# PSoC® 3: CY8C34 Family Data Sheet

## Programmable System-on-Chip (PSoC®)

## General Description

With its unique array of configurable blocks, PSoC® 3 is a true system level solution providing microcontroller unit (MCU), memory, analog, and digital peripheral functions in a single chip. The CY8C34 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 CY8C34 family can handle dozens of data acquisition channels and analog inputs on every general-purpose input/output (GPIO) pin. The CY8C34 family isalso a high-performance configurable digital system with some part numbers including interfaces such as USB, multimaster inter-integrated circuit ($I^2C$), and controller area network (CAN). In addition to communication interfaces, the CY8C34 family has an easy to configure logic array, flexible routing to all I/O pins, and a high-performance single cycle 8051 microprocessor core. You 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 CY8C34 family provides unparalleled opportunities for analog and digital bill of materials integration while easily accommodating last minute design changes through simple firmware updates.

## Features

- Single cycle 8051 CPU core
  - DC to 50 MHz operation
  - Multiply and divide instructions
  - Flash program memory, up to 64 KB, 100,000 write cycles, 20 years retention, and multiple security features
  - Up to 8-KB flash error correcting code (ECC) or configuration storage
  - Up to 8 KB SRAM
  - Up to 2 KB electrically erasable programmable read-only memory (EEPROM), 1 M cycles, and 20 years retention
  - 24-channel direct memory access (DMA) with multilayer AHB[1] bus access
    - Programmable chained descriptors and priorities
    - High bandwidth 32-bit transfer support
- Low voltage, ultra low-power
  - Wide operating voltage range: 0.5 V to 5.5 V
  - High efficiency boost regulator from 0.5-V input through 1.8-V to 5.0-V output
  - 0.8 mA at 3 MHz, 1.2 mA at 6 MHz, and 6.6 mA at 50 MHz
  - Low-power modes including:
    - 1-µA sleep mode with real-time clock (RTC) and low-voltage detect (LVD) interrupt
    - 200-nA hibernate mode with RAM retention
- Versatile I/O system
  - 28 to 72 I/O (62 GPIOs, eight special input/outputs (SIO), two USBIOs[2])
  - Any GPIO to any digital or analog peripheral routability
  - LCD direct drive from any GPIO, up to 46×16 segments[2]
  - CapSense® support from any GPIO[3]
  - 1.2-V to 5.5-V I/O interface voltages, up to four domains
  - Maskable, independent interrupt request (IRQ) on any pin or port
  - Schmitt-trigger transistor-transistor logic (TTL) inputs
  - All GPIOs configurable as open drain high/low, pull-up/pull-down, High Z, or strong output
  - Configurable GPIO pin state at power-on reset (POR)
  - 25 mA sink on SIO
- Digital peripherals
  - 16 to 24 programmable logic device (PLD) based universal digital blocks (UDB)
  - Full CAN 2.0b 16-receive (Rx), 8-transmit (Tx) buffers[2]
  - Full-speed (FS) USB 2.0 12 Mbps using internal oscillator[2]
  - Up to four 16-bit configurable timer, counter, and PWM blocks
  - Library of standard peripherals
    - 8-, 16-, 24-, and 32-bit timers, counters, and PWMs
    - Serial peripheral interface (SPI), universal asynchronous transmitter receiver (UART), $I^2C$
    - 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 ($1.71\ V \le V_{DDA} \le 5.5\ V$)
  - 1.024 V±0.9-percent internal voltage reference across –40 °C to +85 °C (14 ppm/°C)
  - Configurable delta-sigma ADC with 8- to12-bit resolution
    - Programmable gain stage: ×0.25 to ×16
    - 12-bit mode, 192-ksps, 66-dB signal to noise and distortion ratio (SINAD), ±1-bit INL/DNL
  - Two 8-bit, 8-Msps IDACs or 1-Msps VDACs
  - Four comparators with 95-ns response time
  - Two uncommitted opamps with 25-mA drive capability
  - Two configurable multifunction analog blocks. Example configurations are programmable gain amplifier (PGA), transimpedance amplifier (TIA), mixer, and sample and hold
  - CapSense support
- Programming, debug, and trace
  - JTAG (4-wire), serial wire debug (SWD) (2-wire), and single wire viewer (SWV) interfaces
  - Eight address and one data breakpoint
  - 4-KB instruction trace buffer
  - Bootloader programming supportable through $I^2C$, 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 50 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
  - 48-pin SSOP, 48-pin QFN, 68-pin QFN, and 100-pin TQFP package options

**Notes**
1. AHB – AMBA (advanced microcontroller bus architecture) high-performance bus, an ARM data transfer bus
2. This feature on select devices only. See Ordering Information on page 115 for details.
3. GPIOs with opamp outputs are not recommended for use with CapSense.

## Contents

**1. Architectural Overview ..... 3**
**2. Pinouts ..... 5**
**3. Pin Descriptions ..... 10**
**4. CPU ..... 11**
4.1 8051 CPU ..... 11
4.2 Addressing Modes ..... 11
4.3 Instruction Set ..... 12
4.4 DMA and PHUB ..... 16
4.5 Interrupt Controller ..... 17
**5. Memory ..... 21**
5.1 Static RAM ..... 21
5.2 Flash Program Memory ..... 21
5.3 Flash Security ..... 21
5.4 EEPROM ..... 21
5.5 Nonvolatile Latches (NVLs) ..... 22
5.6 External Memory Interface ..... 23
5.7 Memory Map ..... 23
**6. System Integration ..... 25**
6.1 Clocking System ..... 25
6.2 Power System ..... 28
6.3 Reset ..... 31
6.4 I/O System and Routing ..... 32
**7. Digital Subsystem ..... 38**
7.1 Example Peripherals ..... 39
7.2 Universal Digital Block ..... 42
7.3 UDB Array Description ..... 45
7.4 DSI Routing Interface Description ..... 46
7.5 CAN ..... 47
7.6 USB ..... 49
7.7 Timers, Counters, and PWMs ..... 49
7.8 $I^2C$ ..... 49
**8. Analog Subsystem ..... 50**
8.1 Analog Routing ..... 52
8.2 Delta-sigma ADC ..... 54
8.3 Comparators ..... 55
8.4 Opamps ..... 56
8.5 Programmable SC/CT Blocks ..... 56
8.6 LCD Direct Drive ..... 58
8.7 CapSense ..... 59
8.8 Temp Sensor ..... 59
8.9 DAC ..... 59
8.10 Up/Down Mixer ..... 59
8.11 Sample and Hold ..... 60
**9. Programming, Debug Interfaces, Resources ..... 60**
9.1 JTAG Interface ..... 61
9.2 Serial Wire Debug Interface ..... 62
9.3 Debug Features ..... 63
9.4 Trace Features ..... 63
9.5 Single Wire Viewer Interface ..... 63
9.6 Programming Features ..... 63
9.7 Device Security ..... 63
**10. Development Support ..... 64**
10.1 Documentation ..... 64
10.2 Online ..... 64
10.3 Tools ..... 64
**11. Electrical Specifications ..... 65**
11.1 Absolute Maximum Ratings ..... 65
11.2 Device Level Specifications ..... 66
11.3 Power Regulators ..... 70
11.4 Inputs and Outputs ..... 74
11.5 Analog Peripherals ..... 82
11.6 Digital Peripherals ..... 99
11.7 Memory ..... 102
11.8 PSoC System Resources ..... 108
11.9 Clocking ..... 111
**12. Ordering Information ..... 115**
12.1 Part Numbering Conventions ..... 116
**13. Packaging ..... 117**
**14. Acronyms ..... 120**
**15. Reference Documents ..... 121**
**16. Document Conventions ..... 122**
16.1 Units of Measure ..... 122
**17. Revision History ..... 123**
**18. Sales, Solutions, and Legal Information .....127**

# 1. Architectural Overview

Introducing the CY8C34 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 CY8C34 family provides configurable blocks of analog, digital, and interconnect circuitry around a CPU subsystem. The combination of a CPU with a flexible analog subsystem, digital subsystem, routing, and I/O enables a high level of integration in a wide variety of consumer, industrial, and medical applications.

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

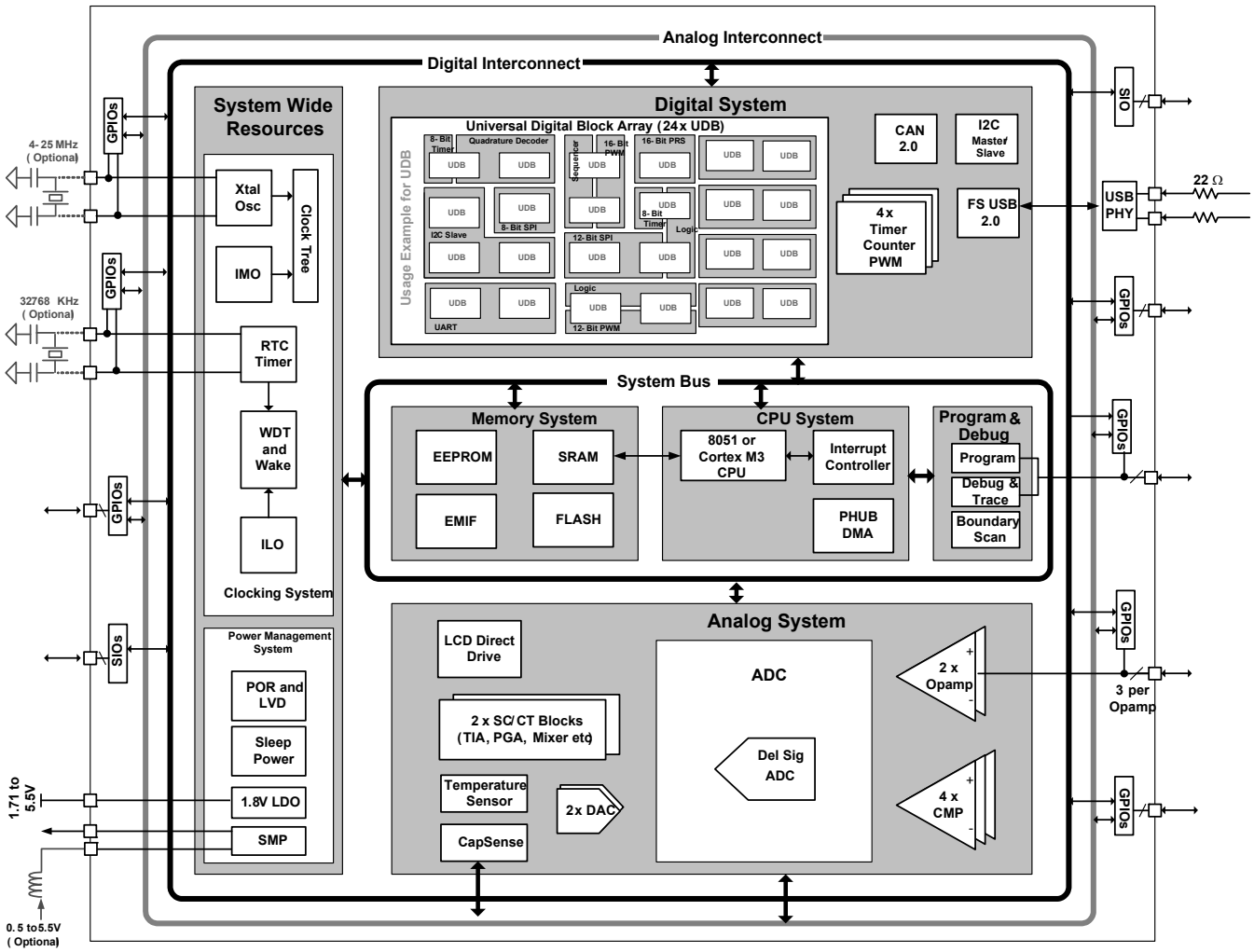



Figure 1-1 illustrates the major components of the CY8C34 family. They are:

- 8051 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 prebuilt and tested standard digital peripherals (UART, SPI, LIN, PRS, CRC, timer, counter, PWM, AND, OR, and so on) that are mapped to the UDB array. You can also easily create a digital circuit using boolean primitives by means of graphical design entry. Each UDB contains programmable array logic (PAL)/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

CY8C34 family these blocks can include four 16-bit timer, counter, and PWM blocks; $I^2C$ slave, master, and multi-master; Full-Speed USB; and Full CAN 2.0b.

For more details on the peripherals see the "Example Peripherals" section on page 39 of this data sheet. For information on UDBs, DSI, and other digital blocks, see the "Digital Subsystem" section on page 38 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 0.9-percent error over temperature and voltage. The configurable analog subsystem includes:

- Analog muxes
- Comparators
- Voltage references
- Analog-to-digital converter (ADC)
- Digital-to-analog converters (DACs)

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 heart of the analog subsystem is a fast, accurate, configurable delta-sigma ADC with these features:

- Less than 100 µV offset
- A gain error of 0.2 percent
- INL less than ±1 LSB
- DNL less than ±1 LSB
- SINAD better than 66 dB

This converter addresses a wide variety of precision analog applications, including some of the most demanding sensors.

Two high-speed voltage or current DACs support 8-bit output signals at update rate of 8 Msps in current DAC (IDAC) and 1 Msps in voltage DAC (VDAC). They 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 DACs, the analog subsystem provides multiple:

- Uncommitted opamps
- Configurable switched capacitor/continuous time (SC/CT) blocks. These support:
  - Transimpedance amplifiers
  - Programmable gain amplifiers
  - Mixers
  - Other similar analog components

See the "Analog Subsystem" section on page 50 of this data sheet for more details.

PSoC's 8051 CPU subsystem is built around a single cycle pipelined 8051 8-bit processor running at up to 50 MHz. The CPU subsystem includes a programmable nested vector interrupt controller, DMA controller, and RAM. PSoC's nested vector interrupt controller provides low latency by allowing the CPU to vector directly to the first address of the interrupt service routine, bypassing the jump instruction required by other architectures. 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 single cycle 8051 CPU runs ten times faster than a standard 8051 processor. The processor speed itself is configurable, allowing you to tune active power consumption for specific applications.

PSoC's nonvolatile subsystem consists of flash, byte-writeable EEPROM, and nonvolatile configuration options. It provides up to 64 KB of on-chip flash. The CPU can reprogram individual blocks of flash, enabling bootloaders. You can enable an error correcting code (ECC) for high reliability applications. A powerful and flexible protection model secures the user's sensitive information, allowing selective memory block locking for read and write protection. Up to 2 KB of byte-writeable EEPROM is available on-chip to store application data. Additionally, selected configuration options such as boot speed and pin drive mode are stored in nonvolatile memory. This allows settings to activate immediately after power-on reset (POR).

The three types of PSoC I/O are extremely flexible. All I/Os have many drive modes that are set at POR. PSoC also provides up to four I/O voltage domains through the $V_{DDIO}$ pins. Every GPIO has analog I/O, LCD drive[4], CapSense[5], flexible interrupt generation, slew rate control, and digital I/O capability. The SIOs on PSoC allow $V_{OH}$ to be set independently of $V_{DDIO}$ 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 $I^2C$ 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 of the features of the PSoC I/Os are covered in detail in the "I/O System and Routing" section on page 32 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-percent 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.

