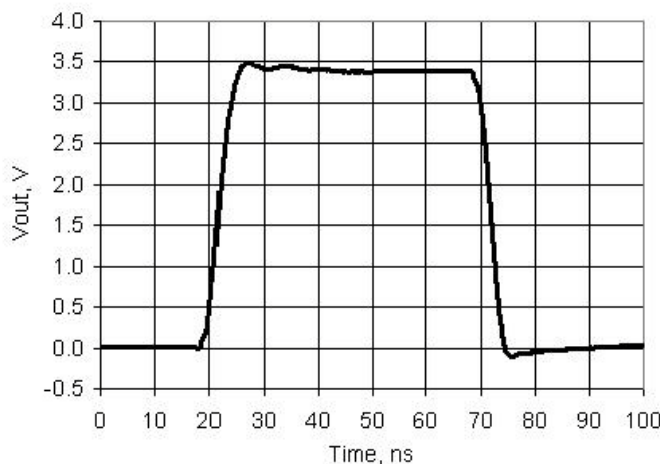
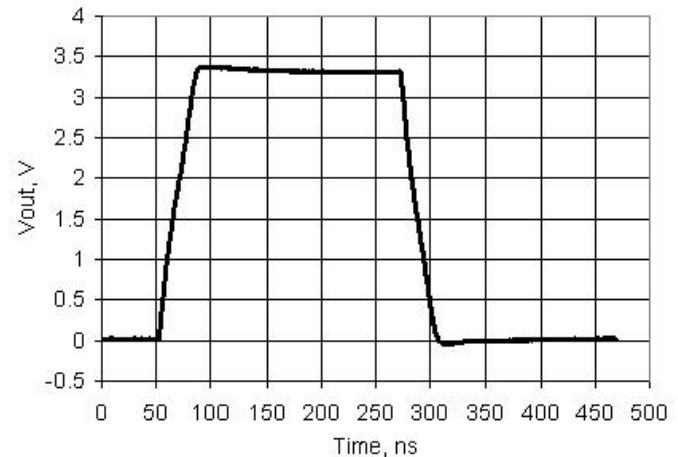


Figure 11-7. GPIO Output Rise and Fall Times, Slow Strong Mode, V<sub>DDIO</sub> = 3.3 V, 25 pF Load



Note

23. Based on device characterization (Not production tested).

11.4.2 SIO

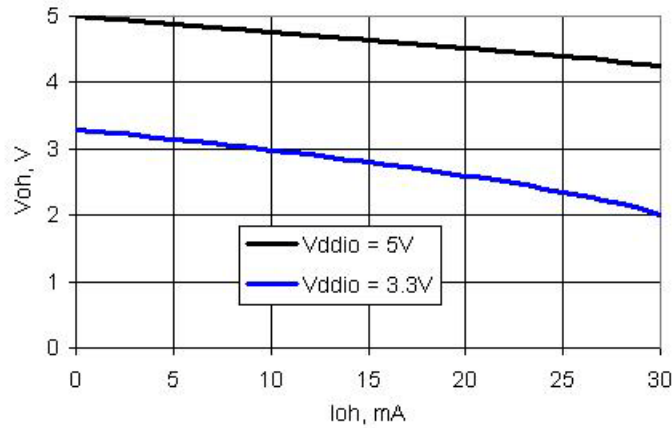
**Table 11-11. SIO DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
V <sub>inmax</sub>	Maximum input voltage	All allowed values of V <sub>ddio</sub> and V <sub>ddd</sub> , see Section 11.2.1	–	–	5.5	V
V <sub>inref</sub>	Input voltage reference (Differential input mode)		0.5	–	0.52 × V <sub>DDIO</sub>	V
V <sub>outref</sub>	Output voltage reference (Regulated output mode)					
		V <sub>DDIO</sub> > 3.7	1	–	V <sub>DDIO</sub> – 1	V
		V <sub>DDIO</sub> < 3.7	1	–	V <sub>DDIO</sub> – 0.5	V
V <sub>IH</sub>	Input voltage high threshold					
	GPIO mode	CMOS input	0.7 × V <sub>DDIO</sub>	–	–	V
	Differential input mode <sup>[24]</sup>	Hysteresis disabled	SIO_ref + 0.2	–	–	V
V <sub>IL</sub>	Input voltage low threshold					
	GPIO mode	CMOS input	–	–	0.3 × V <sub>DDIO</sub>	V
	Differential input mode <sup>[24]</sup>	Hysteresis disabled	–	–	SIO_ref – 0.2	V
V <sub>OH</sub>	Output voltage high					
	Unregulated mode	I <sub>OH</sub> = 4 mA, V <sub>DDIO</sub> = 3.3 V	V <sub>DDIO</sub> – 0.4	–	–	V
	Regulated mode <sup>[24]</sup>	I <sub>OH</sub> = 1 mA	SIO_ref – 0.65	–	SIO_ref + 0.2	V
	Regulated mode <sup>[24]</sup>	I <sub>OH</sub> = 0.1 mA	SIO_ref – 0.3	–	SIO_ref + 0.2	V
V <sub>OL</sub>	Output voltage low	V <sub>DDIO</sub> = 3.30 V, I <sub>OL</sub> = 25 mA	–	–	0.8	V
R <sub>pullup</sub>	Pull up resistor		3.5	5.6	8.5	kΩ
R <sub>pulldown</sub>	Pull down resistor		3.5	5.6	8.5	kΩ
I <sub>IL</sub>	Input leakage current (absolute value) <sup>[25]</sup>					
	V <sub>IH</sub> ≤ V <sub>ddsio</sub>	25 °C, V <sub>ddsio</sub> = 3.0 V, V <sub>IH</sub> = 3.0 V	–	–	14	nA
	V <sub>IH</sub> > V <sub>ddsio</sub>	25 °C, V <sub>ddsio</sub> = 0 V, V <sub>IH</sub> = 3.0 V	–	–	10	μA
C <sub>IN</sub>	Input Capacitance <sup>[25]</sup>		–	–	7	pF
V <sub>H</sub>	Input voltage hysteresis (Schmitt-Trigger) <sup>[25]</sup>					
		Single ended mode (GPIO mode)	–	40	–	mV
		Differential mode	–	35	–	mV
I <sub>diode</sub>	Current through protection diode to V <sub>SSIO</sub>		–	–	100	μA

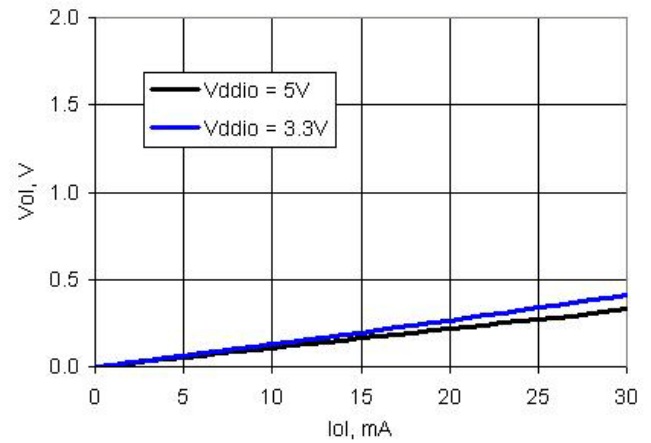
**Notes**

- 24. See Figure 6-9 on page 27 and Figure 6-12 on page 30 for more information on SIO reference.
- 25. Based on device characterization (Not production tested).

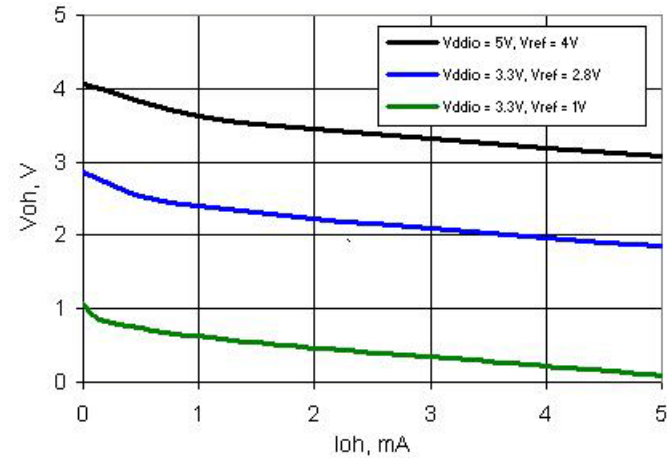
**Figure 11-8. SIO Output High Voltage and Current, Unregulated Mode**



**Figure 11-9. SIO Output Low Voltage and Current, Unregulated Mode**



**Figure 11-10. SIO Output High Voltage and Current, Regulated Mode**



**Table 11-12. SIO AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
TriseF	Rise time in Fast Strong Mode (90/10%) <sup>[26]</sup>	Cl <sub>oad</sub> = 25 pF, V <sub>DDIO</sub> = 3.3 V	–	–	12	ns
TfallF	Fall time in Fast Strong Mode (90/10%) <sup>[26]</sup>	Cl <sub>oad</sub> = 25 pF, V <sub>DDIO</sub> = 3.3 V	–	–	12	ns
TriseS	Rise time in Slow Strong Mode (90/10%) <sup>[26]</sup>	Cl <sub>oad</sub> = 25 pF, V <sub>DDIO</sub> = 3.0 V	–	–	75	ns
TfallS	Fall time in Slow Strong Mode (90/10%) <sup>[26]</sup>	Cl <sub>oad</sub> = 25 pF, V <sub>DDIO</sub> = 3.0 V	–	–	60	ns

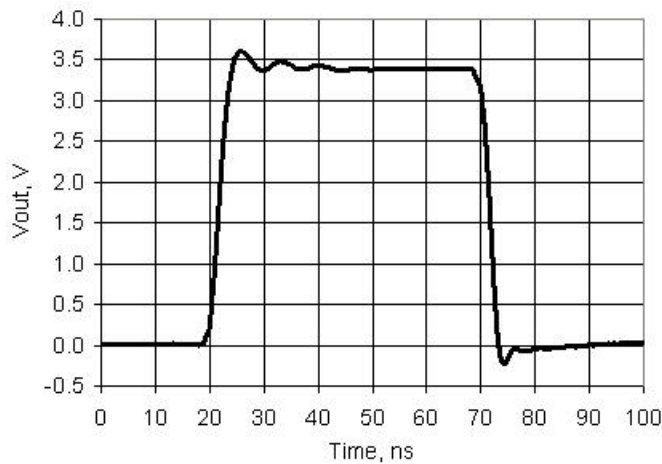
**Note**

26. Based on device characterization (Not production tested).

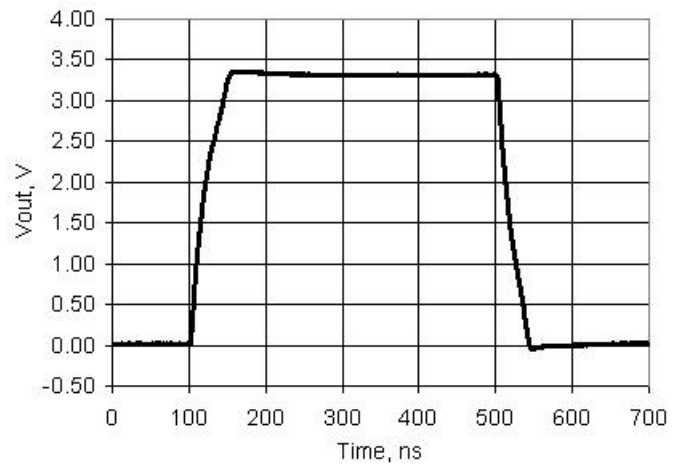
**Table 11-12. SIO AC Specifications (continued)**