**Notes**
4. This feature on select devices only. See Ordering Information on page 115 for details.
5. GPIOs with opamp outputs are not recommended for use with CapSense.

The device provides a PLL to generate system clock frequencies up to 50 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 CY8C34 family supports a wide supply operating range from 1.71 V to 5.5 V. This allows operation from regulated supplies such as 1.8 V ± 5 percent, 2.5 V ±10 percent, 3.3 V ± 10 percent, or 5.0 V ± 10 percent, or directly from a wide range of battery types. In addition, it provides an integrated high efficiency synchronous boost converter that can power the device from supply voltages as low as 0.5 V. This enables the device to be powered directly from a single battery or solar cell. In addition, you can use the boost converter to generate other voltages required by the device, such as a 3.3-V supply for LCD glass drive. The boost's output is available on the $V_{BOOST}$ pin, allowing other devices in the application to be powered from the PSoC.

PSoC supports a wide range of low-power modes. These include a 200-nA hibernate mode with RAM retention and a 1-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 1.2 mA when the CPU is running at 6 MHz, or 0.8 mA running at 3 MHz.

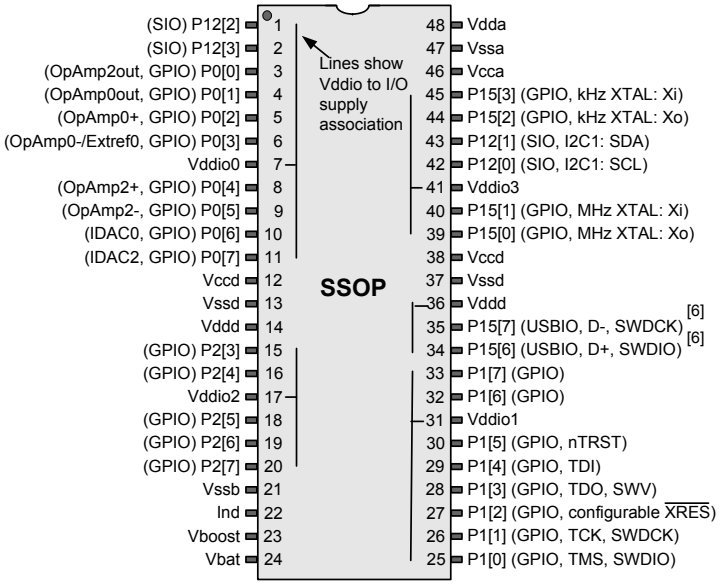
The details of the PSoC power modes are covered in the "Power System" section on page 28 of this data sheet.

PSoC uses JTAG (4-wire) or SWD (2-wire) interfaces for programming, debug, and test. The 1-wire SWV may also be used for 'printf' style debugging. By combining SWD and SWV, you can implement a full debugging interface with just three pins. Using these standard interfaces enables you to debug or program the PSoC with a variety of hardware solutions from Cypress or third party vendors. PSoC supports on-chip break points and 4 KB instruction and data race memory for debug. Details of the programming, test, and debugging interfaces are discussed in the "Programming, Debug Interfaces, Resources" section on page 60 of this data sheet.

## 2. Pinouts

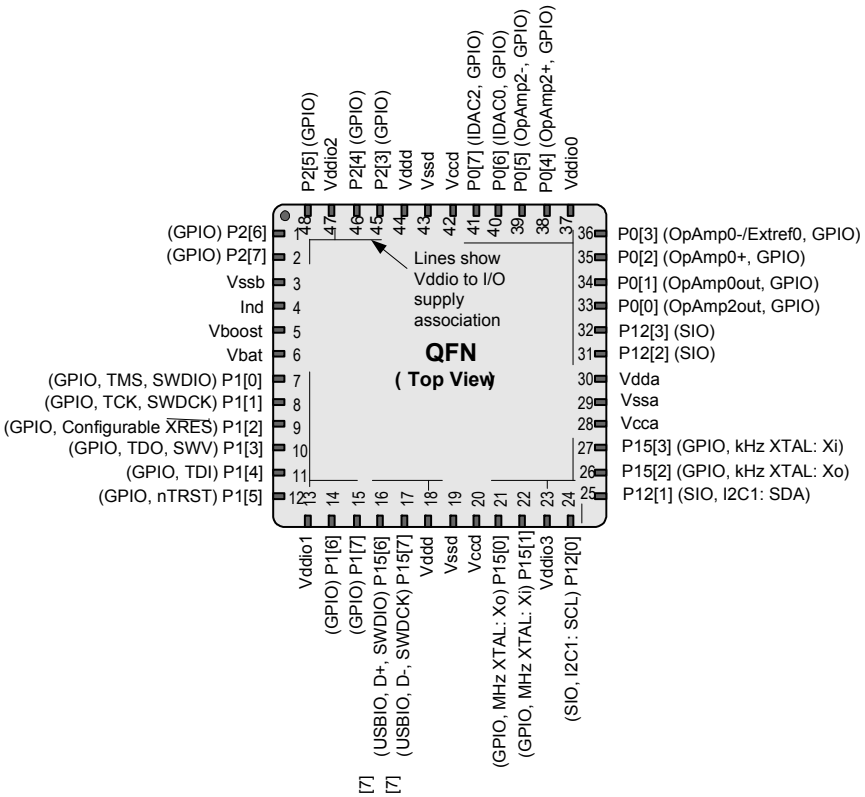
The Vddio pin that supplies a particular set of pins is indicated by the black lines drawn on the pinout diagrams in Figure 2-1 through Figure 2-4. Using the Vddio pins, a single PSoC can support multiple interface voltage levels, eliminating the need for off-chip level shifters. Each Vddio may sink up to 100 mA total to its associated I/O pins and opamps. 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. 48-pin SSOP Part Pinout**



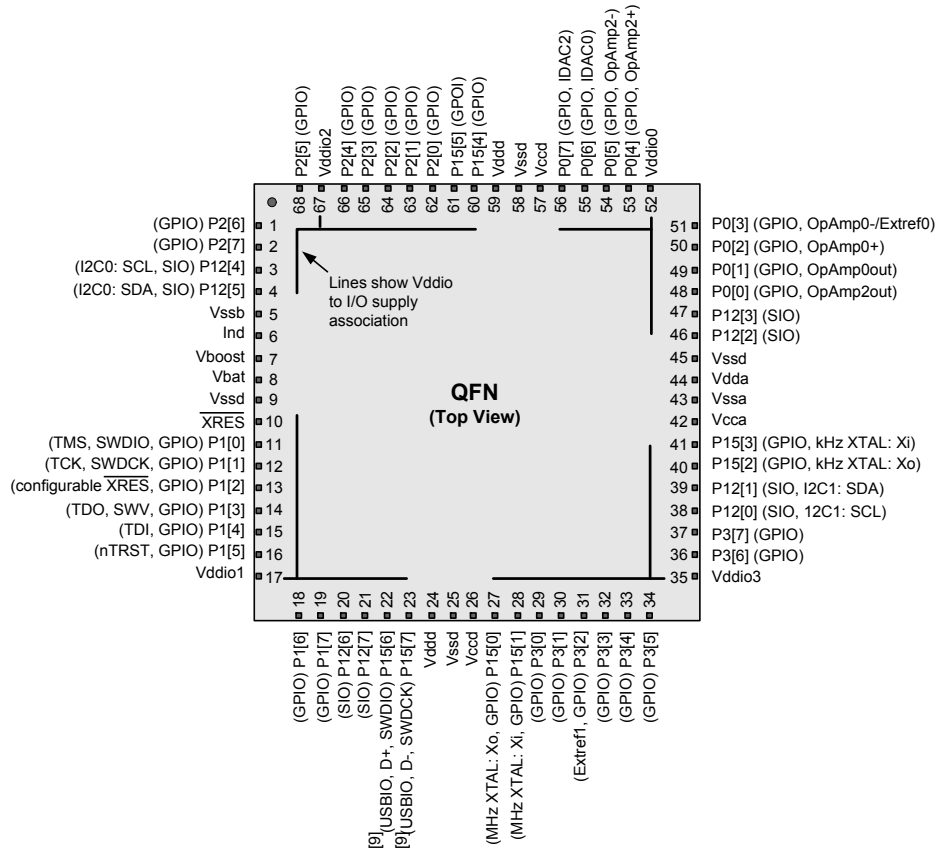

**Note**
6. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.

**Figure 2-2. 48-pin QFN Part Pinout**[8]



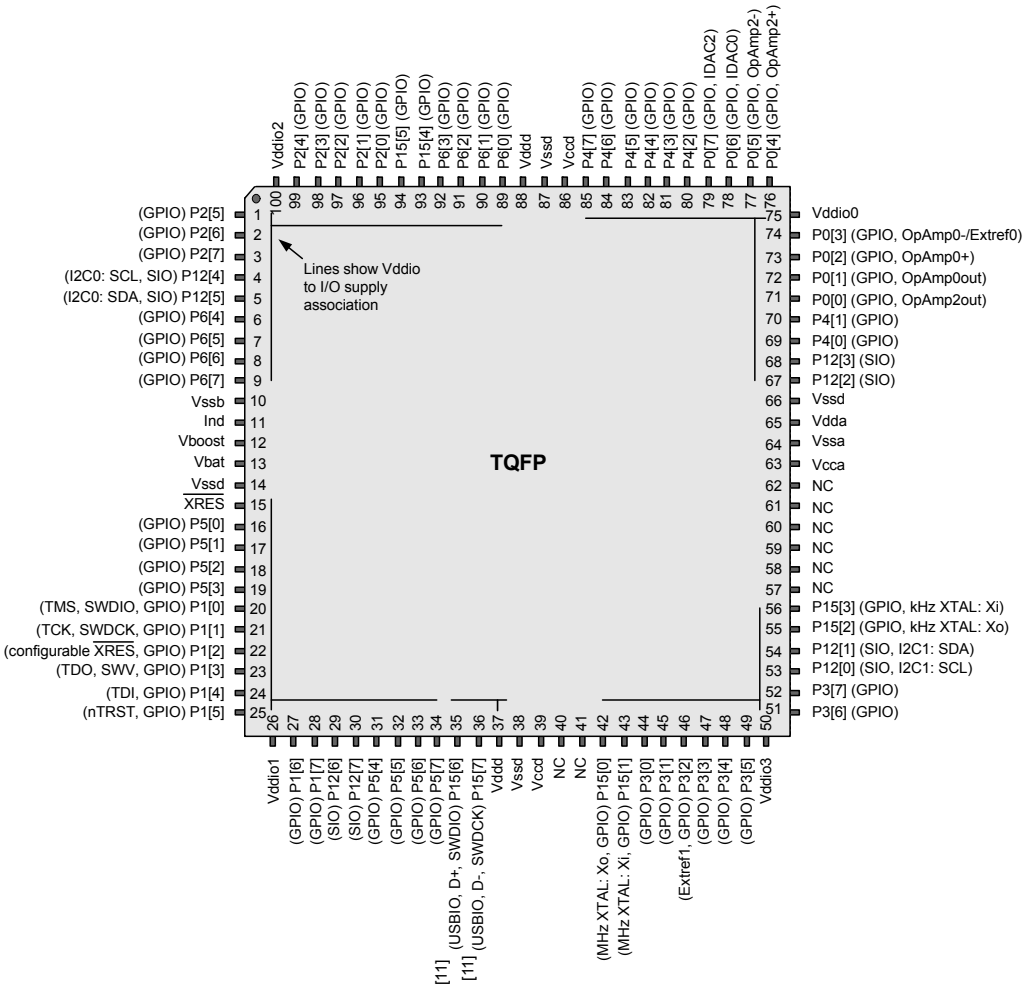

**Notes**
7. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.
8. The center pad on the QFN package should be connected to digital ground ($V_{SSD}$) for best mechanical, thermal, and electrical performance. If not connected to ground, it should be electrically floated and not connected to any other signal.

**Figure 2-3. 68-Pin QFN Part Pinout**[10]




**Notes**
9. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.
10. 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.

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



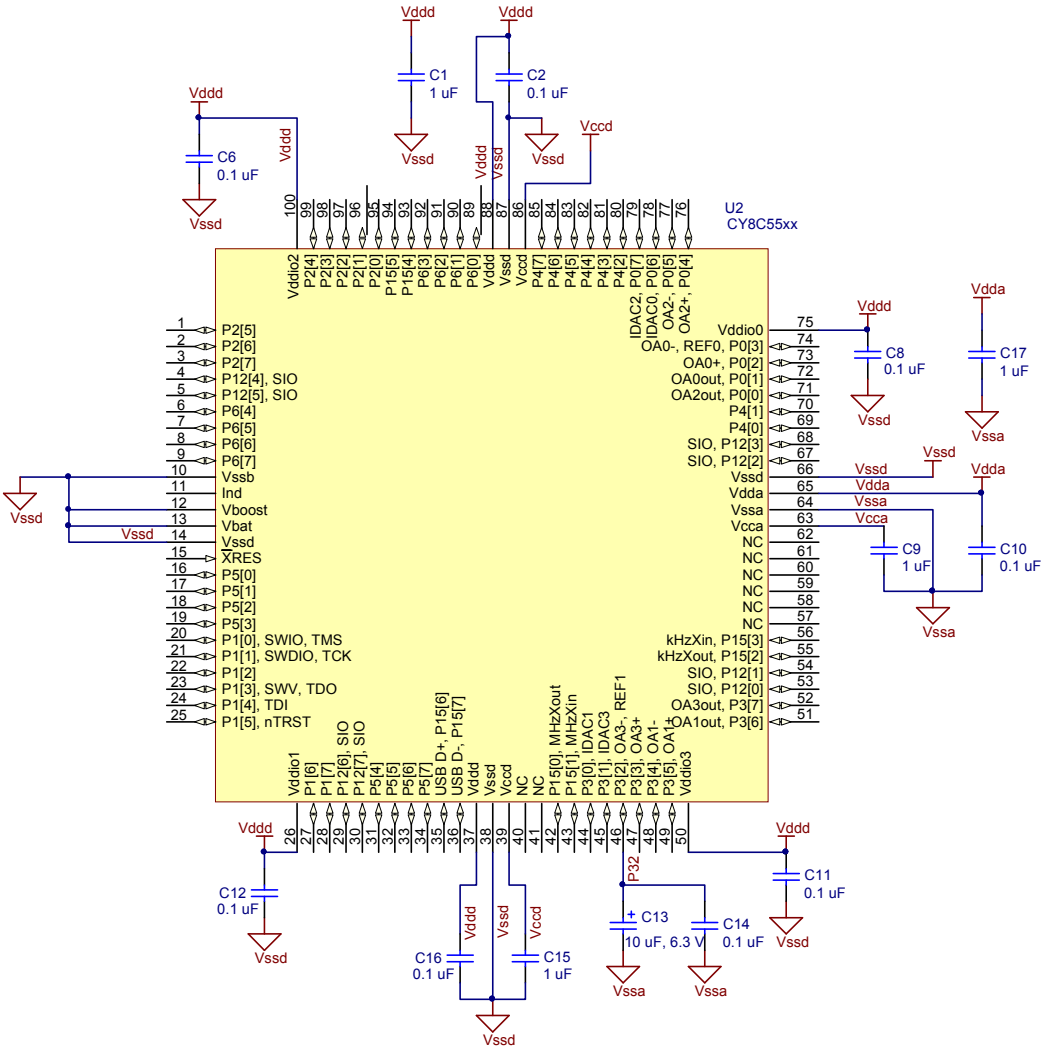

**Note**
11. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.

Figure 2-5 and Figure 2-6 on page 10 show an example schematic and an example PCB layout, for the 100-pin TQFP part, for optimal analog performance on a two-layer board.

- The two pins labeled Vddd must be connected together.
- The two pins labeled Vccd must be connected together, with capacitance added, as shown in Figure 2-5 and Power System on page 28. The trace between the two Vccd pins should be as short as possible.
- The two pins labeled Vssd must be connected together.

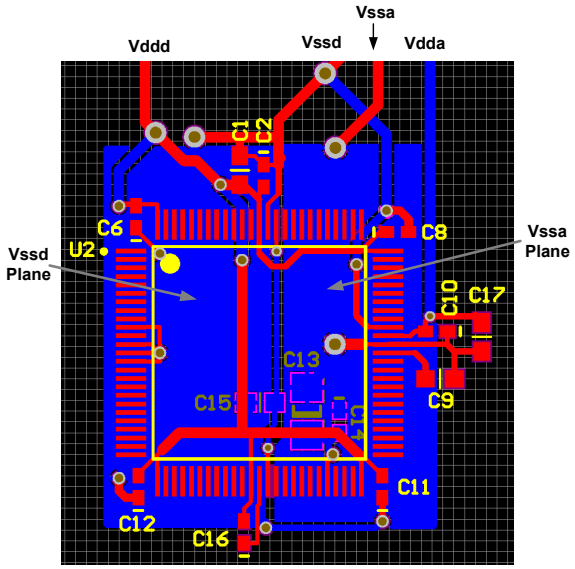
For information on circuit board layout issues for mixed signals, refer to the application note AN57821 - Mixed Signal Circuit Board Layout Considerations for PSoC® 3 and PSoC 5.

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




**Note** The two Vccd pins must be connected together with as short a trace as possible. A trace under the device is recommended, as shown in Figure 2-6 on page 10.

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




## 3. Pin Descriptions

**IDAC0, IDAC2**

Low resistance output pin for high current DACs (IDAC).

**OpAmp0out, OpAmp2out**

High current output of uncommitted opamp[12].

**Extref0, Extref1**

External reference input to the analog system.

**OpAmp0–, OpAmp2–**

Inverting input to uncommitted opamp.

**OpAmp0+, OpAmp2+**

Noninverting input to uncommitted opamp.

**GPIO**

General purpose I/O pin provides interfaces to the CPU, digital peripherals, analog peripherals, interrupts, LCD segment drive, and CapSense[12].

**I2C0: SCL, I2C1: SCL**

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

**I2C0: SDA, I2C1: SDA**

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

**Ind**

Inductor connection to boost pump.

**kHz XTAL: Xo, kHz XTAL: Xi**

32.768-kHz crystal oscillator pin.

**MHz XTAL: Xo, MHz XTAL: Xi**

4- to 25-MHz crystal oscillator pin.

**nTRST**

Optional JTAG test reset programming and debug port connection to reset the JTAG connection.

**SIO**

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

**SWDCK**

Serial wire debug clock programming and debug port connection.

**SWDIO**

Serial wire debug input and output programming and debug port connection.

**SWV.**

Single wire viewer debug output.

**TCK**

JTAG test clock programming and debug port connection.

**TDI**

JTAG test data In programming and debug port connection.

**TDO**

JTAG test data out programming and debug port connection.

**TMS**

JTAG test mode select programming and debug port connection.

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

**USBIO, D+**

Provides D+ connection directly to a USB 2.0 bus. May be used as a digital I/O pin. 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. Pins are No Connect (NC) on devices without USB.

**Vboost**

Power sense connection to boost pump.

**Vbat**

Battery supply to boost pump.

**Vcca**

Output of analog core regulator and input to analog core. Requires a 1-µF capacitor to $V_{SSA}$. Regulator output not for external use.

**Vccd**

Output of digital core regulator and input to digital core. The two $V_{CCD}$ pins must be shorted together, with the trace between them as short as possible, and a 1-µF capacitor to $V_{SSD}$; see Power System on page 28. Regulator output not for external use.

**Vdda**

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

**Vddd**

Supply for all digital peripherals and digital core regulator. Vddd must be less than or equal to Vdda.

**Vssa**

Ground for all analog peripherals.

**Vssb**

Ground connection for boost pump.

**Vssd**

Ground for all digital logic and I/O pins.

**Vddio0, Vddio1, Vddio2, Vddio3**

Supply for I/O pins. See pinouts for specific I/O pin to Vddio mapping. Each Vddio must be tied to a valid operating voltage (1.71 V to 5.5 V), and must be less than or equal to Vdda. If the I/O pins associated with Vddio0, Vddio2 or Vddio3 are not used then that Vddio should be tied to ground (Vssd or Vssa).

**$\overline{\text{XRES}}$ (and configurable $\overline{\text{XRES}}$)**

External reset pin. Active low with internal pull-up. Pin P1[2] may be configured to be a XRES pin; see "Nonvolatile Latches (NVLs)" on page 22.

# 4. CPU

## 4.1 8051 CPU

The CY8C34 devices use a single cycle 8051 CPU, which is fully compatible with the original MCS-51 instruction set. The CY8C34 family uses a pipelined RISC architecture, which executes most instructions in 1 to 2 cycles to provide peak performance of up to 24 MIPS with an average of 2 cycles per instruction. The single cycle 8051 CPU runs ten times faster than a standard 8051 processor.

The 8051 CPU subsystem includes these features:

- Single cycle 8051 CPU
- Up to 64 KB of flash memory, up to 2 KB of EEPROM, and up to 8 KB of SRAM
- Programmable nested vector interrupt controller
- Direct memory access (DMA) controller
- Peripheral HUB (PHUB)
- External memory interface (EMIF)

## 4.2 Addressing Modes

The following addressing modes are supported by the 8051:

- Direct Addressing: The operand is specified by a direct 8-bit address field. Only the internal RAM and the SFRs can be accessed using this mode.
- Indirect Addressing: The instruction specifies the register which contains the address of the operand. The registers R0 or R1 are used to specify the 8-bit address, while the data pointer (DPTR) register is used to specify the 16-bit address.
- Register Addressing: Certain instructions access one of the registers (R0 to R7) in the specified register bank. These instructions are more efficient because there is no need for an address field.
- Register Specific Instructions: Some instructions are specific to certain registers. For example, some instructions always act on the accumulator. In this case, there is no need to specify the operand.
- Immediate Constants: Some instructions carry the value of the constants directly instead of an address.
- Indexed Addressing: This type of addressing can be used only for a read of the program memory. This mode uses the Data Pointer as the base and the accumulator value as an offset to read a program memory.
- Bit Addressing: In this mode, the operand is one of 256 bits.

## 4.3 Instruction Set

The 8051 instruction set is highly optimized for 8-bit handling and Boolean operations. The types of instructions supported include:

- Arithmetic instructions
- Logical instructions
- Data transfer instructions
- Boolean instructions
- Program branching instructions

### *4.3.1 Instruction Set Summary*

#### *4.3.1.1 Arithmetic Instructions*

Arithmetic instructions support the direct, indirect, register, immediate constant, and register-specific instructions. Arithmetic modes are used for addition, subtraction, multiplication, division, increment, and decrement operations. Table 4-1 on page 12 Table 4-1 lists the different arithmetic instructions.

**Table 4-1. Arithmetic Instructions**

| Mnemonic | Description | Bytes | Cycles |
|---|---|---|---|
| ADD A,Rn | Add register to accumulator | 1 | 1 |
| ADD A,Direct | Add direct byte to accumulator | 2 | 2 |
| ADD A,@Ri | Add indirect RAM to accumulator | 1 | 2 |
| ADD A,#data | Add immediate data to accumulator | 2 | 2 |
| ADDC A,Rn | Add register to accumulator with carry | 1 | 1 |
| ADDC A,Direct | Add direct byte to accumulator with carry | 2 | 2 |
| ADDC A,@Ri | Add indirect RAM to accumulator with carry | 1 | 2 |
| ADDC A,#data | Add immediate data to accumulator with carry | 2 | 2 |
| SUBB A,Rn | Subtract register from accumulator with borrow | 1 | 1 |
| SUBB A,Direct | Subtract direct byte from accumulator with borrow | 2 | 2 |
| SUBB A,@Ri | Subtract indirect RAM from accumulator with borrow | 1 | 2 |
| SUBB A,#data | Subtract immediate data from accumulator with borrow | 2 | 2 |
| INC A | Increment accumulator | 1 | 1 |
| INC Rn | Increment register | 1 | 2 |
| INC Direct | Increment direct byte | 2 | 3 |
| INC @Ri | Increment indirect RAM | 1 | 3 |
| DEC A | Decrement accumulator | 1 | 1 |
| DEC Rn | Decrement register | 1 | 2 |
| DEC Direct | Decrement direct byte | 2 | 3 |
| DEC @Ri | Decrement indirect RAM | 1 | 3 |
| INC DPTR | Increment data pointer | 1 | 1 |
| MUL | Multiply accumulator and B | 1 | 2 |
| DIV | Divide accumulator by B | 1 | 6 |
| DAA | Decimal adjust accumulator | 1 | 3 |

#### 4.3.1.2 Logical Instructions

The logical instructions perform Boolean operations such as AND, OR, XOR on bytes, rotate of accumulator contents, and swap of nibbles in an accumulator. The Boolean operations on the bytes are performed on the bit-by-bit basis. Table 4-2 on page 13Table 4-2 shows the list of logical instructions and their description.

**Table 4-2. Logical Instructions**

| Mnemonic | Description | Bytes | Cycles |
|---|---|---|---|
| ANL A,Rn | AND register to accumulator | 1 | 1 |
| ANL A,Direct | AND direct byte to accumulator | 2 | 2 |
| ANL A,@Ri | AND indirect RAM to accumulator | 1 | 2 |
| ANL A,#data | AND immediate data to accumulator | 2 | 2 |
| ANL Direct, A | AND accumulator to direct byte | 2 | 3 |
| ANL Direct, #data | AND immediate data to direct byte | 3 | 3 |
| ORL A,Rn | OR register to accumulator | 1 | 1 |
| ORL A,Direct | OR direct byte to accumulator | 2 | 2 |
| ORL A,@Ri | OR indirect RAM to accumulator | 1 | 2 |
| ORL A,#data | OR immediate data to accumulator | 2 | 2 |
| ORL Direct, A | OR accumulator to direct byte | 2 | 3 |
| ORL Direct, #data | OR immediate data to direct byte | 3 | 3 |
| XRL A,Rn | XOR register to accumulator | 1 | 1 |
| XRL A,Direct | XOR direct byte to accumulator | 2 | 2 |
| XRL A,@Ri | XOR indirect RAM to accumulator | 1 | 2 |
| XRL A,#data | XOR immediate data to accumulator | 2 | 2 |
| XRL Direct, A | XOR accumulator to direct byte | 2 | 3 |
| XRL Direct, #data | XOR immediate data to direct byte | 3 | 3 |
| CLR A | Clear accumulator | 1 | 1 |
| CPL A | Complement accumulator | 1 | 1 |
| RL A | Rotate accumulator left | 1 | 1 |
| RLC A | Rotate accumulator left through carry | 1 | 1 |
| RR A | Rotate accumulator right | 1 | 1 |
| RRC A | Rotate accumulator right though carry | 1 | 1 |
| SWAP A | Swap nibbles within accumulator | 1 | 1 |

#### 4.3.1.3 Data Transfer Instructions

The data transfer instructions are of three types: the core RAM, xdata RAM, and the lookup tables. The core RAM transfer includes transfer between any two core RAM locations or SFRs. These instructions can use direct, indirect, register, and immediate addressing. The xdata RAM transfer includes only the transfer between the accumulator and the xdata RAM location. It can use only indirect addressing. The lookup tables involve nothing but the read of program memory using the Indexed addressing mode. Table 4-3 lists the various data transfer instructions available.

#### 4.3.1.4 Boolean Instructions

The 8051 core has a separate bit addressable memory location. It has 128 bits of bit-addressable RAM and a set of SFRs that are bit addressable. The instruction set includes the whole menu of bit operations such as move, set, clear, toggle, OR, and AND instructions and the conditional jump instructions. Table 4-4 on page 15Table 4-4 lists the available Boolean instructions.

**Table 4-3. Data Transfer Instructions**

| Mnemonic | Description | Bytes | Cycles |
|---|---|---|---|
| MOV A,Rn | Move register to accumulator | 1 | 1 |
| MOV A,Direct | Move direct byte to accumulator | 2 | 2 |
| MOV A,@Ri | Move indirect RAM to accumulator | 1 | 2 |
| MOV A,#data | Move immediate data to accumulator | 2 | 2 |
| MOV Rn,A | Move accumulator to register | 1 | 1 |
| MOV Rn,Direct | Move direct byte to register | 2 | 3 |
| MOV Rn, #data | Move immediate data to register | 2 | 2 |
| MOV Direct, A | Move accumulator to direct byte | 2 | 2 |
| MOV Direct, Rn | Move register to direct byte | 2 | 2 |
| MOV Direct, Direct | Move direct byte to direct byte | 3 | 3 |
| MOV Direct, @Ri | Move indirect RAM to direct byte | 2 | 3 |
| MOV Direct, #data | Move immediate data to direct byte | 3 | 3 |
| MOV @Ri, A | Move accumulator to indirect RAM | 1 | 2 |
| MOV @Ri, Direct | Move direct byte to indirect RAM | 2 | 3 |
| MOV @Ri, #data | Move immediate data to indirect RAM | 2 | 2 |
| MOV DPTR, #data16 | Load data pointer with 16-bit constant | 3 | 3 |
| MOVC A, @A+DPTR | Move code byte relative to DPTR to accumulator | 1 | 5 |
| MOVC A, @A + PC | Move code byte relative to PC to accumulator | 1 | 4 |
| MOVX A,@Ri | Move external RAM (8-bit) to accumulator | 1 | 4 |
| MOVX A, @DPTR | Move external RAM (16-bit) to accumulator | 1 | 3 |
| MOVX @Ri, A | Move accumulator to external RAM (8-bit) | 1 | 5 |
| MOVX @DPTR, A | Move accumulator to external RAM (16-bit) | 1 | 4 |
| PUSH Direct | Push direct byte onto stack | 2 | 3 |
| POP Direct | Pop direct byte from stack | 2 | 2 |
| XCH A, Rn | Exchange register with accumulator | 1 | 2 |
| XCH A, Direct | Exchange direct byte with accumulator | 2 | 3 |
| XCH A, @Ri | Exchange indirect RAM with accumulator | 1 | 3 |
| XCHD A, @Ri | Exchange low order indirect digit RAM with accumulator | 1 | 3 |

**Table 4-4. Boolean Instructions**

| Mnemonic | Description | Bytes | Cycles |
|---|---|---|---|
| CLR C | Clear carry | 1 | 1 |
| CLR bit | Clear direct bit | 2 | 3 |
| SETB C | Set carry | 1 | 1 |
| SETB bit | Set direct bit | 2 | 3 |
| CPL C | Complement carry | 1 | 1 |
| CPL bit | Complement direct bit | 2 | 3 |
| ANL C, bit | AND direct bit to carry | 2 | 2 |
| ANL C, /bit | AND complement of direct bit to carry | 2 | 2 |
| ORL C, bit | OR direct bit to carry | 2 | 2 |
| ORL C, /bit | OR complement of direct bit to carry | 2 | 2 |
| MOV C, bit | Move direct bit to carry | 2 | 2 |
| MOV bit, C | Move carry to direct bit | 2 | 3 |
| JC rel | Jump if carry is set | 2 | 3 |
| JNC rel | Jump if no carry is set | 2 | 3 |
| JB bit, rel | Jump if direct bit is set | 3 | 5 |
| JNB bit, rel | Jump if direct bit is not set | 3 | 5 |
| JBC bit, rel | Jump if direct bit is set and clear bit | 3 | 5 |

#### *4.3.1.5 Program Branching Instructions*

The 8051 supports a set of conditional and unconditional jump instructions that help to modify the program execution flow. Table 4-5 shows the list of jump instructions.