Parameter	Description	Conditions	Min	Typ	Max	Units
F <sub>sioout</sub>	SIO output operating frequency					
	Unregulated output (GPIO) mode, fast strong drive mode	90/10% V <sub>DDIO</sub> into 25 pF	–	–	33	MHz
	3.3 V < V <sub>DDIO</sub> < 5.5 V, Unregulated output (GPIO) mode, slow strong drive mode	90/10% V <sub>DDIO</sub> into 25 pF	–	–	5	MHz
	2.7 V < V <sub>DDIO</sub> < 3.3 V, Unregulated output (GPIO) mode, slow strong drive mode	90/10% V <sub>DDIO</sub> into 25 pF	–	–	4	MHz
	Regulated output mode, fast strong drive mode	Output continuously switching into 25 pF	–	–	20	MHz
	Regulated output mode, slow strong drive mode	Output continuously switching into 25 pF	–	–	2.5	MHz
F <sub>sioin</sub>	SIO input operating frequency	90/10% V <sub>DDIO</sub>	–	–	40	MHz

**Figure 11-11. SIO Output Rise and Fall Times, Fast Strong Mode, V<sub>DDIO</sub> = 3.3 V, 25 pF Load**



**Figure 11-12. SIO Output Rise and Fall Times, Slow Strong Mode, V<sub>DDIO</sub> = 3.3 V, 25 pF Load**



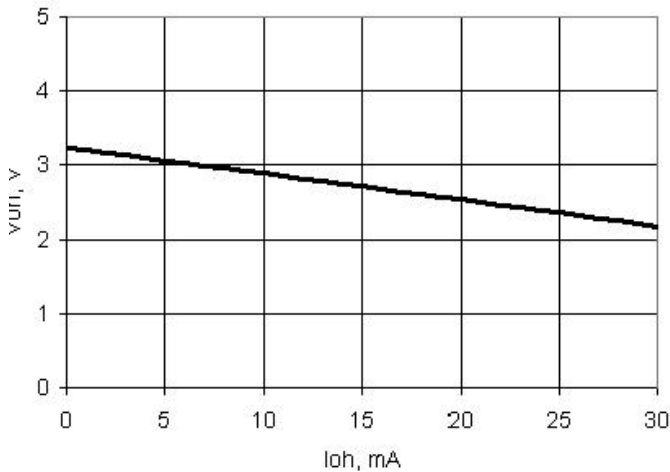
11.4.3 USBIO

For operation in GPIO mode, the standard range for  $V_{DD}$  applies, see [Device Level Specifications](#) on page 54.

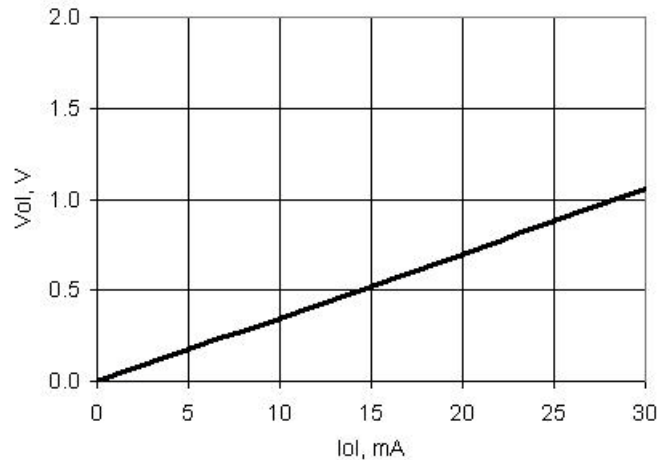
**Table 11-13. USBIO DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
Rusbi	USB D+ pull-up resistance	With idle bus	0.900	–	1.575	k $\Omega$
Rusba	USB D+ pull-up resistance	While receiving traffic	1.425	–	3.090	k $\Omega$
Vohusb	Static output high	15 k $\Omega$ $\pm$ 5% to Vss, internal pull-up enabled	2.8	–	3.6	V
Volusb	Static output low	15 k $\Omega$ $\pm$ 5% to Vss, internal pull-up enabled	–	–	0.3	V
Vihgpio	Input voltage high, GPIO mode	$V_{DD} \geq 3$ V	2	–	–	V
Vilgpio	Input voltage low, GPIO mode	$V_{DD} \geq 3$ V	–	–	0.8	V
Vohgpio	Output voltage high, GPIO mode	$I_{OH} = 4$ mA, $V_{DD} \geq 3$ V	2.4	–	–	V
Volgpio	Output voltage low, GPIO mode	$I_{OL} = 4$ mA, $V_{DD} \geq 3$ 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	In PS/2 mode, with PS/2 pull-up enabled	3	–	7	k $\Omega$
Rext	External USB series resistor	In series with each USB pin	21.78 (–1%)	22	22.22 (+1%)	$\Omega$
Zo	USB driver output impedance	Including Rext	28	–	44	$\Omega$
C <sub>IN</sub>	USB transceiver input capacitance		–	–	20	pF
I <sub>IL</sub>	Input leakage current (absolute value)	25 °C, $V_{DD} = 3.0$ V	–	–	2	nA

**Figure 11-13. USBIO Output High Voltage and Current, GPIO Mode**



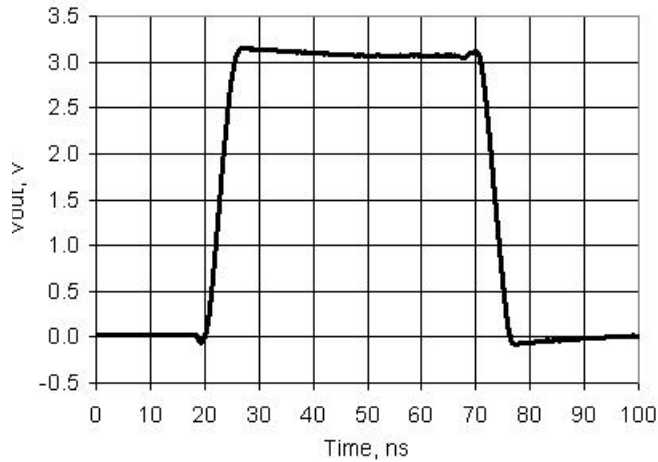
**Figure 11-14. USBIO Output Low Voltage and Current, GPIO Mode**



**Table 11-14. USBIO AC Specifications**

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	$3\text{ V} \leq V_{\text{DDD}} \leq 5.5\text{ V}$	–	–	20	MHz
		$V_{\text{DDD}} = 2.7\text{ V}$	–	–	6	MHz
Tr_gpio	Rise time, GPIO mode, 10%/90% $V_{\text{DDD}}$	$V_{\text{DDD}} > 3\text{ V}$ , 25 pF load	–	–	12	ns
		$V_{\text{DDD}} = 2.7\text{ V}$ , 25 pF load	–	–	40	ns
Tf_gpio	Fall time, GPIO mode, 90%/10% $V_{\text{DDD}}$	$V_{\text{DDD}} > 3\text{ V}$ , 25 pF load	–	–	12	ns
		$V_{\text{DDD}} = 2.7\text{ V}$ , 25 pF load	–	–	40	ns

**Figure 11-15. USBIO Output Rise and Fall Times, GPIO Mode,  $V_{\text{DDD}} = 3.3\text{ V}$ , 25 pF Load**





**Table 11-15. USB Driver AC Specifications**

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 <sub>USB_5</sub> , V <sub>USB_3.3</sub> , see <a href="#">USB DC Specifications</a> on page 79	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 <sub>IH</sub>	Input voltage high threshold		0.7 × V <sub>DDIO</sub>	–	–	V
V <sub>IL</sub>	Input voltage low threshold		–	–	0.3 × V <sub>DDIO</sub>	V
Rpullup	Pull up resistor		3.5	5.6	8.5	kΩ
C <sub>IN</sub>	Input capacitance <sup>[27]</sup>		–	3		pF
V <sub>H</sub>	Input voltage hysteresis (Schmitt-Trigger) <sup>[27]</sup>		–	100	–	mV
I <sub>diode</sub>	Current through protection diode to V <sub>DDIO</sub> and V <sub>SSIO</sub>		–	–	100	μA

**Table 11-17. XRES AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
T <sub>RESET</sub>	Reset pulse width		1	–	–	μs

**Note**

27. Based on device characterization (Not production tested).

**11.5 Analog Peripherals**

Specifications are valid for  $-40\text{ }^{\circ}\text{C} \leq T_A \leq 85\text{ }^{\circ}\text{C}$  and  $T_J \leq 100\text{ }^{\circ}\text{C}$ , except where noted. Specifications are valid for 2.7 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 <sub>REF</sub>	Precision reference voltage	Initial trimming	1.017 (-0.7%)	1.024	1.033 (+0.9%)	V
	Temperature drift <sup>[28]</sup>		-	-	20	ppm/ $^{\circ}\text{C}$
	Long term drift		-	100	-	ppm/Khr
	Thermal cycling drift (stability) <sup>[28]</sup>		-	100	-	ppm

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 <sup>[30]</sup>		Yes	-	-	
Ge	Gain error		-	-	$\pm 0.1$	%
V <sub>OS</sub>	Input offset voltage		-	-	$\pm 0.2$	mV
I <sub>DD</sub>	Current consumption		-	-	1	mA
	Input voltage range – single-ended <sup>[30]</sup>		V <sub>SSA</sub>	-	V <sub>DDA</sub>	V
	Input voltage range – differential <sup>[30]</sup>		V <sub>SSA</sub>	-	V <sub>DDA</sub>	V
PSRR	Power supply rejection ratio <sup>[30]</sup>		70	-	-	dB
CMRR	Common mode rejection ratio		35	-	-	dB
INL	Integral non linearity <sup>[30]</sup>	Internal reference	-	-	$\pm 2$	LSB
DNL	Differential non linearity <sup>[30]</sup>	Internal reference	-	-	$\pm 2$	LSB
R <sub>IN</sub>	Input resistance <sup>[30]</sup>		-	TBD	-	K $\Omega$

**Table 11-20. SAR ADC AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Sample rate <sup>[30]</sup>		-	-	770	ksps
	Startup time <sup>[30]</sup>		-	-	10	$\mu\text{s}$
SINAD	Signal-to-noise ratio <sup>[30]</sup>		44	-	-	dB
THD	Total harmonic distortion <sup>[30]</sup>		-	-	0.02	%

**Note**

28. Based on device characterization (Not production tested).

11.5.3 Analog Globals

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

Parameter	Description	Conditions	Min	Typ	Max	Units
Rppag	Resistance pin-to-pin through analog global	V <sub>DDA</sub> = 3.0 V	–	939	1461	Ω
Rppmuxbus	Resistance pin-to-pin through analog mux bus	V <sub>DDA</sub> = 3.0 V	–	721	1135	Ω

11.5.4 Comparator

**Table 11-22. Comparator DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
V <sub>OS</sub>	Input offset voltage in fast mode	Factory trim, V <sub>in</sub> ≥ 0.5 V	–		10	mV
	Input offset voltage in slow mode	Factory trim, V <sub>in</sub> ≥ 0.5 V	–		9	mV
V <sub>OS</sub>	Input offset voltage in fast mode <sup>[29]</sup>	Custom trim	–	–	4	mV
	Input offset voltage in slow mode <sup>[29]</sup>	Custom trim	–	–	4	mV
V <sub>OS</sub>	Input offset voltage in ultra low power mode		–	±12	–	mV
V <sub>HYST</sub>	Hysteresis	Hysteresis enable mode	–	10	32	mV
V <sub>ICM</sub>	Input common mode voltage	High current / fast mode	V <sub>SSA</sub>	–	V <sub>DDA</sub> – 0.1	V
		Low current / slow mode	V <sub>SSA</sub>	–	V <sub>DDA</sub>	V
		Ultra low power mode	V <sub>SSA</sub>	–	V <sub>DDA</sub> – 0.9	
CMRR	Common mode rejection ratio		–	50	–	dB
I <sub>CMP</sub>	High current mode/fast mode <sup>[30]</sup>		–	–	400	μA
	Low current mode/slow mode <sup>[30]</sup>		–	–	100	μA
	Ultra low power mode <sup>[30]</sup>		–	6	–	μA