**Table 4-5. Jump Instructions**

| Mnemonic | Description | Bytes | Cycles |
|---|---|---|---|
| ACALL addr11 | Absolute subroutine call | 2 | 4 |
| LCALL addr16 | Long subroutine call | 3 | 4 |
| RET | Return from subroutine | 1 | 4 |
| RETI | Return from interrupt | 1 | 4 |
| AJMP addr11 | Absolute jump | 2 | 3 |
| LJMP addr16 | Long jump | 3 | 4 |
| SJMP rel | Short jump (relative address) | 2 | 3 |
| JMP @A + DPTR | Jump indirect relative to DPTR | 1 | 5 |
| JZ rel | Jump if accumulator is zero | 2 | 4 |
| JNZ rel | Jump if accumulator is nonzero | 2 | 4 |
| CJNE A,Direct, rel | Compare direct byte to accumulator and jump if not equal | 3 | 5 |
| CJNE A, #data, rel | Compare immediate data to accumulator and jump if not equal | 3 | 4 |
| CJNE Rn, #data, rel | Compare immediate data to register and jump if not equal | 3 | 4 |
| CJNE @Ri, #data, rel | Compare immediate data to indirect RAM and jump if not equal | 3 | 5 |
| DJNZ Rn,rel | Decrement register and jump if not zero | 2 | 4 |
| DJNZ Direct, rel | Decrement direct byte and jump if not zero | 3 | 5 |
| NOP | No operation | 1 | 1 |

## 4.4 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.

### *4.4.1 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 4-6. PHUB Spokes and Peripherals**

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

### *4.4.2 DMA Features*

- Twenty-four DMA channels
- Each channel has one or more Transaction Descriptors (TDs) to configure channel behavior. Up to 128 total TDs can be defined
- TDs can be dynamically updated
- Eight levels of priority per channel
- Any digitally routable signal, the CPU, or another DMA channel, can trigger a transaction
- Each channel can generate up to two interrupts per transfer
- Transactions can be stalled or canceled
- Supports transaction size of infinite or 1 to 64 KB
- TDs may be nested and/or chained for complex transactions

### *4.4.3 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 percent 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 4-7 after the CPU and DMA priority levels 0 and 1 have satisfied their requirements.

**Table 4-7. Priority Levels**

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

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

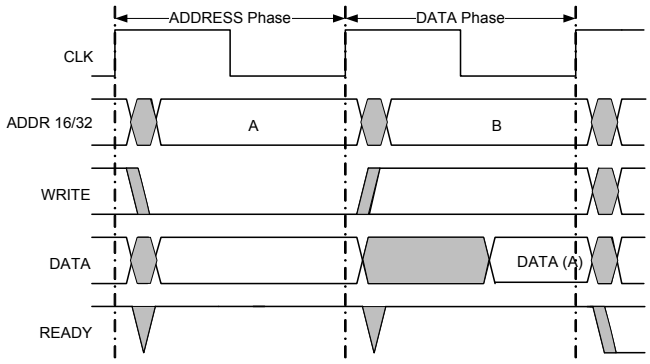
### *4.4.4 Transaction Modes Supported*

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

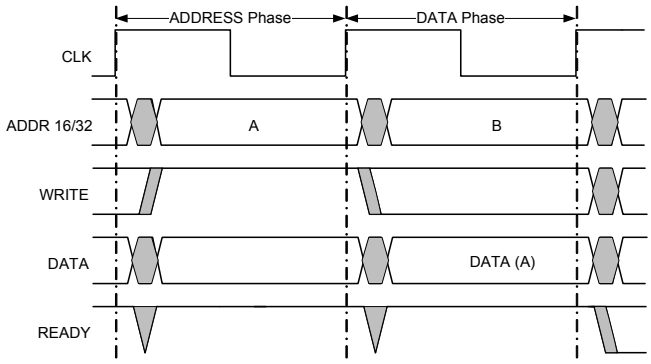
#### 4.4.4.1 Simple DMA

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 4-1. For more description on other transfer modes, refer to the Technical Reference Manual.

**Figure 4-1. DMA Timing Diagram**




Basic DMA Read Transfer without wait states




Basic DMA Write Transfer without wait states

#### 4.4.4.2 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.

#### 4.4.4.3 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.

#### 4.4.4.4 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.

#### 4.4.4.5 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.

#### 4.4.4.6 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.

#### 4.4.4.7 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.

### 4.5 Interrupt Controller

The interrupt controller provides a mechanism for hardware resources to change program execution to a new address, independent of the current task being executed by the main code. The interrupt controller provides enhanced features not found on original 8051 interrupt controllers:

- Thirty two interrupt vectors
- Jumps directly to ISR anywhere in code space with dynamic vector addresses
- Multiple sources for each vector
- Flexible interrupt to vector matching
- Each interrupt vector is independently enabled or disabled
- Each interrupt can be dynamically assigned one of eight priorities
- Eight level nestable interrupts

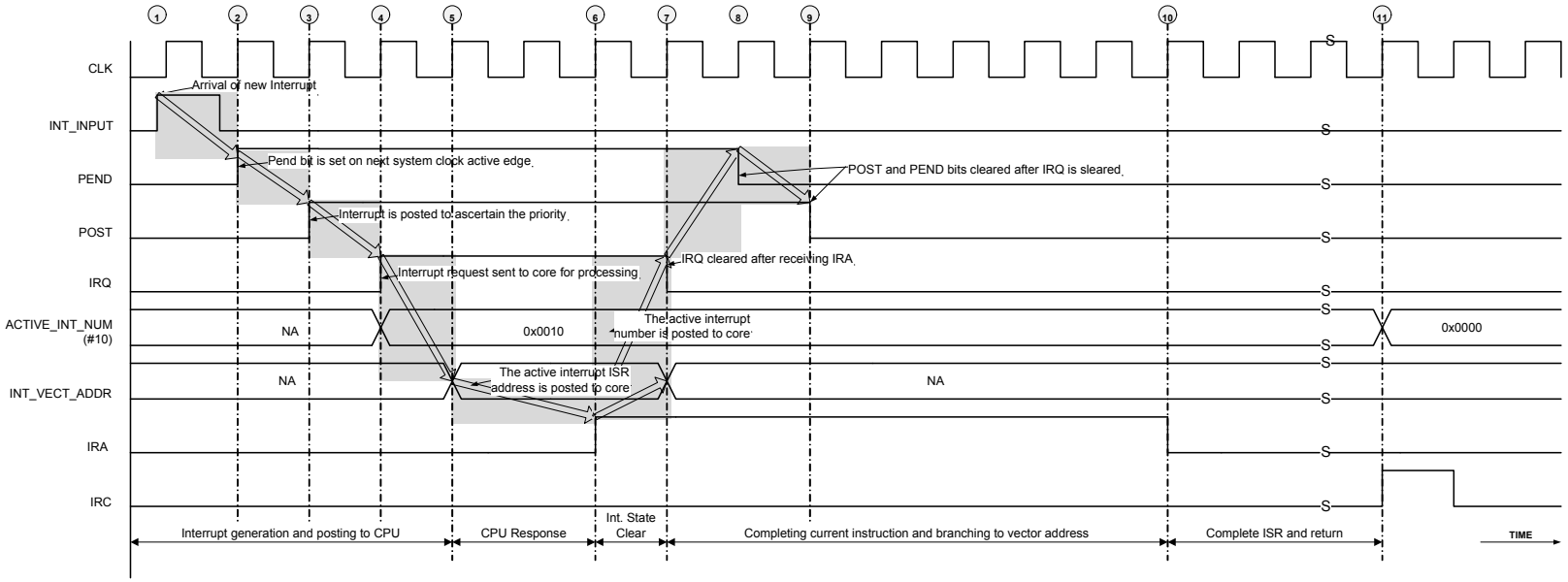
- Multiple I/O interrupt vectors
- Software can send interrupts
- Software can clear pending interrupts

When an interrupt is pending, the current instruction is completed and the program counter is pushed onto the stack. Code execution then jumps to the program address provided by the vector. After the ISR is completed, a RETI instruction is executed and returns execution to the instruction following the previously interrupted instruction. To do this the RETI instruction pops the program counter from the stack.

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. Fixed function interrupts and all interrupt sources may be routed to any interrupt vector using the UDB interrupt source connections.

Figure 4-2 on page 18 represents typical flow of events when an interrupt triggered. Figure 4-3 on page 19 shows the interrupt structure and priority polling.

**Figure 4-2. Interrupt Processing Timing Diagram**




**Notes**

1: Interrupt triggered asynchronous to the clock

2: The PEND bit is set on next active clock edge to indicate the interrupt arrival

3: POST bit is set following the PEND bit

4: Interrupt request and the interrupt number sent to CPU core after evaluation priority (Takes 3 clocks)

5: ISR address is posted to CPU core for branching

6: CPU acknowledges the interrupt request

7: ISR address is read by CPU for branching

8, 9: PEND and POST bits are cleared respectively after receiving the IRA from core

10: IRA bit is cleared after completing the current instruction and starting the instruction execution from ISR location (Takes 7 cycles)

11: IRC is set to indicate the completion of ISR, Active int. status is restored with previous status

The total interrupt latency (ISR execution)

= POST + PEND + IRQ + IRA + Completing current instruction and branching

= 1+1+1+2+7 cycles

= 12 cycles

**Figure 4-3. Interrupt Structure**

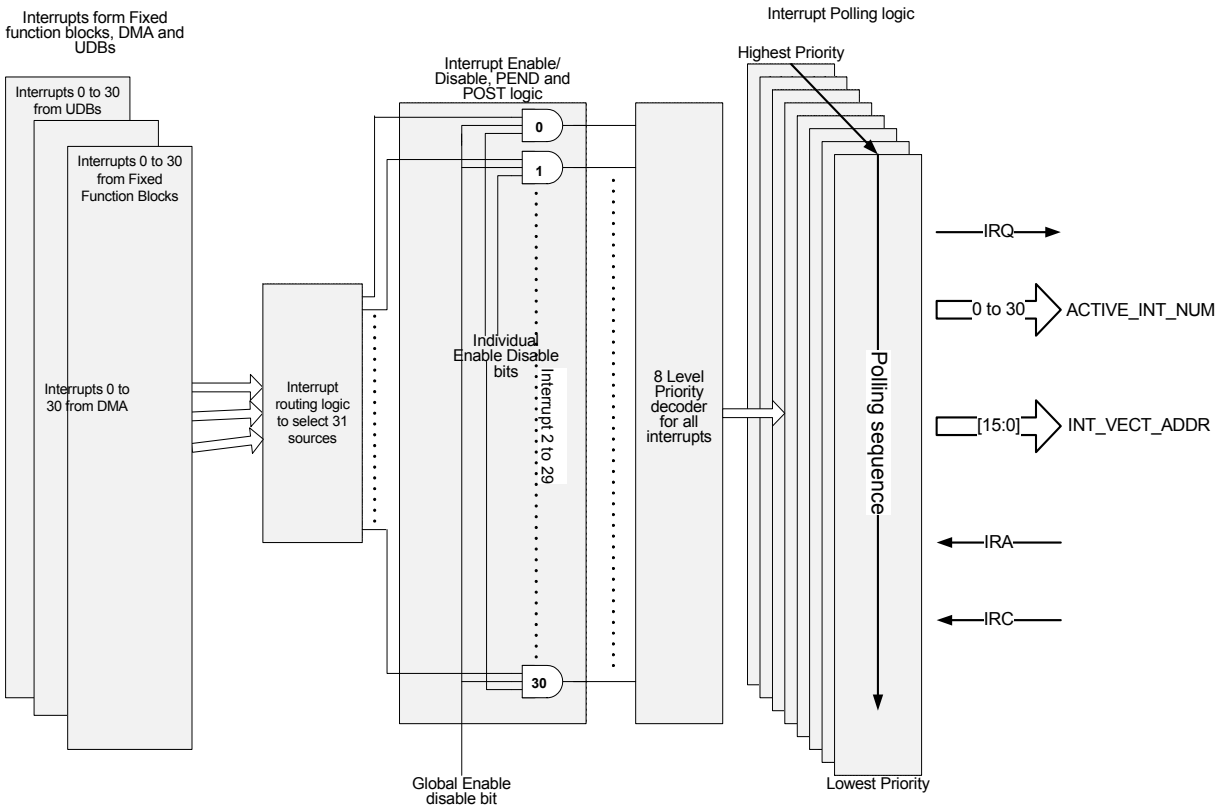


**Table 4-8. Interrupt Vector Table**

| # | Fixed Function | DMA | UDB |
|---|---|---|---|
| 0 | LVD | phub_termout0[0] | udb_intr[0] |
| 1 | ECC | phub_termout0[1] | udb_intr[1] |
| 2 | Reserved | phub_termout0[2] | udb_intr[2] |
| 3 | Sleep (Pwr Mgr) | phub_termout0[3] | udb_intr[3] |
| 4 | PICU[0] | phub_termout0[4] | udb_intr[4] |
| 5 | PICU[1] | phub_termout0[5] | udb_intr[5] |
| 6 | PICU[2] | phub_termout0[6] | udb_intr[6] |
| 7 | PICU[3] | phub_termout0[7] | udb_intr[7] |
| 8 | PICU[4] | phub_termout0[8] | udb_intr[8] |
| 9 | PICU[5] | phub_termout0[9] | udb_intr[9] |
| 10 | PICU[6] | phub_termout0[10] | udb_intr[10] |
| 11 | PICU[12] | phub_termout0[11] | udb_intr[11] |
| 12 | PICU[15] | phub_termout0[12] | udb_intr[12] |
| 13 | Comparators Combined | phub_termout0[13] | udb_intr[13] |
| 14 | Switched Caps Combined | phub_termout0[14] | udb_intr[14] |
| 15 | $I^2C$ | phub_termout0[15] | udb_intr[15] |
| 16 | CAN | phub_termout1[0] | udb_intr[16] |
| 17 | Timer/Counter0 | phub_termout1[1] | udb_intr[17] |
| 18 | Timer/Counter1 | phub_termout1[2] | udb_intr[18] |
| 19 | Timer/Counter2 | phub_termout1[3] | udb_intr[19] |
| 20 | Timer/Counter3 | phub_termout1[4] | udb_intr[20] |
| 21 | USB SOF Int | phub_termout1[5] | udb_intr[21] |
| 22 | USB Arb Int | phub_termout1[6] | udb_intr[22] |
| 23 | USB Bus Int | phub_termout1[7] | udb_intr[23] |
| 24 | USB Endpoint[0] | phub_termout1[8] | udb_intr[24] |
| 25 | USB Endpoint Data | phub_termout1[9] | udb_intr[25] |
| 26 | Reserved | phub_termout1[10] | udb_intr[26] |
| 27 | LCD | phub_termout1[11] | udb_intr[27] |
| 28 | Reserved | phub_termout1[12] | udb_intr[28] |
| 29 | Decimator Int | phub_termout1[13] | udb_intr[29] |
| 30 | PHUB Error Int | phub_termout1[14] | udb_intr[30] |
| 31 | EEPROM Fault Int | phub_termout1[15] | udb_intr[31] |

# 5. Memory

## 5.1 Static RAM

CY8C34 Static RAM (SRAM) is used for temporary data storage. Up to 8 KB of SRAM is provided and can be accessed by the 8051 or the DMA controller. See Memory Map on page 23. Simultaneous access of SRAM by the 8051 and the DMA controller is possible if different 4-KB blocks are accessed.

## 5.2 Flash Program Memory

Flash memory in PSoC devices provides nonvolatile storage for user firmware, user configuration data, bulk data storage, and optional ECC data. The main flash memory area contains up to 64 KB of user program space.

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

Flash is read in units of rows; each row is 9 bytes wide with 8 bytes of data and 1 byte of ECC data. When a row is read, the data bytes are copied into an 8-byte instruction buffer. The CPU fetches its instructions from this buffer, for improved CPU performance.

Flash programming is performed through a special interface and preempts code execution out of flash. The flash programming interface performs flash erasing, programming and setting code protection levels. Flash in-system serial programming (ISSP), typically used for production programming, is possible through both the SWD and JTAG interfaces. In-system programming, typically used for bootloaders, is also possible using serial interfaces such as $I^2C$, USB, UART, and SPI, or any communications protocol.

## 5.3 Flash Security

All PSoC devices include a flexible flash-protection model that prevents access and visibility to on-chip flash memory. This prevents duplication or reverse engineering of proprietary code. Flash memory is organized in blocks, where each block contains 256 bytes of program or data and 32 bytes of ECC or configuration data. A total of up to 256 blocks is provided on 64-KB flash devices.

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 63). For more information about how to take full advantage of the security features in PSoC, see the PSoC 3 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 CY8C34 has up to 2 KB of EEPROM memory to store user data. Reads from EEPROM are random access at the byte level. Reads are done directly; writes are done by sending write commands to an EEPROM programming interface. CPU code execution can continue from flash during EEPROM writes. EEPROM is erasable and writeable at the row level. The EEPROM is divided into 128 rows of 16 bytes each.

The CPU can not execute out of EEPROM. There is no ECC hardware associated with EEPROM. If ECC is required it must be handled in firmware.

## 5.5 Nonvolatile Latches (NVLs)

PSoC has a 4-byte array of nonvolatile latches (NVLs) that are used to configure the device at reset. The NVL register map is shown in Table 5-2.

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

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

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

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

| Field | Description | Settings |
|---|---|---|
| PRTxRDM[1:0] | Controls reset drive mode of the corresponding IO port. See "Reset Configuration" on page 38. All pins of the port are set to the same mode. | 00b (default) - high impedance analog<br>01b - high impedance digital<br>10b - resistive pull up<br>11b - resistive pull down |
| XRESMEN | Controls whether pin P1[2] is used as a GPIO or as an external reset. See "Pin Descriptions" on page 10, XRES description. | 0 (default for 68-pin and 100-pin parts) - GPIO<br>1 (default for 48-pin parts) - external reset |
| DPS{1:0] | Controls the usage of various P1 pins as a debug port. See "Programming, Debug Interfaces, Resources" on page 60. | 00b - 5-wire JTAG<br>01b (default) - 4-wire JTAG<br>10b - SWD<br>11b - debug ports disabled |
| ECCEN | Controls whether ECC flash is used for ECC or for general configuration and data storage. See "Flash Program Memory" on page 21. | 0 (default) - ECC disabled<br>1 - ECC enabled |
| DIG_PHS_DLY[3:0] | Selects the digital clock phase delay. | See the TRM for details. |

Although PSoC Creator provides support for modifying the device configuration NVLs, the number of NVL erase / write cycles is limited – see "Nonvolatile Latches (NVL))" on page 103.

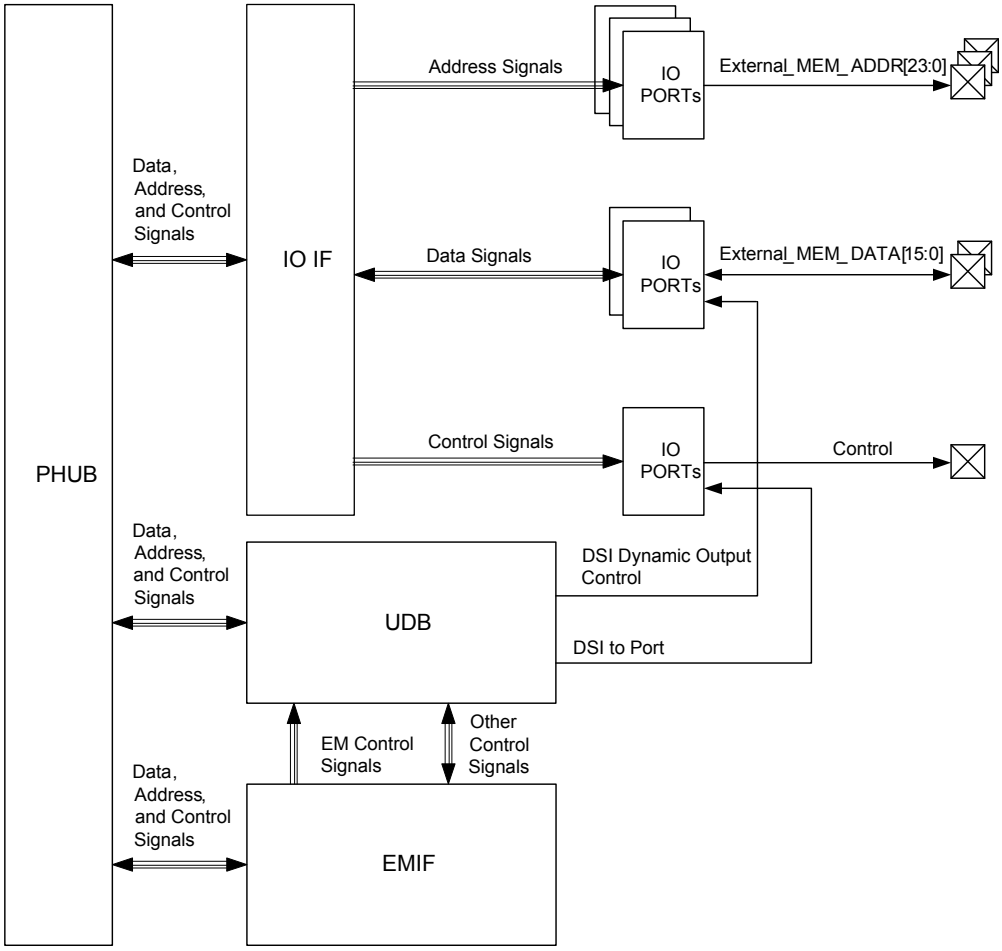
## 5.6 External Memory Interface

CY8C34 provides an External Memory Interface (EMIF) for connecting to external memory devices. The connection allows read and write accesses to external memories. The EMIF operates in conjunction with UDBs, I/O ports, and other hardware to generate external memory address and control signals. At 33 MHz, each memory access cycle takes four bus clock cycles.

Figure 5-1 is the EMIF block diagram. The EMIF supports synchronous and asynchronous memories. The CY8C34 supports only one type of external memory device at a time.

External memory can be accessed via the 8051 xdata space; up to 24 address bits can be used. See "xdata Space" section on page 25. The memory can be 8 or 16 bits wide.

**Figure 5-1. EMIF Block Diagram**




## 5.7 Memory Map

The CY8C34 8051 memory map is very similar to the MCS-51 memory map.

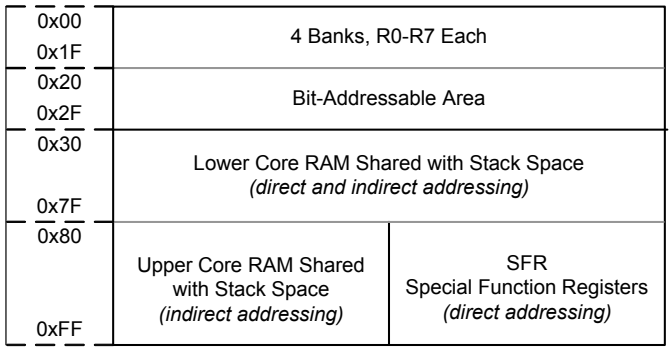
### *5.7.1 Code Space*

The CY8C34 8051 code space is 64 KB. Only main flash exists in this space. See the "Flash Program Memory" section on page 21.

### *5.7.2 Internal Data Space*

The CY8C34 8051 internal data space is 384 bytes, compressed within a 256-byte space. This space consists of 256 bytes of RAM (in addition to the SRAM mentioned in Static RAM on page 21) and a 128-byte space for Special Function Registers (SFRs). See Figure 5-2. The lowest 32 bytes are used for 4 banks of registers R0-R7. The next 16 bytes are bit-addressable.

**Figure 5-2. 8051 Internal Data Space**




In addition to the register or bit address modes used with the lower 48 bytes, the lower 128 bytes can be accessed with direct or indirect addressing. With direct addressing mode, the upper 128 bytes map to the SFRs. With indirect addressing mode, the upper 128 bytes map to RAM. Stack operations use indirect addressing; the 8051 stack space is 256 bytes. See the "Addressing Modes" section on page 11

### *5.7.3 SFRs*

The Special Function Register (SFR) space provides access to frequently accessed registers. The memory map for the SFR memory space is shown in Table 5-4.

**Table 5-4. SFR Map**

| Address | 0/8 | 1/9 | 2/A | 3/B | 4/C | 5/D | 6/E | 7/F |
|---|---|---|---|---|---|---|---|---|
| 0×F8 | SFRPRT15DR | SFRPRT15PS | SFRPRT15SEL | | | | | |
| 0×F0 | B | | SFRPRT12SEL | | | | | |
| 0×E8 | SFRPRT12DR | SFRPRT12PS | MXAX | | | | | |
| 0×E0 | ACC | | | | | | | |
| 0×D8 | SFRPRT6DR | SFRPRT6PS | SFRPRT6SEL | | | | | |
| 0×D0 | PSW | | | | | | | |
| 0×C8 | SFRPRT5DR | SFRPRT5PS | SFRPRT5SEL | | | | | |
| 0×C0 | SFRPRT4DR | SFRPRT4PS | SFRPRT4SEL | | | | | |
| 0×B8 | | | | | | | | |
| 0×B0 | SFRPRT3DR | SFRPRT3PS | SFRPRT3SEL | | | | | |
| 0×A8 | IE | | | | | | | |
| 0×A0 | P2AX | | SFRPRT1SEL | | | | | |
| 0×98 | SFRPRT2DR | SFRPRT2PS | SFRPRT2SEL | | | | | |
| 0×90 | SFRPRT1DR | SFRPRT1PS | | DPX0 | | DPX1 | | |
| 0×88 | | SFRPRT0PS | SFRPRT0SEL | | | | | |
| 0×80 | SFRPRT0DR | SP | DPL0 | DPH0 | DPL1 | DPH1 | DPS | |

The CY8C34 family provides the standard set of registers found on industry standard 8051 devices. In addition, the CY8C34 devices add SFRs to provide direct access to the I/O ports on the device. The following sections describe the SFRs added to the CY8C34 family.

#### XData Space Access SFRs

The 8051 core features dual DPTR registers for faster data transfer operations. The data pointer select SFR, DPS, selects which data pointer register, DPTR0 or DPTR1, is used for the following instructions:

- MOVX @DPTR, A
- MOVX A, @DPTR
- MOVC A, @A+DPTR
- JMP @A+DPTR
- INC DPTR
- MOV DPTR, #data16

The extended data pointer SFRs, DPX0, DPX1, MXAX, and P2AX, hold the most significant parts of memory addresses during access to the xdata space. These SFRs are used only with the MOVX instructions.

During a MOVX instruction using the DPTR0/DPTR1 register, the most significant byte of the address is always equal to the contents of DPX0/DPX1. During a MOVX instruction using the R0 or R1 register, the most significant byte of the address is always equal to the contents of MXAX, and the next most significant byte is always equal to the contents of P2AX.

#### I/O Port SFRs

The I/O ports provide digital input sensing, output drive, pin interrupts, connectivity for analog inputs and outputs, LCD, and

access to peripherals through the DSI. Full information on I/O ports is found in I/O System and Routing on page 32.

I/O ports are linked to the CPU through the PHUB and are also available in the SFRs. Using the SFRs allows faster access to a limited set of I/O port registers, while using the PHUB allows boot configuration and access to all I/O port registers.

Each SFR supported I/O port provides three SFRs:

- SFRPRTxDR sets the output data state of the port (where x is port number and includes ports 0-6, 12 and 15)
- The SFRPRTxSEL selects whether the PHUB PRTxDR register or the SFRPRTxDR controls each pin's output buffer within the port. If a SFRPRTxSEL[y] bit is high, the corresponding SFRPRTxDR[y] bit sets the output state for that pin. If a SFRPRTxSEL[y] bit is low, the corresponding PRTxDR[y] bit sets the output state of the pin (where y varies from 0 to 7).
- The SFRPRTxPS is a read only register that contains pin state values of the port pins.

#### *5.7.3.1 xdata Space*

The 8051 xdata space is 24-bit, or 16 MB in size. The majority of this space is not "external"—it is used by on-chip components. See Table 5-5. External, that is, off-chip, memory can be accessed using the EMIF. See External Memory Interface on page 23.

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

| Address Range | Purpose |
|---|---|
| 0×00 0000 – 0×00 1FFF | SRAM |
| 0×00 4000 – 0×00 42FF | Clocking, PLLs, and oscillators |
| 0×00 4300 – 0×00 43FF | Power management |
| 0×00 4400 – 0×00 44FF | Interrupt controller |
| 0×00 4500 – 0×00 45FF | Ports interrupt control |
| 0×00 4700 – 0×00 47FF | Flash programming interface |
| 0×00 4900 – 0×00 49FF | $I^2C$ controller |
| 0×00 4E00 – 0×00 4EFF | Decimator |
| 0×00 4F00 – 0×00 4FFF | Fixed timer/counter/PWMs |
| 0×00 5000 – 0×00 51FF | I/O ports control |
| 0×00 5400 – 0×00 54FF | External Memory Interface (EMIF) control registers |
| 0×00 5800 – 0×00 5FFF | Analog Subsystem interface |
| 0×00 6000 – 0×00 60FF | USB controller |
| 0×00 6400 – 0×00 6FFF | UDB configuration |
| 0×00 7000 – 0×00 7FFF | PHUB configuration |
| 0×00 8000 – 0×00 8FFF | EEPROM |
| 0×00 A000 – 0×00 A400 | CAN |
| 0×01 0000 – 0×01 FFFF | Digital Interconnect configuration |
| 0×05 0220 – 0×05 02F0 | Debug controller |
| 0×08 0000 – 0×08 1FFF | Flash ECC bytes |
| 0×80 0000 – 0×FF FFFF | External Memory Interface |

# 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 50 MHz clock, accurate to ±1 percent 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 the user wants, 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 you to build clocking systems with minimal input. You can specify desired clock frequencies and accuracies, and the software locates or builds a clock that meets the required specifications. This is possible because of the programmability inherent PSoC.