**Table 11-23. Comparator AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
T <sub>RESP</sub>	Response time, high current mode <sup>[30]</sup>	50 mV overdrive, measured pin-to-pin	–	75	110	ns
	Response time, low current mode <sup>[30]</sup>	50 mV overdrive, measured pin-to-pin	–	155	200	ns
	Response time, ultra low power mode <sup>[30]</sup>	50 mV overdrive, measured pin-to-pin	–	55	–	μs

**Notes**

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

11.5.5 Current Digital-to-analog Converter (IDAC)

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 11-24. IDAC DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Resolution		–	–	8	bits
I <sub>OUT</sub>	Output current at code = 255	Range = 2.04 mA, code = 255, Rload = 600 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	±2.5	LSB
Eg	Gain error		–	–	±5	%
TC_Eg	Temperature coefficient of gain error	Range = 2.04 mA	–	–	0.04	% / °C
		Range = 255 μA	–	–	0.04	% / °C
		Range = 31.875 μA	–	–	0.05	% / °C
INL	Integral nonlinearity	Range = 255 μA, Codes 8 – 255, Rload = 600 Ω, Cload = 15 pF	–	–	±3	LSB
DNL	Differential nonlinearity, non-monotonic	Range = 255 μA, Rload = 600 Ω, Cload = 15 pF	–	–	±1.6	LSB
Vcompliance	Dropout voltage, source or sink mode	Voltage headroom at max current, Rload to Vdda or Rload to Vssa, Vdiff from Vdda	1	–	–	V

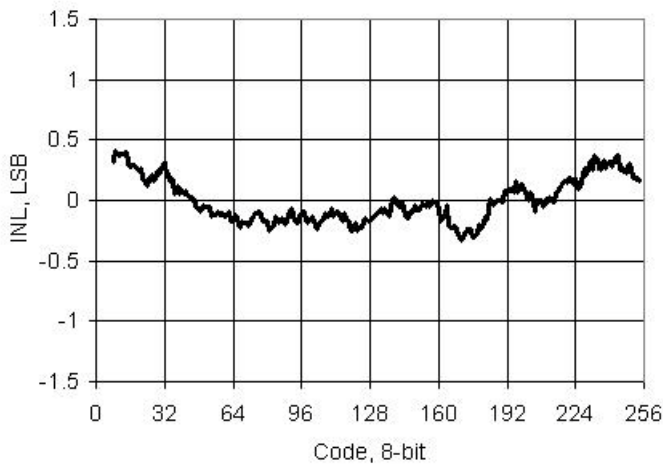
**Note**

31. Based on device characterization (Not production tested).

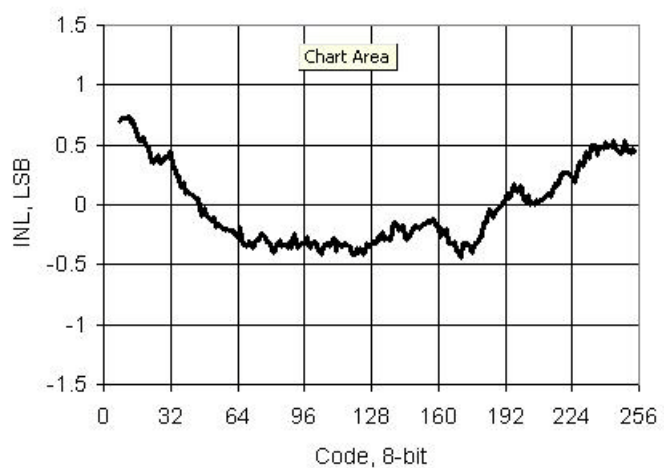
**Table 11-24. IDAC DC Specifications (continued)**

Parameter	Description	Conditions	Min	Typ	Max	Units
I <sub>DD</sub>	Operating current, code = 0	Slow mode, source mode, range = 31.875 $\mu$ A	–	44	100	$\mu$ A
		Slow mode, source mode, range = 255 $\mu$ A,	–	33	100	$\mu$ A
		Slow mode, source mode, range = 2.04 mA	–	33	100	$\mu$ A
		Slow mode, sink mode, range = 31.875 $\mu$ A	–	36	100	$\mu$ A
		Slow mode, sink mode, range = 255 $\mu$ A	–	33	100	$\mu$ A
		Slow mode, sink mode, range = 2.04 mA	–	33	100	$\mu$ A
		Fast mode, source mode, range = 31.875 $\mu$ A	–	310	500	$\mu$ A
		Fast mode, source mode, range = 255 $\mu$ A	–	305	500	$\mu$ A
		Fast mode, source mode, range = 2.04 mA	–	305	500	$\mu$ A
		Fast mode, sink mode, range = 31.875 $\mu$ A	–	310	500	$\mu$ A
		Fast mode, sink mode, range = 255 $\mu$ A	–	300	500	$\mu$ A
		Fast mode, sink mode, range = 2.04 mA	–	300	500	$\mu$ A

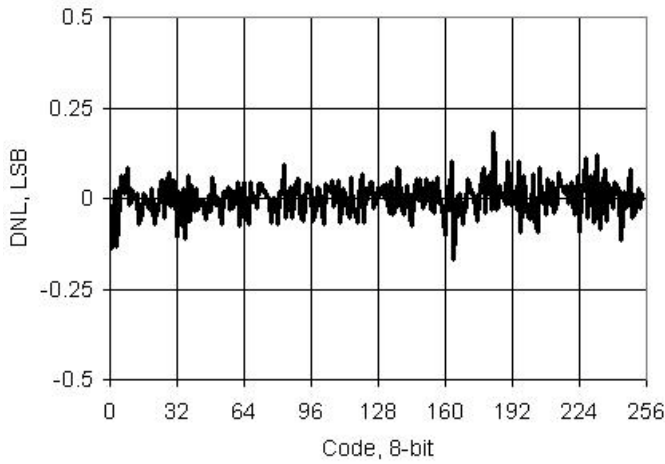
**Figure 11-16. IDAC INL vs Input Code, Range = 255  $\mu$ A, Source Mode**



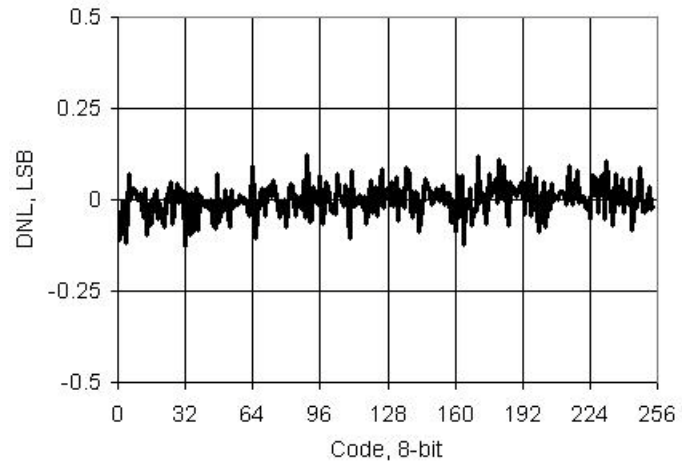
**Figure 11-17. IDAC INL vs Input Code, Range = 255  $\mu$ A, Sink Mode**



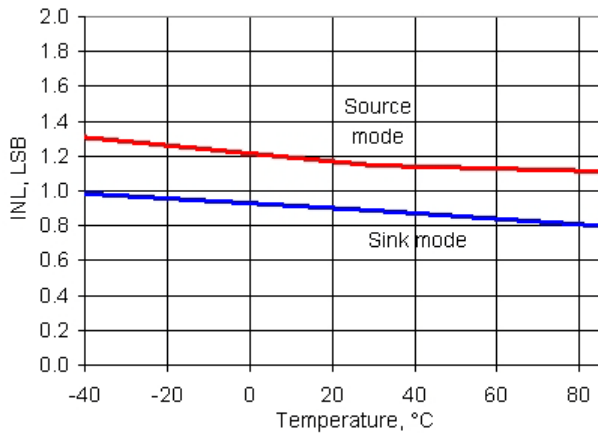
**Figure 11-18. IDAC DNL vs Input Code, Range = 255  $\mu$ A, Source Mode**



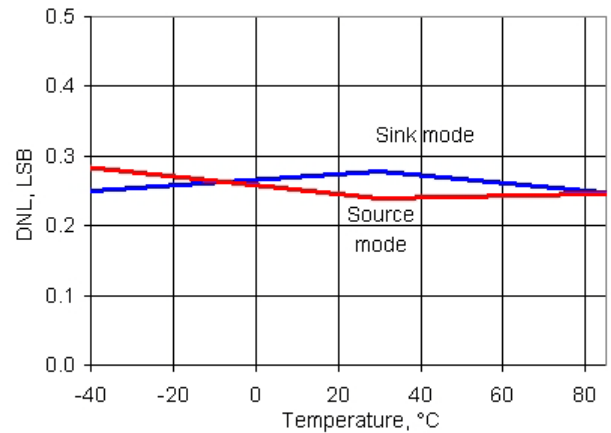
**Figure 11-19. IDAC DNL vs Input Code, Range = 255  $\mu$ A, Sink Mode**



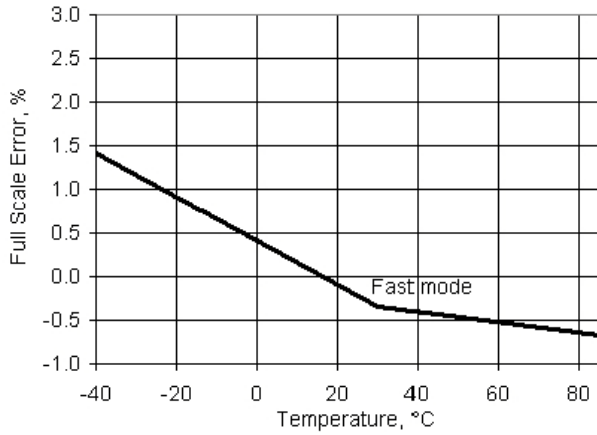
**Figure 11-20. IDAC INL vs Temperature, Range = 255  $\mu$ A, Fast Mode**



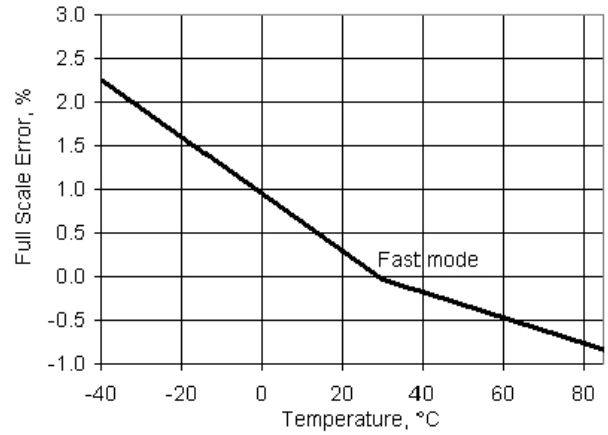
**Figure 11-21. IDAC DNL vs Temperature, Range = 255  $\mu$ A, Fast Mode**



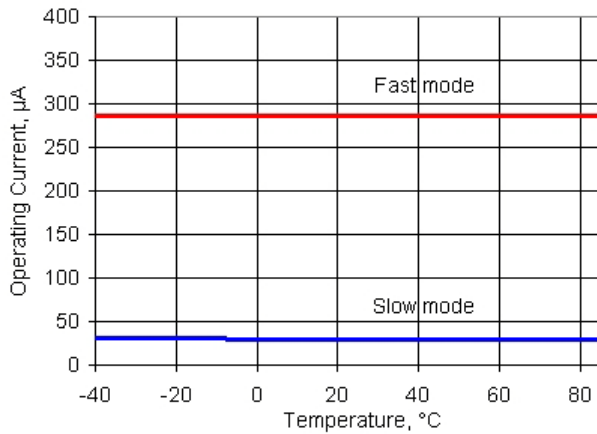
**Figure 11-22. IDAC Full Scale Error vs Temperature, Range = 255  $\mu$ A, Source Mode**



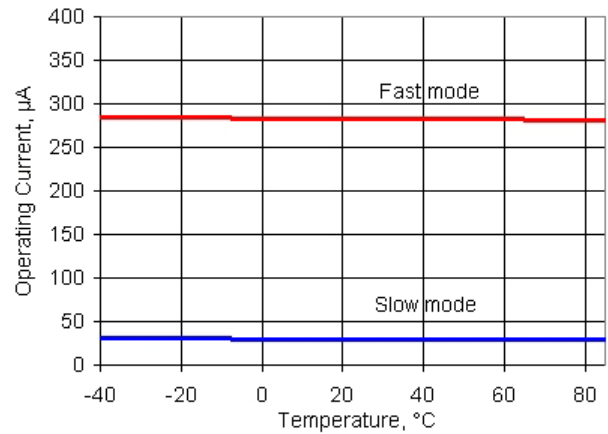
**Figure 11-23. IDAC Full Scale Error vs Temperature, Range = 255  $\mu$ A, Sink Mode**



**Figure 11-24. IDAC Operating Current vs Temperature, Range = 255  $\mu$ A, Code = 0, Source Mode**



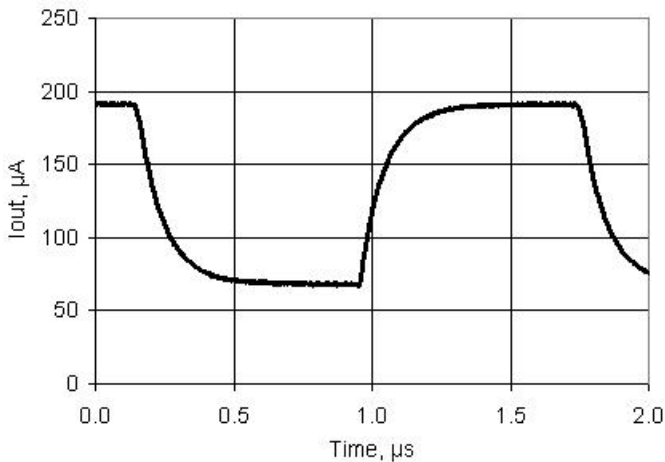
**Figure 11-25. IDAC Operating Current vs Temperature, Range = 255  $\mu$ A, Code = 0, Sink Mode**



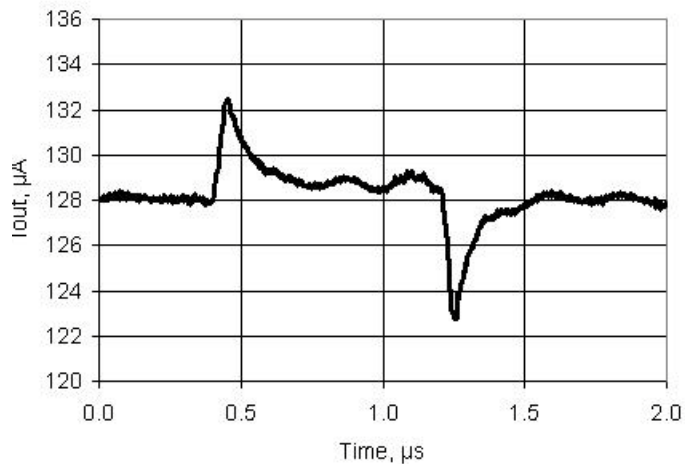
**Table 11-25. IDAC AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
$F_{DAC}$	Update rate		–	–	8	Msp/s
$T_{SETTLE}$	Settling time to 0.5 LSB	Range = 31.875 $\mu$ A or 255 $\mu$ A, full scale transition, fast mode, 600 $\Omega$ 15-pF load	–	–	125	ns

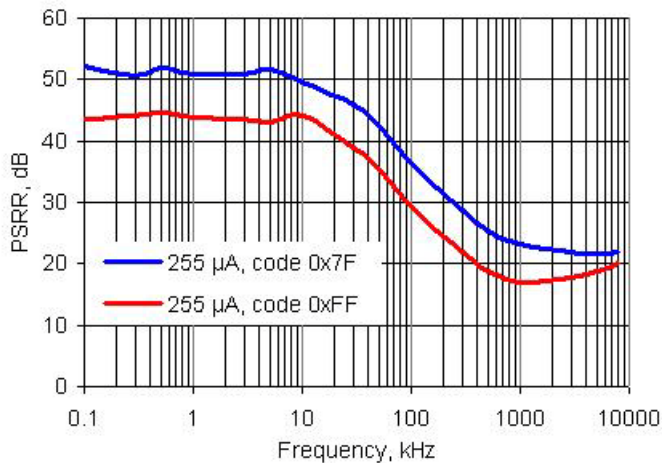
**Figure 11-26. IDAC Step Response, Codes 0x40 - 0xC0, 255  $\mu$ A Mode, Source Mode, Fast Mode, V<sub>dda</sub> = 5 V**



**Figure 11-27. IDAC Glitch Response, Codes 0x7F - 0x80, 255  $\mu$ A Mode, Source Mode, Fast Mode, V<sub>dda</sub> = 5 V**



**Figure 11-28. IDAC PSRR vs Frequency**





11.5.6 Voltage Digital to Analog Converter (VDAC)

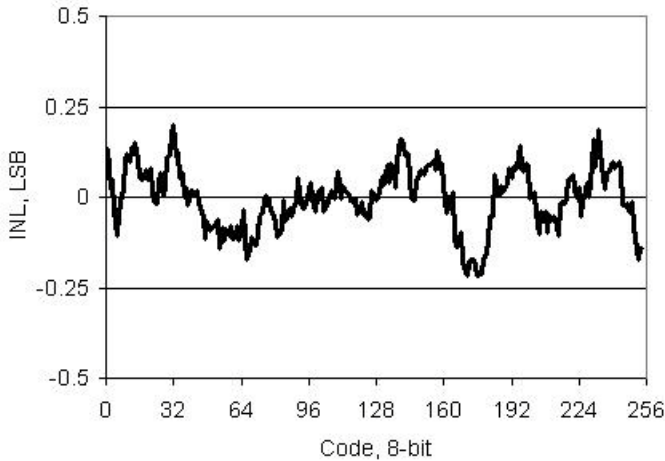
See the VDAC component data sheet in PSoC Creator for full electrical specifications and APIs.

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

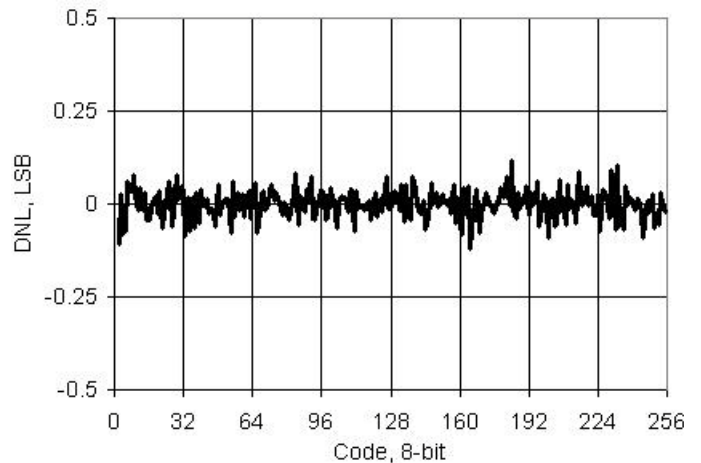
**Table 11-26. VDAC DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Resolution		–	8	–	bits
INL1	Integral nonlinearity	1 V scale	–	±2.1	±2.5	LSB
DNL1	Differential nonlinearity	1 V scale	–	±0.3	±1	LSB
Rout	Output resistance	1 V scale	–	4	–	kΩ
		4 V scale	–	16	–	kΩ
V <sub>OUT</sub>	Output voltage range, code = 255	1 V scale	–	1	–	V
		4 V scale, V <sub>dda</sub> = 5 V	–	4	–	V
	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	Slow mode	–	–	100	μA
		Fast mode	–	–	500	μA

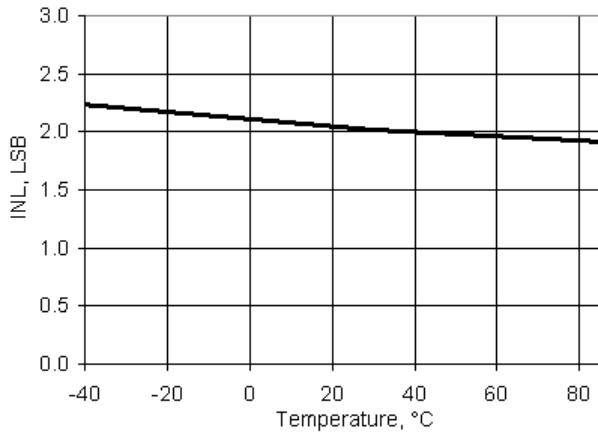
**Figure 11-29. VDAC INL vs Input Code, 1 V Mode**



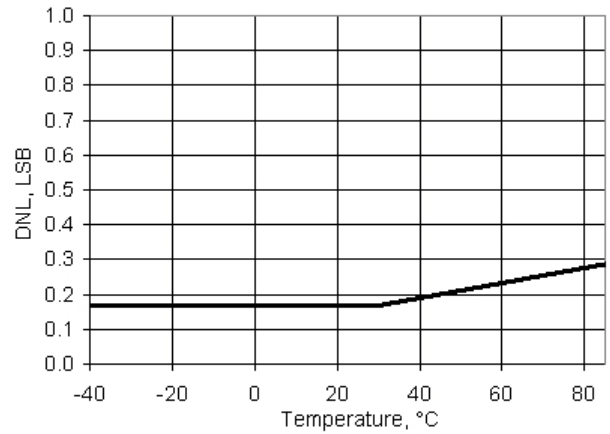
**Figure 11-30. VDAC DNL vs Input Code, 1 V Mode**



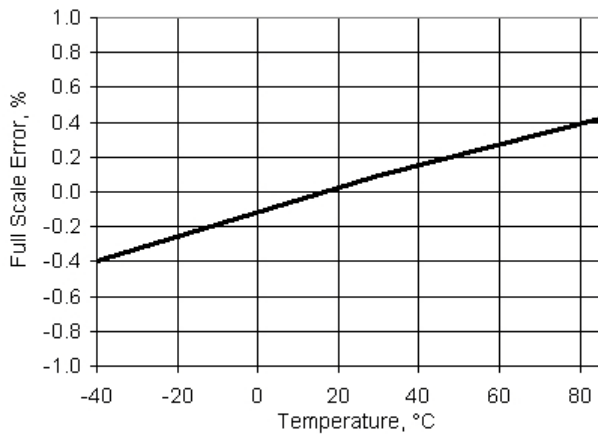
**Figure 11-31. VDAC INL vs Temperature, 1 V Mode**