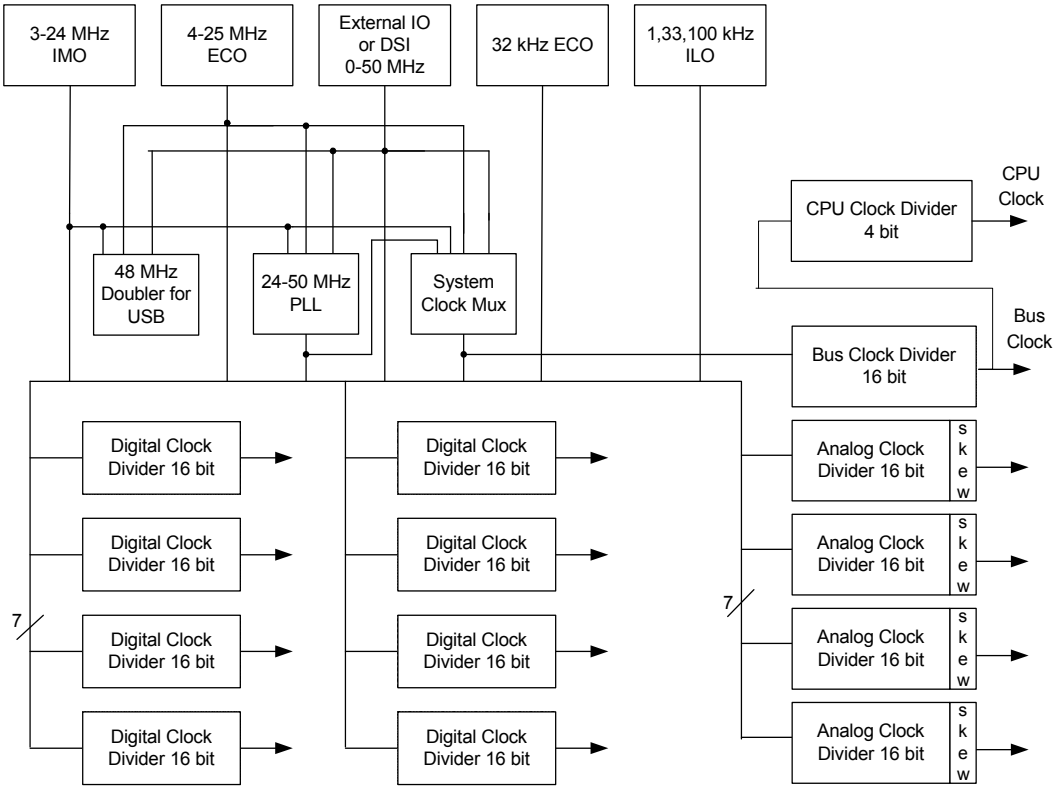
Key features of the clocking system include:

- Seven general purpose clock sources
  - 3- to 24-MHz IMO, ±1 percent 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 28
  - DSI signal from an external I/O pin or other logic
  - 24- to 50- MHz fractional PLL sourced from IMO, MHzECO, or DSI
  - 1 kHz, 33 kHz, 100 kHz ILO for Watch Dog Timer (WDT) and Sleep Timer
  - 32.768-kHz external crystal oscillator (kHzECO) for RTC
- IMO has a USB mode that auto locks to the USB bus clock requiring no external crystal for USB. (USB equipped parts only)
- Independently sourced clock in all clock dividers
- Eight 16-bit clock dividers for the digital system
- Four 16-bit clock dividers for the analog system
- Dedicated 16-bit divider for the bus clock
- Dedicated 4-bit divider for the 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 | ±1% over voltage and temperature | 24 MHz | ±4% | 10 µs max |
| MHzECO | 4 MHz | Crystal dependent | 25 MHz | Crystal dependent | 5 ms typ, max is crystal dependent |
| DSI | 0 MHz | Input dependent | 50 MHz | Input dependent | Input dependent |
| PLL | 24 MHz | Input dependent | 50 MHz | Input dependent | 250 µs max |
| Doubler | 48 MHz | Input dependent | 48 MHz | Input dependent | 1 µs max |
| ILO | 1 kHz | –50%, +100% | 100 kHz | –55%, +100% | 15 ms max in lowest power mode |
| kHzECO | 32 kHz | Crystal dependent | 32 kHz | Crystal dependent | 500 ms typ, max is crystal dependent |

**Figure 6-1. Clocking Subsystem**




### 6.1.1 Internal Oscillators

#### 6.1.1.1 Internal Main Oscillator

In most designs the IMO is the only clock source required, due to its ±1-percent 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 ±1 percent at 3 MHz, up to ±4 percent 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 Phase-Locked Loop).

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 trade off 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 50 MHz. Its input and feedback dividers supply 4032 discrete ratios to create almost any desired system clock frequency. The accuracy of the PLL output depends on the accuracy of the PLL input source. The most common PLL use is to multiply the IMO clock at 3 MHz, where it is most accurate to generate the CPU and system clocks up to the device's maximum frequency.

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

#### *6.1.1.4 Internal Low-Speed Oscillator*

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

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

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

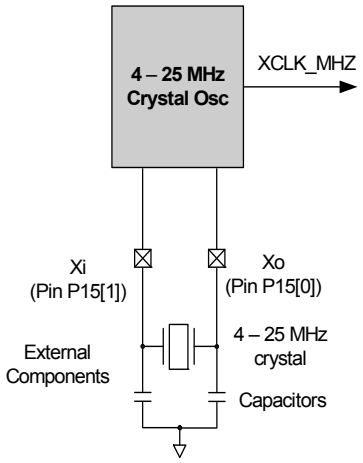
The 33 kHz clock (CLK33K) comes from a divide-by-3 operation on CLK100K. This output can be used as a reduced accuracy version of the 32.768-kHz ECO clock with no need for a crystal.

### *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 "Phase-Locked Loop" section on page 26). The GPIO pins connecting to the external crystal and capacitors are fixed. MHzECO accuracy depends on the crystal chosen.

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

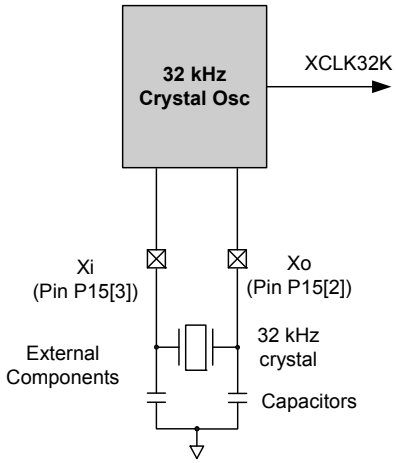



#### *6.1.2.2 32.768-kHz ECO*

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

The oscillator works in two distinct power modes. This allows users 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 Universal Digital Blocks.

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. Bus clock is the source clock for the CPU clock divider.
- 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 ADC and mixers. 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 percent 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.

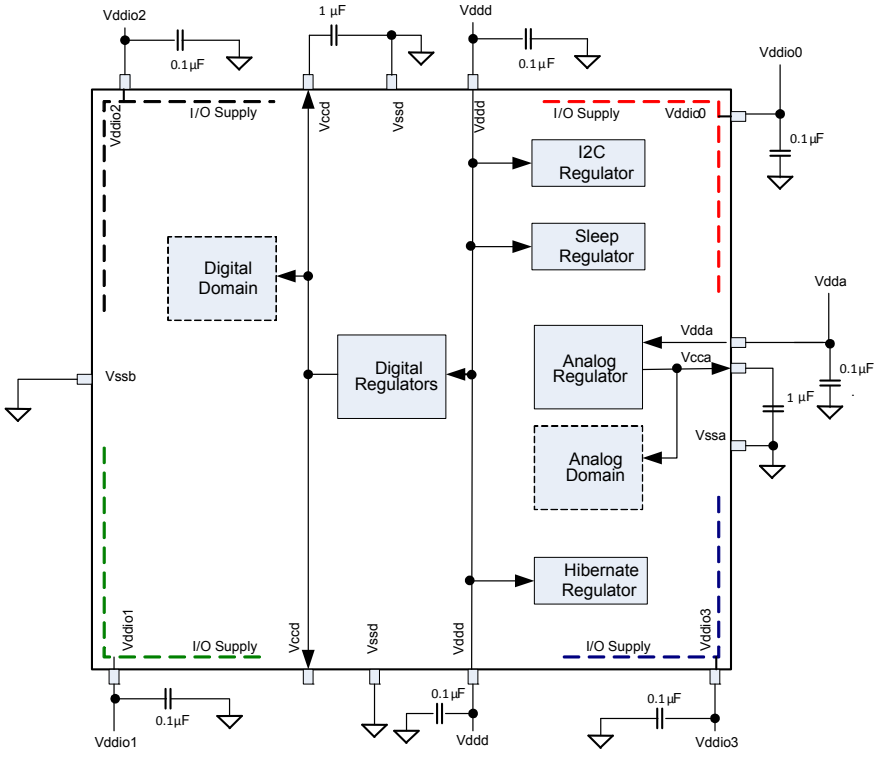
### 6.1.4 USB Clock Domain

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 can be generated from different sources, including DSI clock at 48 MHz or doubled value of 24 MHz from internal oscillator, DSI signal, or crystal oscillator.

## 6.2 Power System

The power system consists of separate analog, digital, and I/O supply pins, labeled Vdda, Vddd, and Vddiox, respectively. It also includes two internal 1.8 V regulators that provide the digital (Vccd) and analog (Vcca) supplies for the internal core logic. The output pins of the regulators (Vccd and Vcca) and the Vddio pins must have capacitors connected as shown in Figure 6-4. The two Vccd pins must be shorted together, with as short a trace as possible, and connected to a 1 µF ±10 percent X5R capacitor. The power system also contains a sleep regulator, an $I^2C$ regulator, and a hibernate regulator.

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



**Note** The two Vccd pins must be connected together with as short a trace as possible. A trace under the device is recommended, as shown in Figure 2-6 on page 10.

### 6.2.1 Power Modes

PSoC 3 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 3 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 RTC 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 30 illustrates the allowable transitions between power modes.

**Table 6-2. Power Modes**

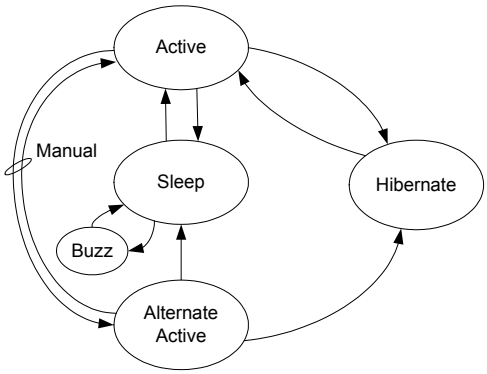
| Power Modes | Description | Entry Condition | Wakeup Source | Active Clocks | Regulator |
|---|---|---|---|---|---|
| Active | Primary mode of operation, all peripherals available (programmable) | Wakeup, reset, manual register entry | Any interrupt | Any (programmable) | All regulators available. 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 | Comparator, PICU, $I^2C$, 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<br>Lowest power consuming mode with all peripherals and internal regulators disabled, except hibernate regulator is enabled<br>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 | – | 1.2 mA[13] | Yes | All | All | All | – | All |
| Alternate Active | – | – | User defined | All | All | All | – | All |
| Sleep | <15 µs | 1 µA | No | $I^2C$ | Comparator | ILO/kHzECO | Comparator, PICU, $I^2C$, RTC, CTW, LVD | XRES, LVD, WDR |
| Hibernate | <100 µs | 200 nA | No | None | None | None | PICU | XRES |

**Note**
13. Bus clock off. Execute from CPU instruction buffer at 6 MHz. See Table 11-2 on page 66.

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




#### 6.2.1.1 Active Mode

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

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

#### 6.2.1.2 Alternate Active Mode

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

#### 6.2.1.3 Sleep Mode

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

#### 6.2.1.4 Hibernate Mode

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

#### 6.2.1.5 Wakeup Events

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

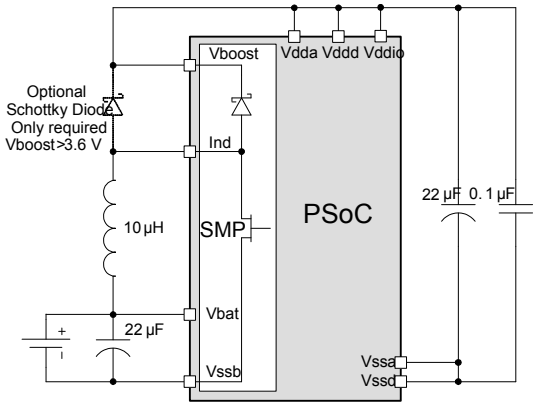
### 6.2.2 Boost Converter

Applications that use a supply voltage of less than 1.71 V, such as solar 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 0.5 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 from 0.5 V to 5.5 V (Vbat), and can start up with Vbat as low as 0.5 V. The converter provides a user configurable output voltage of 1.8 to 5.0 V (Vboost). Vbat is typically less than Vboost; if Vbat is greater than or equal to Vboost, then $V_{BOOST}$ will be the same as Vbat. The block can deliver up to 50 mA ($I_{BOOST}$) depending on configuration.

Four pins are associated with the boost converter: Vbat, Vssb, Vboost, and Ind. The boosted output voltage is sensed at the Vboost pin and must be connected directly to the chip's supply inputs. An inductor is connected between the Vbat and Ind pins. You 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 Vboost > 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 Vboost output is limited to 2 × Vbat, and at 400 kHz Vboost is limited to 4 × Vbat.

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 |

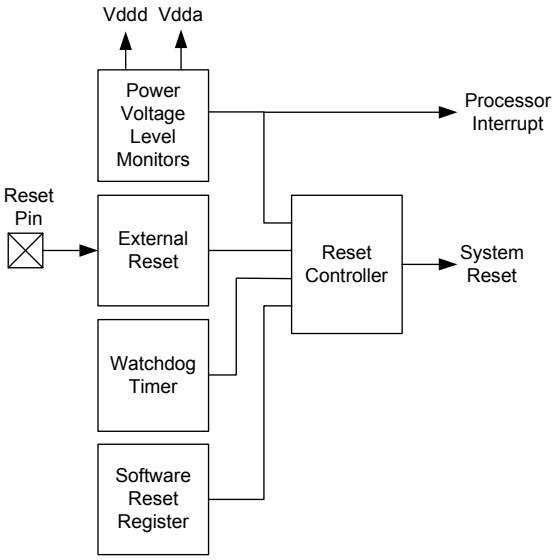
If the boost converter is not used in a given application, tie the $V_{BAT}$, $V_{SSB}$, and $V_{BOOST}$ pins to ground and leave the Ind pin unconnected.