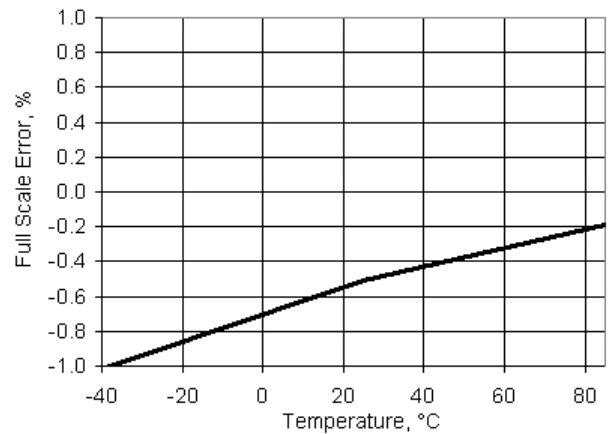
**Figure 11-32. VDAC DNL vs Temperature, 1 V Mode**



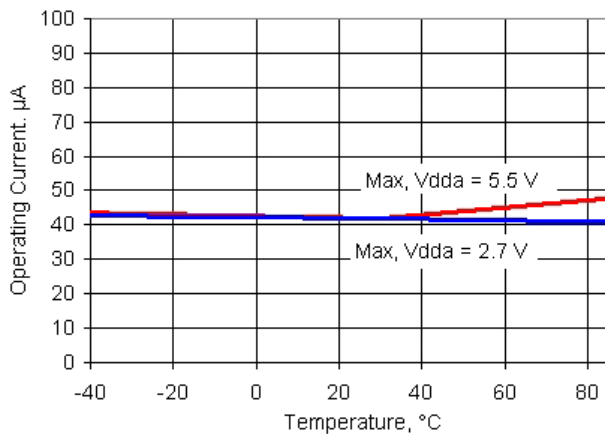
**Figure 11-33. VDAC Full Scale Error vs Temperature, 1 V Mode**



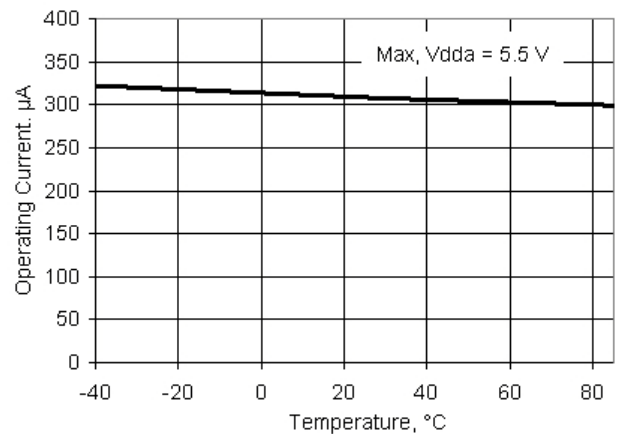
**Figure 11-34. VDAC Full Scale Error vs Temperature, 4 V Mode**



**Figure 11-35. VDAC Operating Current vs Temperature, 1 V Mode, Slow Mode**



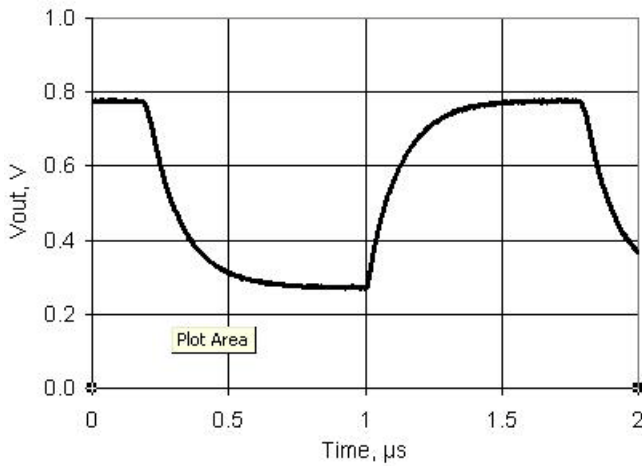
**Figure 11-36. VDAC Operating Current vs Temperature, 1 V Mode, Fast Mode**



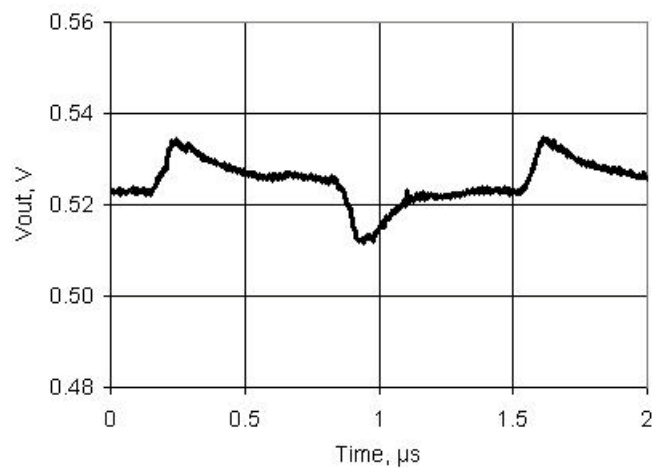
**Table 11-27. VDAC AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
F <sub>DAC</sub>	Update rate	1 V scale	–	–	1000	ksps
		4 V scale	–	–	250	ksps
T <sub>settleP</sub>	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
T <sub>settleN</sub>	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

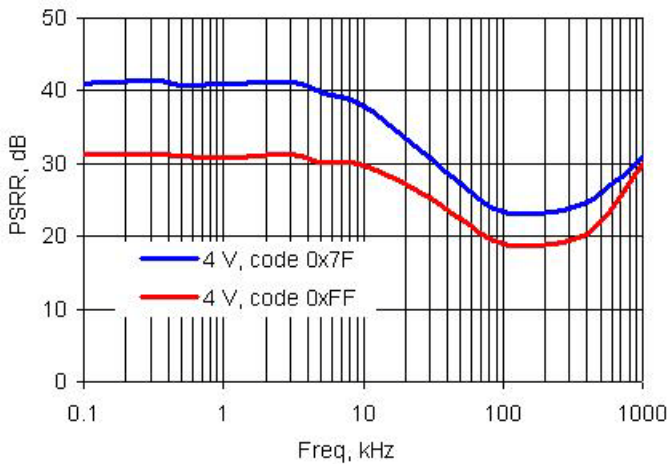
**Figure 11-37. VDAC Step Response, Codes 0x40 - 0xC0, 1 V Mode, Fast Mode, V<sub>dda</sub> = 5 V**



**Figure 11-38. VDAC Glitch Response, Codes 0x7F - 0x80, 1 V Mode, Fast Mode, V<sub>dda</sub> = 5 V**



**Figure 11-39. VDAC PSRR vs Frequency**



11.5.7 Temperature Sensor

**Table 11-28. Temperature Sensor Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Temp sensor accuracy	Range: -40 °C to +85 °C	-	±8	-	°C

11.5.8 LCD Direct Drive

**Table 11-29. LCD Direct Drive DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
I <sub>CC</sub>	LCD system operating current	Bus clock = 3 Mhz, V <sub>ddio</sub> = V <sub>dda</sub> = 3 V, 4 commons, 16 segments, 1/4 duty cycle, 50 Hz frame rate, no glass connected	-	63	-	μA
I <sub>CC_SEG</sub>	Current per segment driver	Strong drive mode	-	260	-	μA
V <sub>BIAS</sub>	LCD bias range (V <sub>BIAS</sub> refers to the main output voltage(V <sub>O</sub> ) of LCD DAC)	V <sub>D<sub>DDA</sub></sub> ≥ 3 V and V <sub>D<sub>DDA</sub></sub> ≥ V <sub>BIAS</sub>	2	-	5	V
	LCD bias step size	V <sub>D<sub>DDA</sub></sub> ≥ 3 V and V <sub>D<sub>DDA</sub></sub> ≥ V <sub>BIAS</sub>	-	9.1 × V <sub>D<sub>DDA</sub></sub>	-	mV
	LCD capacitance per segment/common driver	Drivers may be combined	-	500	5000	pF
	Long term segment offset		-	-	20	mV
I <sub>OUT</sub>	Output drive current per segment driver)	V <sub>ddio</sub> = 5.5V, strong drive mode	355	-	710	μA

**Table 11-30. LCD Direct Drive AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
f <sub>LCD</sub>	LCD frame rate		10	50	150	Hz

**11.6 Digital Peripherals**

Specifications are valid for  $-40\text{ }^{\circ}\text{C} \leq T_A \leq 85\text{ }^{\circ}\text{C}$  and  $T_J \leq 100\text{ }^{\circ}\text{C}$ , except where noted. Specifications are valid for 2.7 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 data sheet in PSoC Creator.

**Table 11-31. Timer DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Block current consumption	16-bit timer, at listed input clock frequency	–	–	–	$\mu\text{A}$
	3 MHz		–	15	–	$\mu\text{A}$
	12 MHz		–	60	–	$\mu\text{A}$
	40 MHz		–	260	–	$\mu\text{A}$

**Table 11-32. Timer AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Operating frequency		DC	–	40.01	MHz
	Capture pulse width (Internal)		25	–	–	ns
	Capture pulse width (external)		30	–	–	ns
	Timer resolution		25	–	–	ns
	Enable pulse width		25	–	–	ns
	Enable pulse width (external)		30	–	–	ns
	Reset pulse width		25	–	–	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 data sheet in PSoC Creator.

**Table 11-33. Counter DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Block current consumption	16-bit counter, at listed input clock frequency	–	–	–	$\mu\text{A}$
	3 MHz		–	15	–	$\mu\text{A}$
	12 MHz		–	60	–	$\mu\text{A}$
	40 MHz		–	260	–	$\mu\text{A}$

**Table 11-34. Counter AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Operating frequency		DC	–	40.01	MHz
	Capture pulse		25	–	–	ns
	Resolution		25	–	–	ns
	Pulse width		25	–	–	ns
	Pulse width (external)		30	–	–	ns
	Enable pulse width		25	–	–	ns
	Enable pulse width (external)		30	–	–	ns
	Reset pulse width		25	–	–	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 data sheet in PSoC Creator.

**Table 11-35. PWM DC Specifications**

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
	40 MHz		–	260	–	μA

**Table 11-36. PWM AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Operating frequency		DC	–	40.01	MHz
	Pulse width		25	–	–	ns
	Pulse width (external)		30	–	–	ns
	Kill pulse width		25	–	–	ns
	Kill pulse width (external)		30	–	–	ns
	Enable pulse width		25	–	–	ns
	Enable pulse width (external)		30	–	–	ns
	Reset pulse width		25	–	–	ns
	Reset pulse width (external)		30	–	–	ns

11.6.4 I<sup>2</sup>C

**Table 11-37. Fixed I<sup>2</sup>C DC Specifications**

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 11-38. Fixed I<sup>2</sup>C AC Specifications**

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

Controller Area Network<sup>[32]</sup>

**Table 11-39. CAN DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
I <sub>DD</sub>	Block current consumption		–	–	200	μA

**Table 11-40. CAN AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Bit rate	Minimum 8 MHz clock	–	–	1	Mbit

**Note**

32. Refer to ISO 11898 specification for details.

11.6.5 USB

**Table 11-41. USB DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
V <sub>USB_5</sub>	Device supply for USB operation	USB configured, USB regulator enabled	4.35	–	5.25	V
V <sub>USB_3.3</sub>		USB configured, USB regulator bypassed	3.15	–	3.6	V
V <sub>USB_3</sub>		USB configured, USB regulator bypassed <sup>[33]</sup>	2.85	–	3.6	V
I <sub>USB_Configured</sub>	Device supply current in device active mode, bus clock and IMO = 24 MHz	V <sub>DDD</sub> = 5 V, F <sub>CPU</sub> = 1.5 MHz	–	10	–	mA
		V <sub>DDD</sub> = 3.3 V, F <sub>CPU</sub> = 1.5 MHz	–	8	–	mA
I <sub>USB_Suspended</sub>	Device supply current in device sleep mode	V <sub>DDD</sub> = 5 V, connected to USB host, PICU configured to wake on USB resume signal	–	0.5	–	mA
		V <sub>DDD</sub> = 5 V, disconnected from USB host	–	0.3	–	mA
		V <sub>DDD</sub> = 3.3 V, connected to USB host, PICU configured to wake on USB resume signal	–	0.5	–	mA
		V <sub>DDD</sub> = 3.3 V, disconnected from USB host	–	0.3	–	mA

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 data sheets in PSoC Creator for full AC/DC specifications, APIs, and example code.

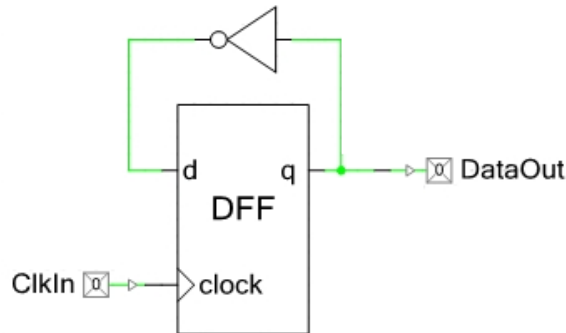
**Table 11-42. UDB AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
Datapath Performance						
F <sub>MAX_TIMER</sub>	Maximum frequency of 16-bit timer in a UDB pair		–	–	40.01	MHz
F <sub>MAX_ADDER</sub>	Maximum frequency of 16-bit adder in a UDB pair		–	–	40.01	MHz
F <sub>MAX_CRC</sub>	Maximum frequency of 16-bit CRC/PRS in a UDB pair		–	–	40.01	MHz
PLD Performance						
F <sub>MAX_PLD</sub>	Maximum frequency of a two-pass PLD function in a UDB pair		–	–	40.01	MHz
Clock to Output Performance						
t <sub>CLK_OUT</sub>	Propagation delay for clock in to data out, see Figure 11-40.	25 °C	–	20	25	ns
t <sub>CLK_OUT</sub>	Propagation delay for clock in to data out, see Figure 11-40.	Worst-case placement, routing, and pin selection	–	–	55	ns

**Note**

33. Rise/fall time matching (TR) not guaranteed, see [USB Driver AC Specifications](#) on page 65.

**Figure 11-40. Clock to Output Performance**



**11.7 Memory**

Specifications are valid for  $-40\text{ }^{\circ}\text{C} \leq T_A \leq 85\text{ }^{\circ}\text{C}$  and  $T_J \leq 100\text{ }^{\circ}\text{C}$ , except where noted. Specifications are valid for 2.7 V to 5.5 V, except where noted.

*11.7.1 Flash*

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

Parameter	Description	Conditions	Min	Typ	Max	Units
	Erase and program voltage	V <sub>DD</sub> pin	2.7	–	5.5	V

**Table 11-44. Flash AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
T <sub>WRITE</sub>	Row write time (erase + program)		–	15	20	ms
T <sub>ERASE</sub>	Row erase time		–	10	13	ms
	Row program time		–	5	7	ms
T <sub>BULK</sub>	Bulk erase time (256 KB)		–	–	140	ms
	Sector erase time (16 KB)		–	–	15	ms
	Total device program time (including SWD and other overhead)		–	–	20	seconds
	Flash data retention time, retention period measured from last erase cycle	Average ambient temp. T <sub>A</sub> ≤ 55 °C, 100 K erase/program cycles	20	–	–	years
		Average ambient temp. T <sub>A</sub> ≤ 85 °C, 10 K erase/program cycles	10	–	–	

*11.7.2 EEPROM*

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

Parameter	Description	Conditions	Min	Typ	Max	Units
	Erase and program voltage		2.7	–	5.5	V



**Table 11-46. EEPROM AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
T <sub>WRITE</sub>	Single row erase/write cycle time		–	2	20	ms
	EEPROM data retention time, retention period measured from last erase cycle	Average ambient temp, T <sub>A</sub> ≤ 25 °C, 1M erase/program cycles	20	–	–	years
		Average ambient temp, T <sub>A</sub> ≤ 55 °C, 100 K erase/program cycles	20	–	–	
		Average ambient temp, T <sub>A</sub> ≤ 85 °C, 10 K erase/program cycles	10	–	–	

11.7.3 SRAM

**Table 11-47. SRAM DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
V <sub>SRAM</sub>	SRAM retention voltage		1.2	–	–	V
	EEPROM data retention time, retention period measured from last erase cycle	Average ambient temp, T <sub>A</sub> ≤ 25 °C, 1M erase/program cycles	20	–	–	years
		Average ambient temp, T <sub>A</sub> ≤ 55 °C, 100 K erase/program cycles	20	–	–	
		Average ambient temp, T <sub>A</sub> ≤ 85 °C, 10 K erase/program cycles	10	–	–	

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

Parameter	Description	Conditions	Min	Typ	Max	Units
F <sub>SRAM</sub>	SRAM operating frequency		DC	–	40.01	MHz

**11.8 PSoC System Resources**

Specifications are valid for  $-40\text{ }^{\circ}\text{C} \leq T_A \leq 85\text{ }^{\circ}\text{C}$  and  $T_J \leq 100\text{ }^{\circ}\text{C}$ , except where noted. Specifications are valid for 2.7 V to 5.5 V, except where noted.

*11.8.1 Voltage Monitors*

**Table 11-49. Voltage Monitors DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
LVI	Trip voltage					
	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 11-50. Voltage Monitors AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Response time		–	–	1	μs

*11.8.2 Interrupt Controller*

**Table 11-51. Interrupt Controller AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Delay from interrupt signal input to ISR code execution from main line code <sup>[34]</sup>		–	–	12	Tcy CPU
	Delay from interrupt signal input to ISR code execution from ISR code (tail-chaining) <sup>[34]</sup>		–	–	6	Tcy CPU

**Note**

34. ARM Cortex-M3 NVIC spec. Visit [www.arm.com](http://www.arm.com) for detailed documentation about the Cortex-M3 CPU.

11.8.3 SWD Interface

**Table 11-55. SWD Interface AC Specifications<sup>[35]</sup>**

Parameter	Description	Conditions	Min	Typ	Max	Units
f_SWDCCK	SWDCLK frequency	$3.3\text{ V} \leq V_{\text{DDD}} \leq 5\text{ V}$	–	–	12 <sup>[37]</sup>	MHz
		$2.7\text{ V} \leq V_{\text{DDD}} < 3.3\text{ V}$	–	–	7 <sup>[37]</sup>	MHz
		$2.7\text{ V} \leq V_{\text{DDD}} < 3.3\text{ V}$ , SWD over USBIO pins	–	–	5.5 <sup>[37]</sup>	MHz
T_SWDI_setup	SWDIO input setup before SWDCK high	$T = 1/f_{\text{SWDCCK}}$ max	T/4	–	–	
T_SWDI_hold	SWDIO input hold after SWDCK high	$T = 1/f_{\text{SWDCCK}}$ max	T/4	–	–	
T_SWDO_valid	SWDCK high to SWDIO output	$T = 1/f_{\text{SWDCCK}}$ max	–	–	2T/5	
T_SWDO_hold	SWDIO output hold after SWDCK low	$T = 1/f_{\text{SWDCCK}}$ max	T/4	–	–	

11.8.4 TPIU Interface

**Table 11-56. TPIU Interface AC Specifications<sup>[35]</sup>**

Parameter	Description	Conditions	Min	Typ	Max	Units
	SWV bit rate		–	–	33 <sup>[38]</sup>	Mbit

**Notes**

- 35. Based on device characterization (Not production tested).
- 36. f\_TCK must also be no more than 1/3 CPU clock frequency.
- 37. f\_SWDCCK must also be no more than 1/3 CPU clock frequency.
- 38. SWV signal frequency and bit rate are limited by GPIO output frequency, see “GPIO AC Specifications” on page 59.

**11.9 Clocking**

Specifications are valid for  $-40\text{ }^{\circ}\text{C} \leq T_A \leq 85\text{ }^{\circ}\text{C}$  and  $T_J \leq 100\text{ }^{\circ}\text{C}$ , except where noted. Specifications are valid for 2.7 V to 5.5 V, except where noted. Unless otherwise specified, all charts and graphs show typical values.

*11.9.1 32 kHz External Crystal*

**Table 11-57. 32 kHz External Crystal DC Specifications<sup>[39]</sup>**

Parameter	Description	Conditions	Min	Typ	Max	Units
I <sub>CC</sub>	Operating current	Low power mode	–	0.25	1.0	μA
CL	External crystal capacitance		–	6	–	pF
DL	Drive level		–	–	1	μW

**Table 11-58. 32 kHz External Crystal AC Specifications**

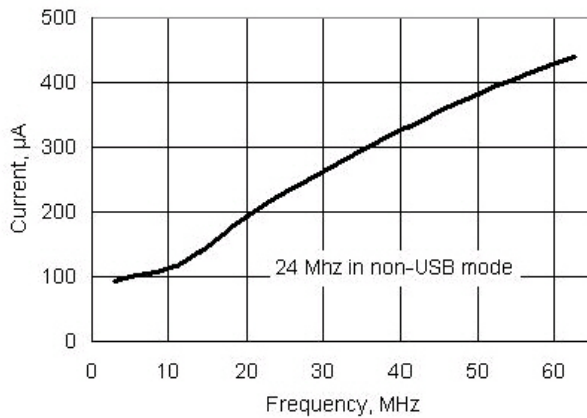
Parameter	Description	Conditions	Min	Typ	Max	Units
F	Frequency		–	32.768	–	kHz
T <sub>ON</sub>	Startup time	High power mode	–	1	–	s

*11.9.2 Internal Main Oscillator*

**Table 11-59. IMO DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Supply current					
	24 MHz		–	–	300	μA
	12 MHz		–	–	200	μA
	6 MHz		–	–	180	μA
	3 MHz		–	–	150	μA

**Figure 11-41. IMO Current vs. Frequency**



**Table 11-60. IMO AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
F <sub>IMO</sub>	IMO frequency stability (with factory trim)					
	24 MHz		–6	–	6	%
	12 MHz		–6	–	6	%
	6 MHz		–4	–	4	%
	3 MHz		–4	–	4	%

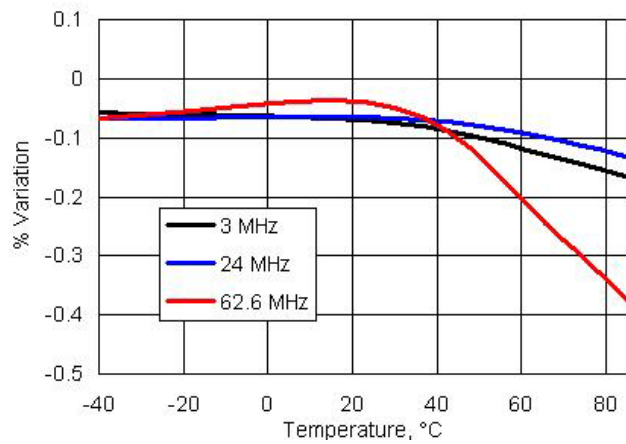
**Note**

<sup>39</sup>. Based on device characterization (Not production tested).