## 6.3 Reset

CY8C34 has multiple internal and external reset sources available. The reset sources are:

- Power source monitoring – The analog and digital power voltages, Vdda, Vddd, Vcca, and Vccd are monitored in several different modes during power up, active mode, and sleep mode (buzzing). If any of the voltages goes outside predetermined ranges then a reset is generated. The monitors are programmable to generate an interrupt to the processor under certain conditions before reaching the reset thresholds.
- External – The device can be reset from an external source by pulling the reset pin ($\overline{XRES}$) low. The $\overline{XRES}$ pin includes an internal pull-up to Vddio1. Vddd, Vdda, and Vddio1 must all have voltage applied before the part comes out of reset.
- Watchdog timer – A watchdog timer monitors the execution of instructions by the processor. If the watchdog timer is not reset by firmware within a certain period of time, the watchdog timer generates a reset.
- Software – The device can be reset under program control.

**Figure 6-7. Resets**




The term **device 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 Vddd and Vdda, 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. Voltage supervision is then handed off to the precise low voltage reset (PRES) circuit. When the voltage is high enough for PRES to release, the IMO starts.

- PRES – Precise Low Voltage Reset

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

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

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

  Interrupt circuits are available to detect when $V_{DDA}$ and $V_{DDD}$ go outside a voltage range. For AHVI, $V_{DDA}$ is compared to a fixed trip level. For ALVI and DLVI, $V_{DDA}$ and $V_{DDD}$ 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 | $V_{DDD}$ | 1.71 V to 5.5 V | 1.70 V to 5.45 V in 250 mV increments | ±2% |
| ALVI | $V_{DDA}$ | 1.71 V to 5.5 V | 1.70 V to 5.45 V in 250 mV increments | ±2% |
| AHVI | $V_{DDA}$ | 1.71 V to 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 wake up sequence. The interrupt is then recognized and may be serviced.

*6.3.1.2 Other Reset Sources*

- XRES – External Reset

  PSoC 3 has either a single GPIO pin that is configured as an external reset or a dedicated XRES pin. Either the dedicated XRES pin or the GPIO pin, if configured, 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.

## 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 $V_{DDIO}$ 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[14], and LCD segment drive, while SIO pins are used for voltages in excess of $V_{DDA}$ and for programmable output voltages.

- Features supported by both GPIO and SIO:
  - User programmable port reset state
  - Separate I/O supplies and voltages for up to four groups of I/O
  - Digital peripherals use DSI to connect the pins
  - Input or output or both for CPU and DMA
  - Eight drive modes
  - Every pin can be an interrupt source configured as rising edge, falling edge or both edges. If required, level sensitive interrupts are supported through the DSI
  - Dedicated port interrupt vector for each port
  - Slew rate controlled digital output drive mode
  - Access port control and configuration registers on either port basis or pin basis
  - Separate port read (PS) and write (DR) data registers to avoid read modify write errors
  - Special functionality on a pin by pin basis
- Additional features only provided on the GPIO pins:
  - LCD segment drive on LCD equipped devices
  - CapSense[14]
  - Analog input and output capability
  - Continuous 100 µA clamp current capability
  - Standard drive strength down to 1.7 V
- Additional features only provided on SIO pins:
  - Higher drive strength than GPIO
  - Hot swap capability (5 V tolerance at any operating $V_{DD}$)
  - Programmable and regulated high input and output drive levels down to 1.2 V
  - No analog input, CapSense, or LCD capability
  - Over voltage tolerance up to 5.5 V
  - SIO can act as a general purpose analog comparator

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

- 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

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

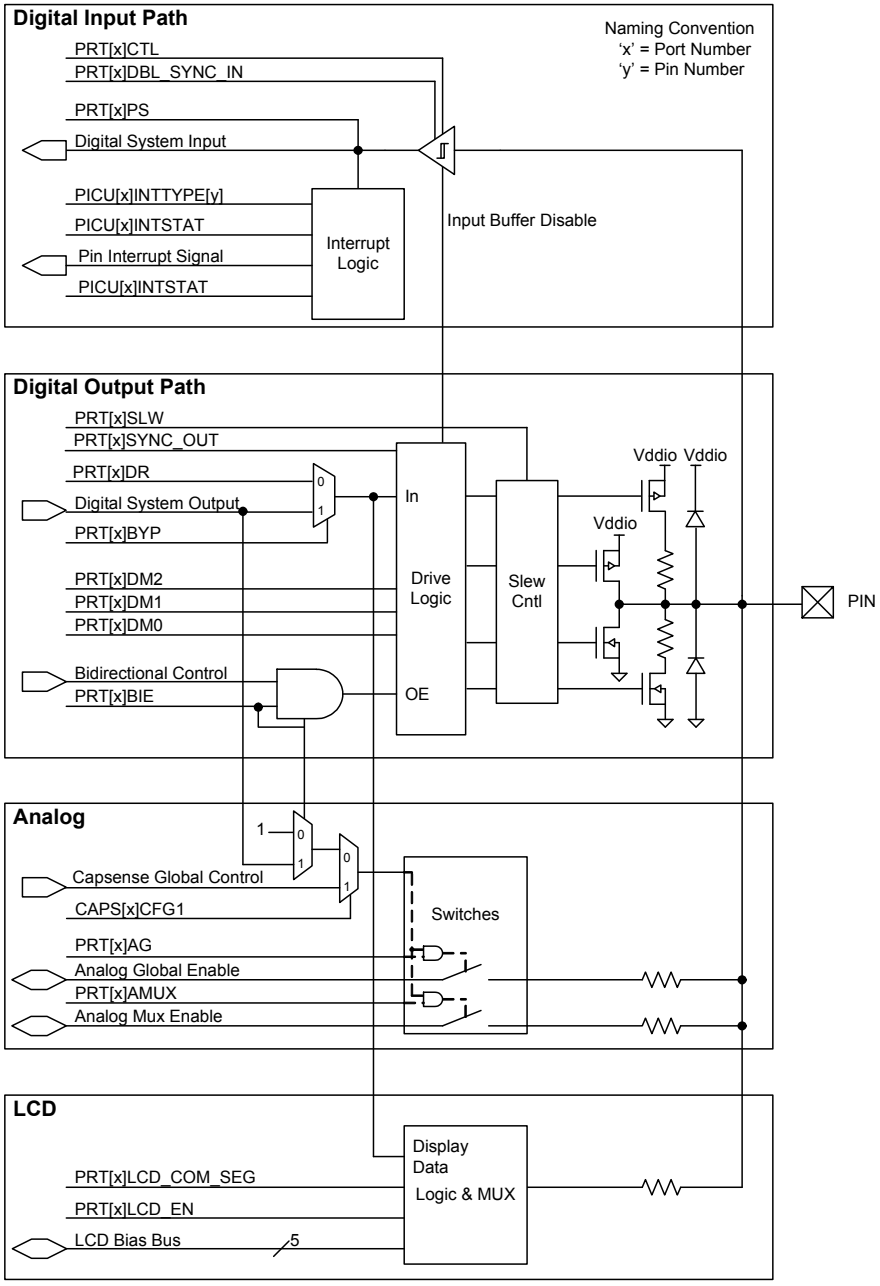


**Figure 6-9. SIO Input/Output Block Diagram**

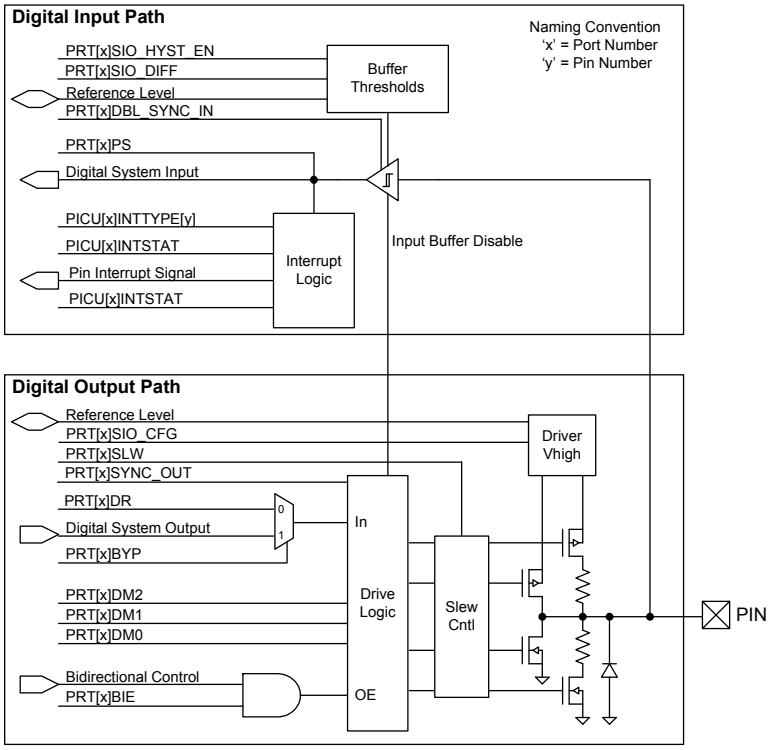
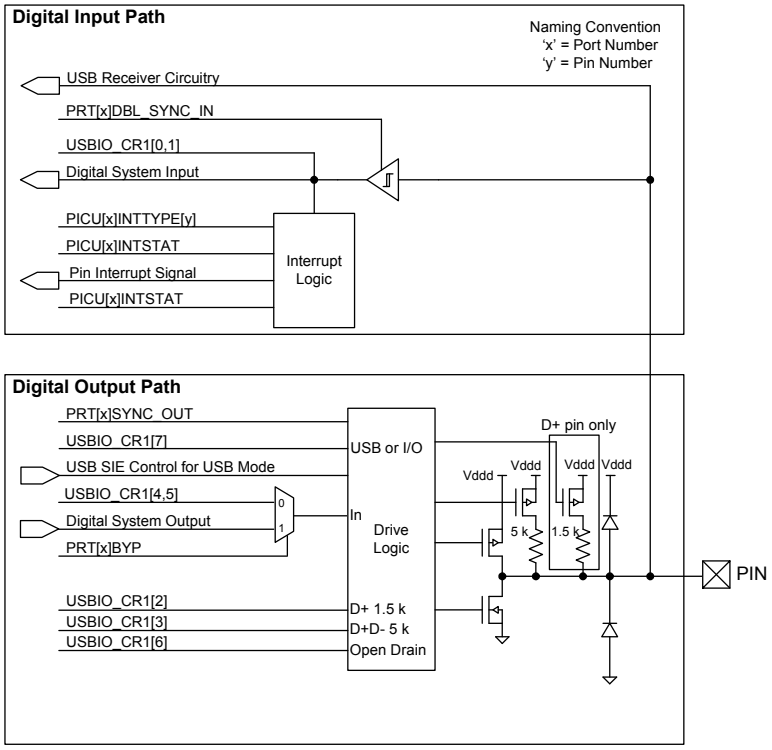



**Figure 6-10. USBIO Block Diagram**



### 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**

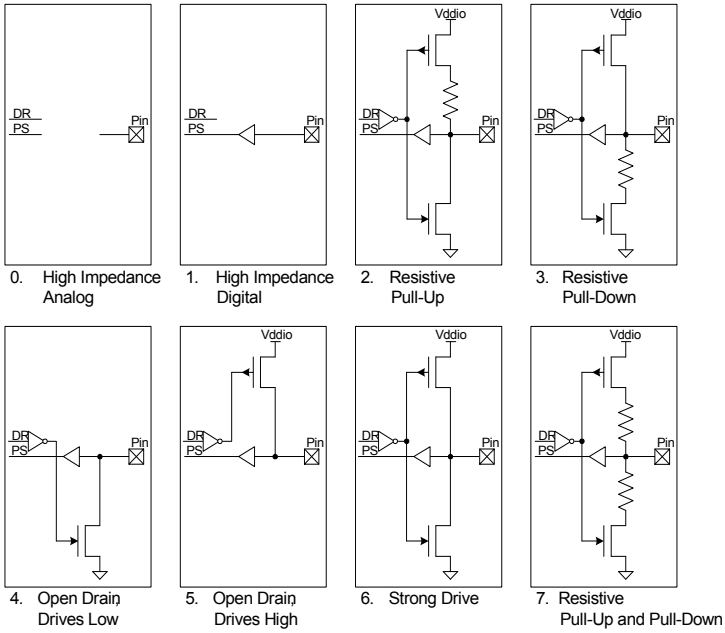



**Table 6-6. Drive Modes**

| Diagram | Drive Mode | PRT×DM2 | PRT×DM1 | PRT×DM0 | PRT×DR = 1 | PRT×DR = 0 |
|---|---|---|---|---|---|---|
| 0 | High impedence analog | 0 | 0 | 0 | High Z | High Z |
| 1 | High Impedance digital | 0 | 0 | 1 | High Z | High Z |
| 2 | Resistive pull-up[15] | 0 | 1 | 0 | Res High (5K) | Strong Low |
| 3 | Resistive pull-down[15] | 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[15] | 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 (High Z) state recommended for digital inputs.

**Note**
15. 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 $I^2C$ 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 PRT×DM[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; Universal Digital Blocks (UDB) provide this functionality to the system when needed.

### *6.4.6 Input Buffer Mode*

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

### *6.4.7 I/O Power Supplies*

Up to four I/O pin power supplies are provided depending on the device and package. Each I/O supply must be less than or equal to the voltage on the chip’s analog ($V_{DDA}$) pin. This feature allows users 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, select pins provide direct connections to specific analog features such as the high current DACs or uncommitted opamps.

### *6.4.9 CapSense*

This section applies only to GPIO pins. All GPIO pins may be used to create CapSense buttons and sliders[16]. See the “CapSense” section on page 59 for more information.

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

### 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 58 for details.

### 6.4.11 Adjustable Output Level

This section applies only to SIO pins. SIO port pins support the ability to provide a regulated high output level for interface to external signals that are lower in voltage than the SIO's respective Vddio. SIO pins are individually configurable to output either the standard $V_{DDIO}$ level or the regulated output, which is based on an internally generated reference. Typically a voltage DAC (VDAC) is used to generate the reference (see Figure 6-12). The "DAC" section on page 59 has more details on VDAC use and reference routing to the SIO pins. Resistive pull-up and pull-down drive modes are not available with SIO in regulated output mode.