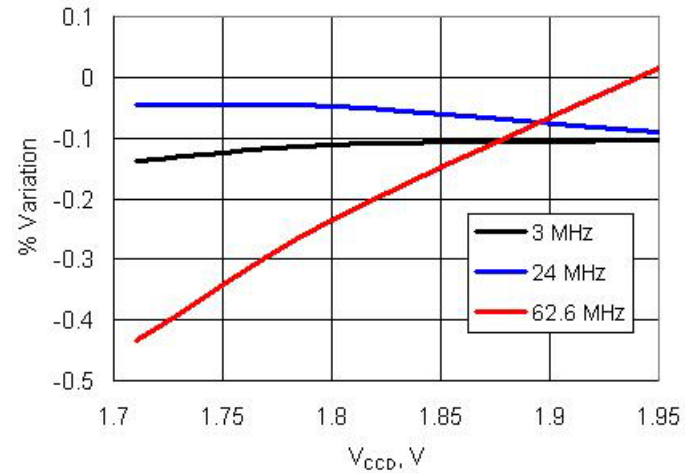
**Table 11-60. IMO AC Specifications (continued)**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Startup time <sup>[39]</sup>	From enable (during normal system operation) or wakeup from low power state	–	–	12	μs
Jp-p	Jitter (peak to peak) <sup>[39]</sup>					
	F = 24 MHz		–	0.9	–	ns
	F = 3 MHz		–	1.6	–	ns
Jperiod	Jitter (long term) <sup>[39]</sup>					
	F = 24 MHz		–	0.9	–	ns
	F = 3 MHz		–	12	–	ns

**Figure 11-42. IMO Frequency Variation vs. Temperature**



**Figure 11-43. IMO Frequency Variation vs. V<sub>CCD</sub>**



11.9.3 Internal Low Speed Oscillator

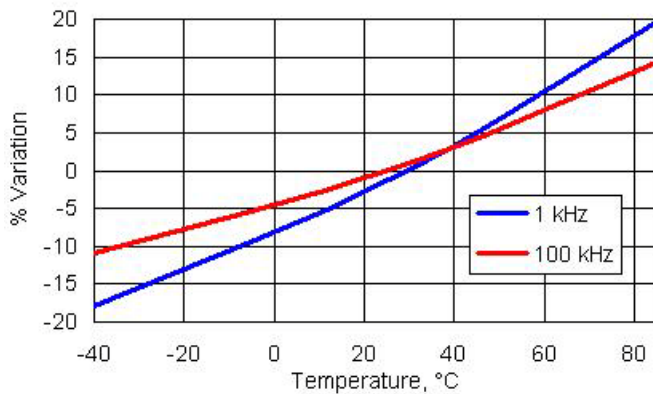
**Table 11-61. ILO DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
I <sub>CC</sub>	Operating current	F <sub>OUT</sub> = 1 kHz	–	0.3	1.7	μA
		F <sub>OUT</sub> = 33 kHz	–	1.0	2.6	μA
		F <sub>OUT</sub> = 100 kHz	–	1.0	2.6	μA
	Leakage current	Power down mode	–	2.0	15	nA

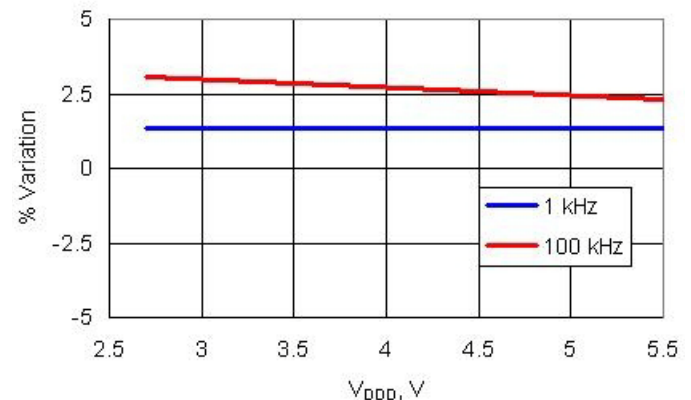
**Table 11-62. ILO AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
	Startup time, all frequencies	Turbo mode	–	–	2.5	ms
F <sub>ILO</sub>	ILO frequencies (trimmed)					
		100 kHz	45	100	200	kHz
		1 kHz	0.5	1	2	kHz
	ILO frequencies (untrimmed)					
		100 kHz	30	100	300	kHz
		1 kHz	0.3	1	3.5	kHz

**Figure 11-44. ILO Frequency Variation vs. Temperature**



**Figure 11-45. ILO Frequency Variation vs. V<sub>DD</sub>**



11.9.4 External Crystal Oscillator

**Table 11-63. ECO AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
F	Crystal frequency range		4	–	25	MHz

11.9.5 External Clock Reference

**Table 11-64. External Clock Reference AC Specifications<sup>[40]</sup>**

Parameter	Description	Conditions	Min	Typ	Max	Units
	External frequency range		0	–	33	MHz
	Input duty cycle range	Measured at V <sub>DDIO</sub> /2	30	50	70	%
	Input edge rate	V <sub>IL</sub> to V <sub>IH</sub>	0.1	–	–	V/ns

**Note**

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

11.9.6 Phase-Locked Loop

**Table 11-65. PLL DC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
I <sub>DD</sub>	PLL operating current	In = 3 MHz, Out = 24 MHz	–	200	–	μA

**Table 11-66. PLL AC Specifications**

Parameter	Description	Conditions	Min	Typ	Max	Units
F <sub>pllin</sub>	PLL input frequency <sup>[41]</sup>		1	–	40	MHz
	PLL intermediate frequency <sup>[42]</sup>	Output of prescaler	1	–	3	MHz
F <sub>plout</sub>	PLL output frequency <sup>[41]</sup>		24	–	40	MHz
	Lock time at startup		–	–	250	μs
J <sub>period-rms</sub>	Jitter (rms) <sup>[43]</sup>		–	–	250	ps

**Notes**

- 41. This specification is guaranteed by testing the PLL across the specified range using the IMO as the source for the PLL.
- 42. 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.
- 43. Based on device characterization (Not production tested).

## 12. Ordering Information

In addition to the features listed in [Table 12-1](#), every CY8C52 device includes: up to 256 KB flash, 64 KB SRAM, 2 KB EEPROM, a precision on-chip voltage reference, precision oscillators, flash, DMA, a fixed function I<sup>2</sup>C, SWD programming and debug, 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 CY8C52 derivatives incorporate device and flash security in user-selectable security levels; see the TRM for details.

**Table 12-1. CY8C52 Family with ARM Cortex-M3 CPU**

Part Number	MCU Core				Analog							Digital				I/O <sup>[45]</sup>				Package	Device ID <sup>[46]</sup>	
	CPU Speed (MHz)	Flash (KB)	SRAM (KB)	EEPROM (KB)	LCD Segment Drive	ADC	DAC	Comparators	SC/CT Analog Blocks	Opamps	DFB	CapSense	UDBs <sup>[44]</sup>	16-bit Timer/PWM	FS USB	CAN 2.0b	Total I/O	GPIO	SIO			USBIO
CY8C5248LTI-030	40	256	64	2	✓	1x12-bit SAR	1	2	-	-	-	✓	24	4	✓	-	46	36	8	2	68-pin QFN	0x0E116069
CY8C5248AXI-047	40	256	64	2	✓	1x12-bit SAR	1	2	-	-	-	✓	24	4	✓	-	70	60	8	2	100-pin TQFP	0x0E134069
CY8C5247LTI-089	40	128	32	2	✓	1x12-bit SAR	1	2	-	-	-	✓	24	4	✓	-	46	36	8	2	68-pin QFN	0x0E141069
CY8C5247AXI-051	40	128	32	2	✓	1x12-bit SAR	1	2	-	-	-	✓	24	4	✓	-	70	60	8	2	100-pin TQFP	0x0E114069
CY8C5246LTI-029	40	64	16	2	✓	1x12-bit SAR	1	2	-	-	-	✓	24	4	✓	-	46	36	8	2	68-pin QFN	0x0E143069
CY8C5246AXI-054	40	64	16	2	✓	1x12-bit SAR	1	2	-	-	-	✓	24	4	✓	-	70	60	8	2	100-pin TQFP	0x0E13C069

### 12.1 Part Numbering Conventions

PSoC 5 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-xxx

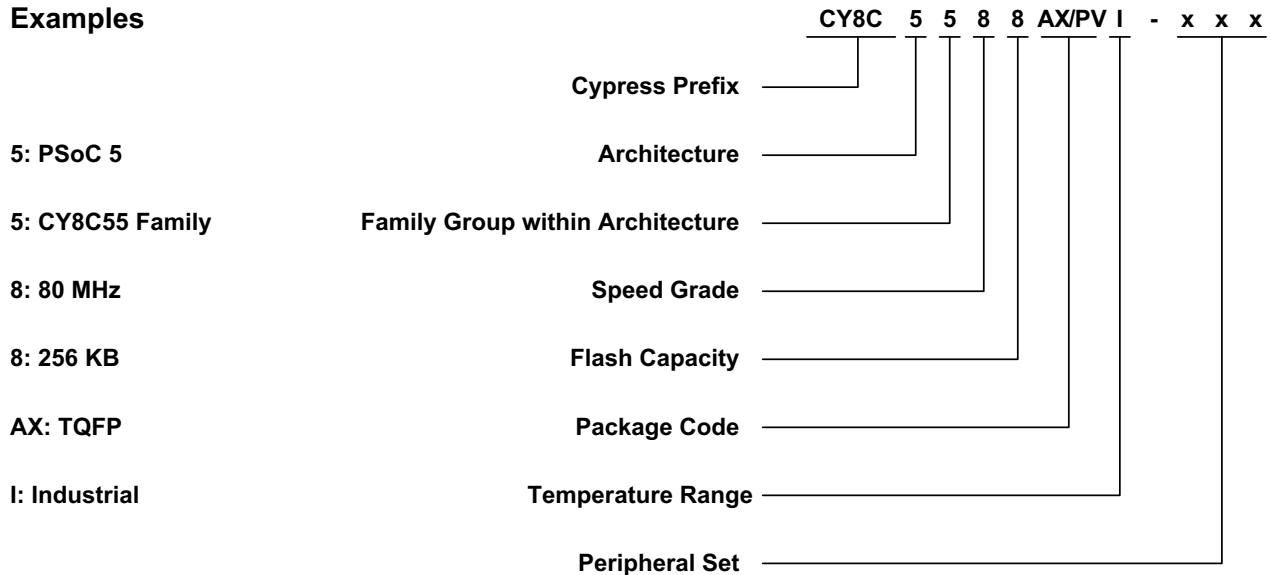
- a: Architecture
  - 3: PSoC 3
  - 5: PSoC 5
- b: Family group within architecture
  - 2: CY8C52 family
  - 3: CY8C53 family
  - 4: CY8C54 family
  - 5: CY8C55 family
- c: Speed grade
  - 4: 40 MHz
  - 8: 67 MHz
- d: Flash capacity
  - 5: 32 KB
  - 6: 64 KB
  - 7: 128 KB
  - 8: 256 KB
- ef: Package code
  - Two character alphanumeric
  - AX: TQFP
  - LT: QFN
- g: Temperature range
  - C: commercial
  - I: industrial
  - A: automotive
- xxx: Peripheral set
  - Three character numeric
  - No meaning is associated with these three characters

#### Notes

44. 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 31 for more information on how UDBs can be used.
45. 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 25 for details on the functionality of each of these types of I/O.
46. The device 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.



**Examples**



All devices in the PSoC 5 CY8C52 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 data sheets (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 <sub>A</sub>	Operating ambient temperature		-40	25	85	°C
T <sub>J</sub>	Operating junction temperature		-40	-	100	°C
T <sub>ja</sub>	Package θJA (68-pin QFN)		-	15	-	°C/Watt
T <sub>ja</sub>	Package θJA (100-pin TQFP)		-	34	-	°C/Watt
T <sub>jc</sub>	Package θJC (68-pin QFN)		-	13	-	°C/Watt
T <sub>jc</sub>	Package θJC (100-pin TQFP)		-	10	-	°C/Watt

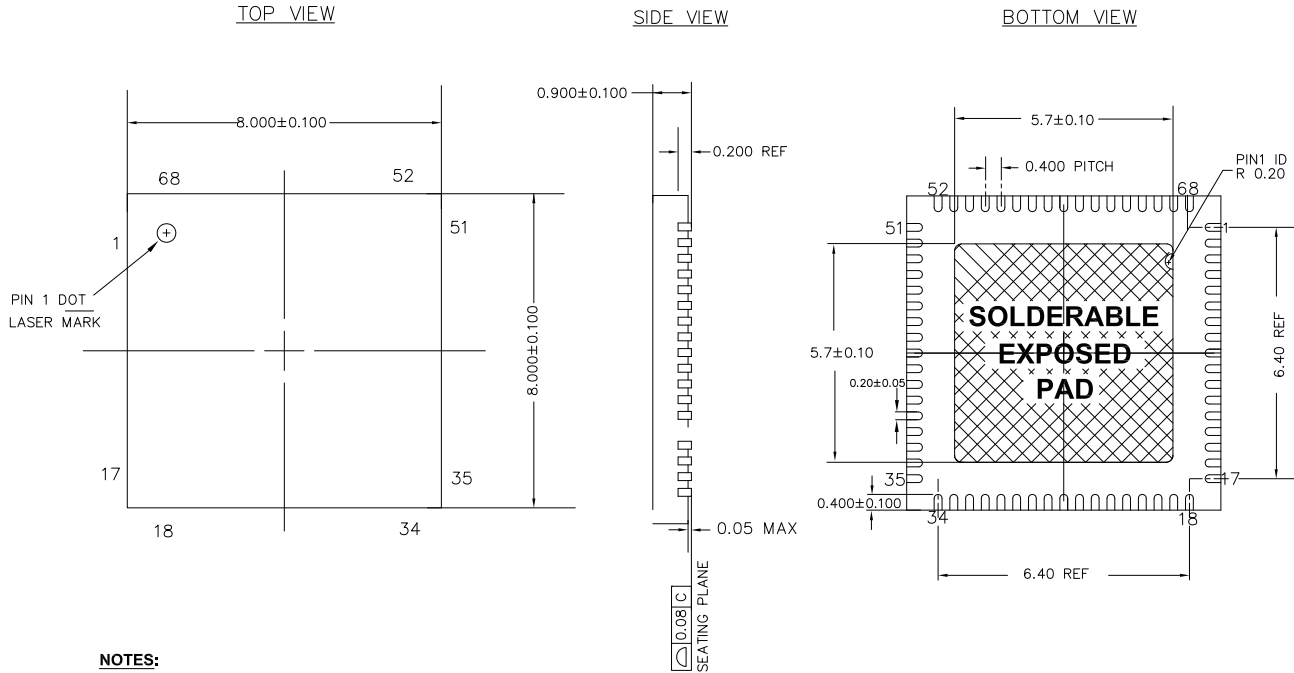
**Table 13-2. Solder Reflow Peak Temperature**

Package	Maximum Peak Temperature	Maximum Time at Peak Temperature
68-pin QFN	260 °C	30 seconds
100-pin TQFP	260 °C	30 seconds


**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

**Figure 13-1. 68-pin QFN 8x8 with 0.4 mm Pitch Package Outline (Sawn Version)**

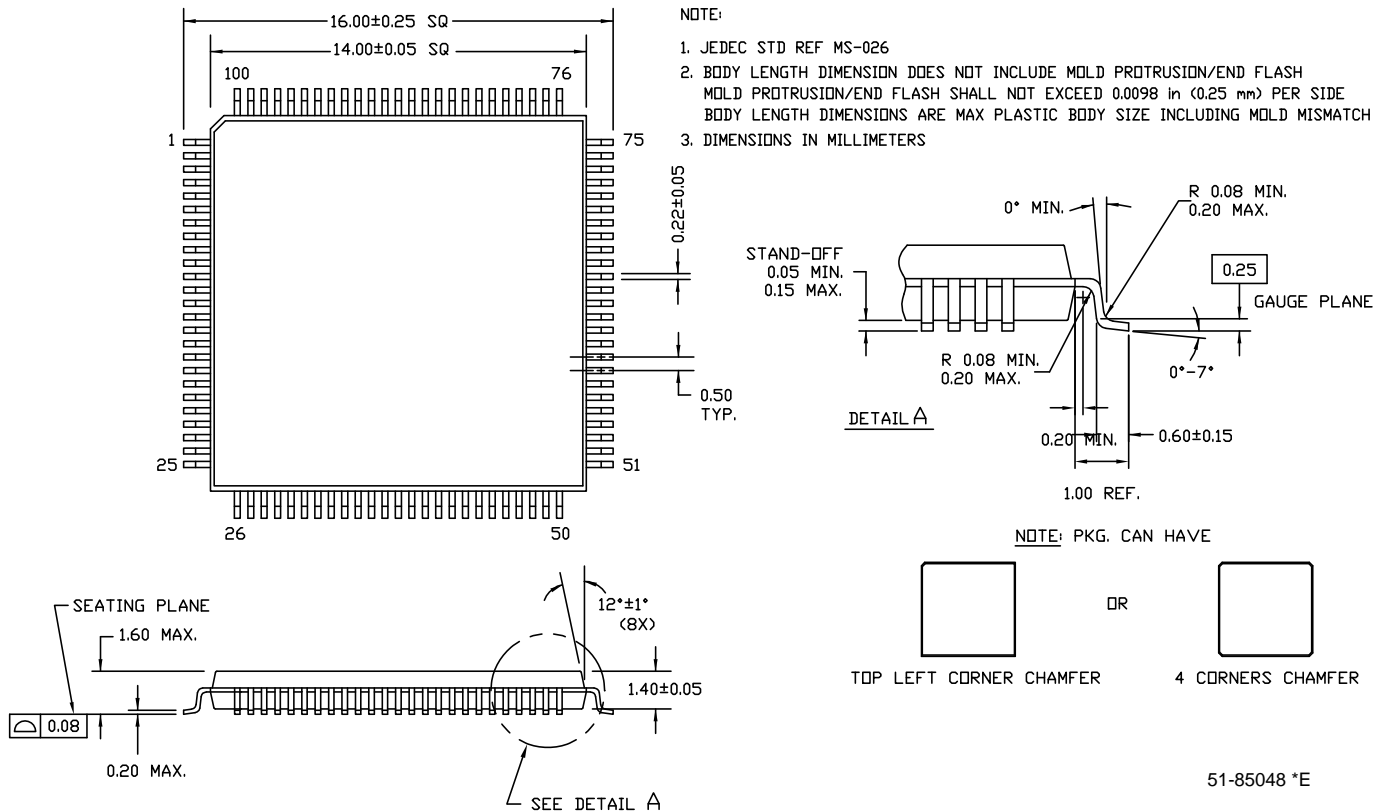


**NOTES:**

1.  HATCH AREA IS SOLDERABLE EXPOSED METAL.
2. REFERENCE JEDEC#: MO-220
3. PACKAGE WEIGHT: 0.17g
4. ALL DIMENSIONS ARE IN MILLIMETERS

001-09618 \*C

**Figure 13-2. 100-pin TQFP (14 x 14 x 1.4 mm) Package Outline**



51-85048 \*E

**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 <sup>®</sup>	advanced RISC machine, a CPU architecture
ATM	automatic thump mode
BW	bandwidth
CAN	Controller Area Network, a communications protocol
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
ECO	external crystal oscillator
EEPROM	electrically erasable programmable read-only memory
EMI	electromagnetic interference
EOC	end of conversion
EOF	end of frame
EPSR	execution program status register
ESD	electrostatic discharge
FIR	finite impulse response, see also IIR
FPB	flash patch and breakpoint
FS	full-speed

**Table 14-1. Acronyms Used in this Document (continued)**

Acronym	Description
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
I <sup>2</sup> C, 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
PICU	port interrupt control unit

**Table 14-1. Acronyms Used in this Document** (continued)

Acronym	Description
PLA	programmable logic array
PLD	programmable logic device, see also PAL
PLL	phase-locked loop
PMDD	package material declaration data sheet
POR	power-on 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 <sup>2</sup> C serial clock
SDA	I <sup>2</sup> 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

**Table 14-1. Acronyms Used in this Document** (continued)

Acronym	Description
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
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 I/O pin
XTAL	crystal

**15. Reference Documents**

- [PSoC® 3, PSoC® 5 Architecture TRM](#)
- [PSoC® 5 Registers TRM](#)

## 16. Document Conventions

### 16.1 Units of Measure

**Table 16-1. 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
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

**17. Revision History**

Description Title: PSoC <sup>®</sup> 5: CY8C52 Family Datasheet Programmable System-on-Chip (PSoC <sup>®</sup> ) Document Number: 001-66236				
Rev.	ECN No.	Submission Date	Orig. of Change	Description of Change
**	3198501	03/17/2011	MKEA	New data sheet.
*A	3279676	06/10/2011	MKEA	Changed MHzECO range Updated Flash and EEPROM AC specs Added solder reflow peak temperature table Changed IDAC IDD numbers and VDAC Added flash retention specs Added JTAG and SWD interface diagrams Removed mention of comparator wakeup from sleep Updated PSoC Power system diagram Updated clocking sections Removed references to JTAG interface Updated I/O graphs Added note that Interrupt Specs are ARM Specs Changed JTAG and SWD max speeds Modified ILO startup time Removed references to ETM and TRACEPORT Updated IDAC range limits Updated Vddio pin description Updated Power Modes section Added note on watchdog timer in the Reset section Updaed ESD <sub>HBM</sub> value Updated Boost Converter section



## 18. Sales, Solutions, and Legal Information

### Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturers' representatives, and distributors. To find the office closest to you, visit us at [Cypress Locations](#).

#### Products

Automotive	<a href="http://cypress.com/go/automotive">cypress.com/go/automotive</a>
Clocks & Buffers	<a href="http://cypress.com/go/clocks">cypress.com/go/clocks</a>
Interface	<a href="http://cypress.com/go/interface">cypress.com/go/interface</a>
Lighting & Power Control	<a href="http://cypress.com/go/powerpsoc">cypress.com/go/powerpsoc</a> <a href="http://cypress.com/go/plc">cypress.com/go/plc</a>
Memory	<a href="http://cypress.com/go/memory">cypress.com/go/memory</a>
Optical & Image Sensing	<a href="http://cypress.com/go/image">cypress.com/go/image</a>
PSoC	<a href="http://cypress.com/go/psoc">cypress.com/go/psoc</a>
Touch Sensing	<a href="http://cypress.com/go/touch">cypress.com/go/touch</a>
USB Controllers	<a href="http://cypress.com/go/USB">cypress.com/go/USB</a>
Wireless/RF	<a href="http://cypress.com/go/wireless">cypress.com/go/wireless</a>

#### PSoC Solutions

[psoc.cypress.com/solutions](http://psoc.cypress.com/solutions)  
PSoC 1 | PSoC 3 | PSoC 5

---

© Cypress Semiconductor Corporation, 2011. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.