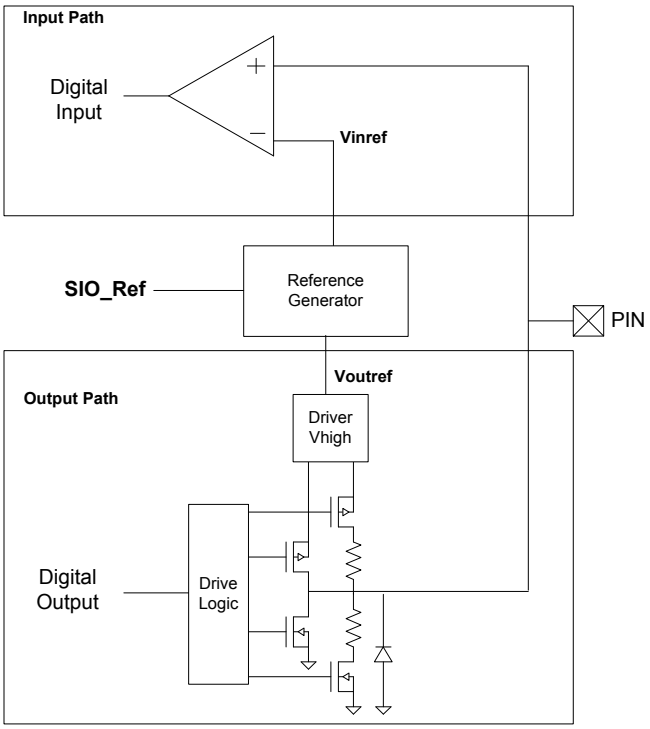
### 6.4.12 Adjustable Input Level

This section applies only to SIO pins. SIO pins by default support the standard CMOS and LVTTL input levels but also support a differential mode with programmable levels. SIO pins are grouped into pairs. Each pair shares a reference generator block which, is used to set the digital input buffer reference level for interface to external signals that differ in voltage from $V_{DDIO}$. 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 a voltage DAC (VDAC) generates the $V_{REF}$ reference. "DAC" section on page 59 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 34 illustrates this functionality. In the figure, 'Reference level' is the analog signal routed through the analog global. The hysteresis feature can also be enabled for the input buffer of the SIO, which increases noise immunity for the comparator.

### 6.4.14 Hot Swap

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

### 6.4.15 Over Voltage Tolerance

All I/O pins provide an over voltage 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 where Vddio ≤ $V_{IN}$ ≤ 5.5 V.
- The GPIO pins must be limited to 100 µA using a current limiting resistor. GPIO pins clamp the pin voltage to approximately one diode above the Vddio supply where Vddio ≤ $V_{IN}$ ≤ $V_{DDA}$.
- In case of a GPIO pin configured for analog input/output, the analog voltage on the pin must not exceed the Vddio supply voltage to which the GPIO belongs.

A common application for this feature is connection to a bus such as $I^2C$ where different devices are running from different supply voltages. In the $I^2C$ 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 $I^2C$ bus voltage above the PSoC pin supply. For example, the PSoC chip could operate at 1.8 V, and an external device could run from 5 V. Note that the SIO pin's $V_{IH}$ and $V_{IL}$ levels are determined by the associated Vddio 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

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

### 6.4.17 Low-Power Functionality

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

### 6.4.18 Special Pin Functionality

Some pins on the device include additional special functionality in addition to their GPIO or SIO functionality. The specific special function pins are listed in Pinouts on page 5. The special features are:

- Digital
  - 4 to 25 MHz crystal oscillator
  - 32.768-kHz crystal oscillator
  - Wake from sleep on $I^2C$ address match. Any pin can be used for $I^2C$ if wake from sleep is not required.
  - JTAG interface pins
  - SWD interface pins
  - SWV interface pins
  - External reset
- Analog
  - Opamp inputs and outputs
  - High current IDAC outputs
  - External reference inputs

### 6.4.19 JTAG Boundary Scan

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

## 7. Digital Subsystem

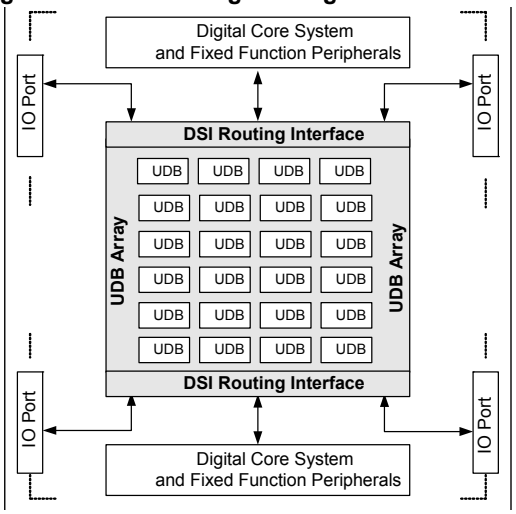
The digital programmable system creates application specific combinations of both standard and advanced digital peripherals and custom logic functions. These peripherals and logic are then interconnected to each other and to any pin on the device, providing a high level of design flexibility and IP security.

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

The main components of the digital programmable system are:

- Universal Digital Blocks (UDB) – These form the core functionality of the digital programmable system. UDBs are a collection of uncommitted logic (PLD) and structural logic (Datapath) optimized to create all common embedded peripherals and customized functionality that are application or design specific.
- Universal Digital Block Array – UDB blocks are arrayed within a matrix of programmable interconnect. The UDB array structure is homogeneous and allows for flexible mapping of digital functions onto the array. The array supports extensive and flexible routing interconnects between UDBs and the Digital System Interconnect.
- Digital System Interconnect (DSI) – Digital signals from Universal Digital Blocks (UDBs), fixed function peripherals, I/O pins, interrupts, DMA, and other system core signals are attached to the Digital System Interconnect to implement full featured device connectivity. The DSI allows any digital function to any pin or other feature routability when used with the Universal Digital Block Array.

**Figure 7-1. CY8C34 Digital Programmable Architecture**



## 7.1 Example Peripherals

The flexibility of the CY8C34 family's Universal Digital Blocks (UDBs) and Analog Blocks allow the user 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, users may also create their own custom components using PSoC Creator. Using PSoC Creator, users 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 CY8C34 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 CY8C34 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
  - $I^2C$
  - UART
  - SPI
- Functions
  - EMIF
  - PWMs
  - Timers
  - Counters
- Logic
  - NOT
  - OR
  - XOR
  - AND

### *7.1.2 Example Analog Components*

The following is a sample of the analog components available in PSoC Creator for the CY8C34 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.

- Amplifiers
  - TIA
  - PGA
  - opamp
- ADC
  - Delta-Sigma
- DACs
  - Current
  - Voltage
  - PWM
- Comparators
- Mixers

### *7.1.3 Example System Function Components*

The following is a sample of the system function components available in PSoC Creator for the CY8C34 family. The exact amount of hardware resources (UDBs, SC/CT blocks, routing, RAM, flash) used by a component varies with the features selected in PSoC Creator for the component.

- CapSense
- LCD Drive
- LCD Control

### *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.

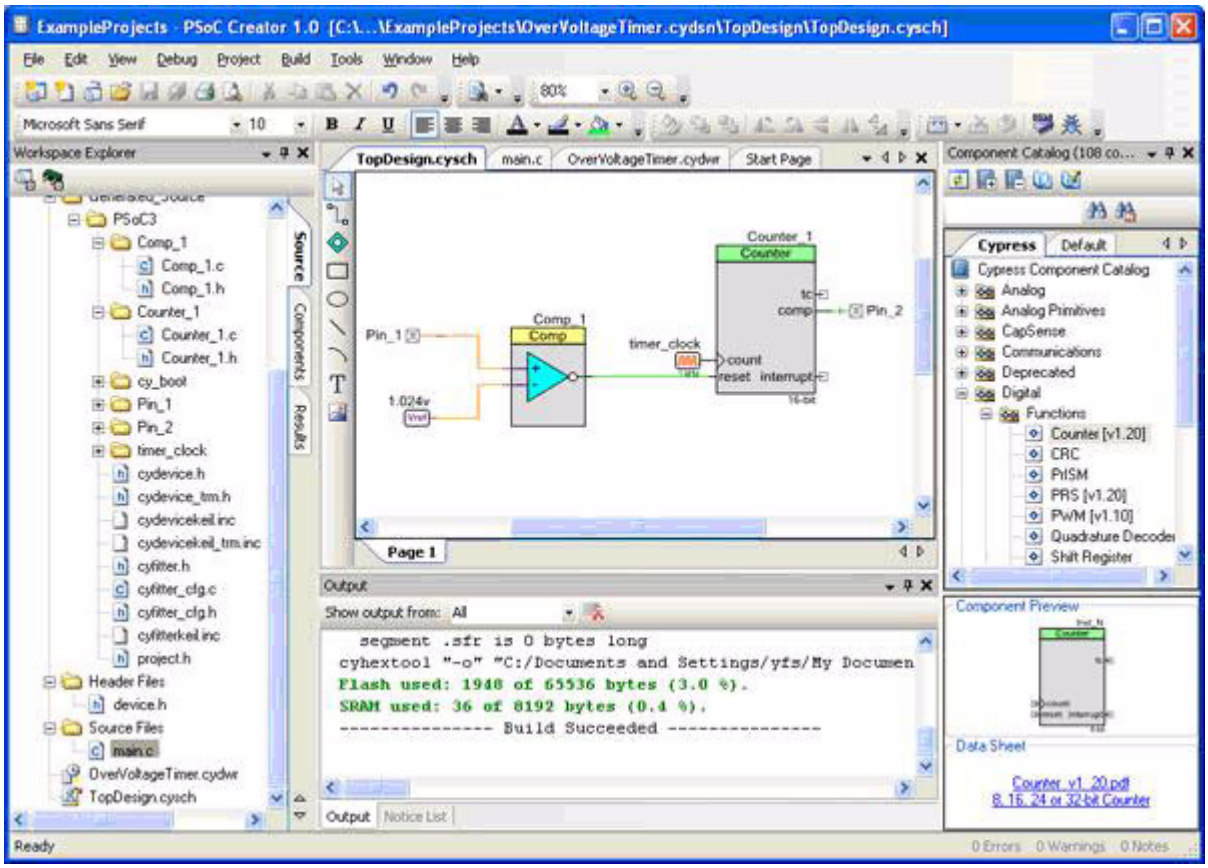
Graphical design entry simplifies the task of configuring a particular part. You can select the required functionality from an extensive catalog of components and place it in your design. All components are parameterized and have an editor dialog that allows you to tailor functionality to your needs.

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.

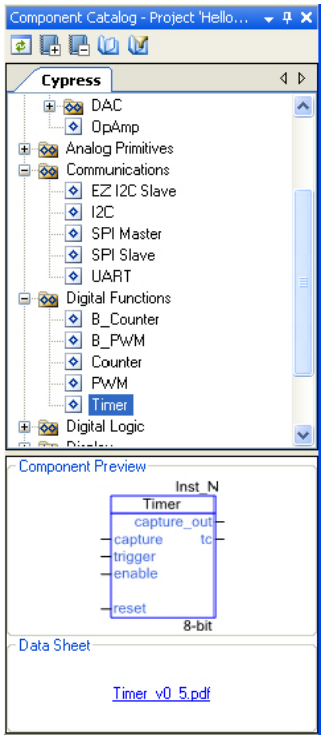
You also have the ability to change the C compiler and evaluate an alternative. Components are designed for portability and are validated against all devices, from all families, and against all supported tool chains. Switching compilers is as easy as editing the from the project options and rebuilding 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**



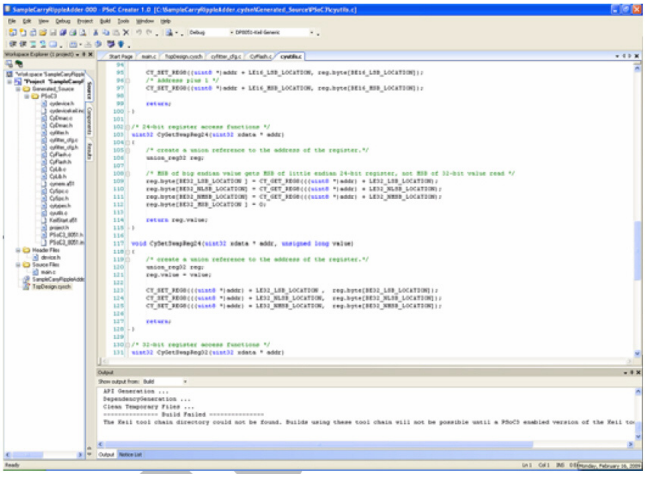

The component catalog is a repository of reusable design elements that select device functionality and customize your PSoC device. It is populated with an impressive selection of content; from simple primitives such as logic gates and device registers, through the digital timers, counters and PWMs, plus analog components such as ADC, DACs, and filters, and communication protocols, such as $I^2C$, USB, and CAN. See Example Peripherals on page 39 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**

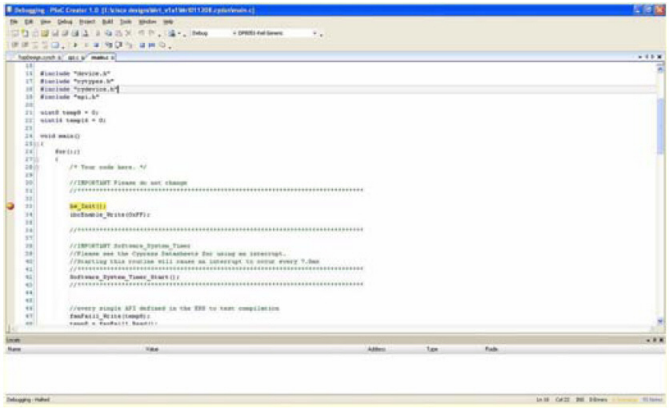


Anchoring the tool is a modern, highly customizable user interface. It includes project management and integrated editors for C and assembler source code, as well the design entry tools.

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

### 7.1.4.5 Nonintrusive Debugging

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



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

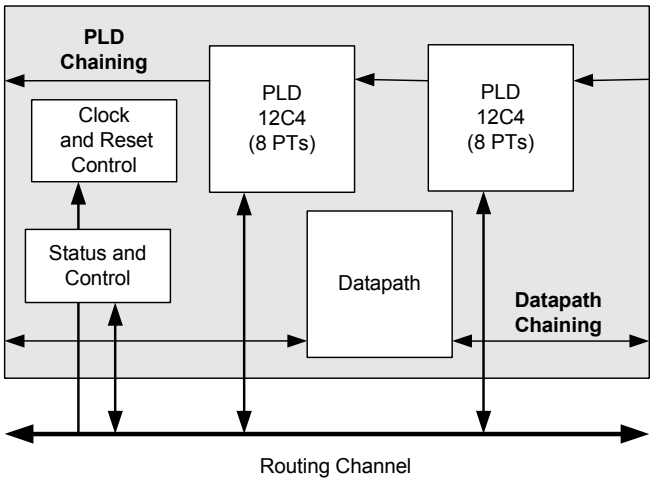
PSoC Creator contains all the tools necessary to complete a design, and then to maintain and extend that design for years to come. All steps of the design flow are carefully integrated and optimized for ease-of-use and to maximize productivity.

## 7.2 Universal Digital Block

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

To achieve this, UDBs consist of a combination of uncommitted logic (PLD), structured logic (Datapath), and a flexible routing scheme to provide interconnect between these elements, I/O connections, and other peripherals. UDB functionality ranges from simple self contained functions that are implemented in one UDB, or even a portion of a UDB (unused resources are available for other functions), to more complex functions that require multiple UDBs. Examples of basic functions are timers, counters, CRC generators, PWMs, dead band generators, and communications functions, such as UARTs, SPI, and $I^2C$. 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**



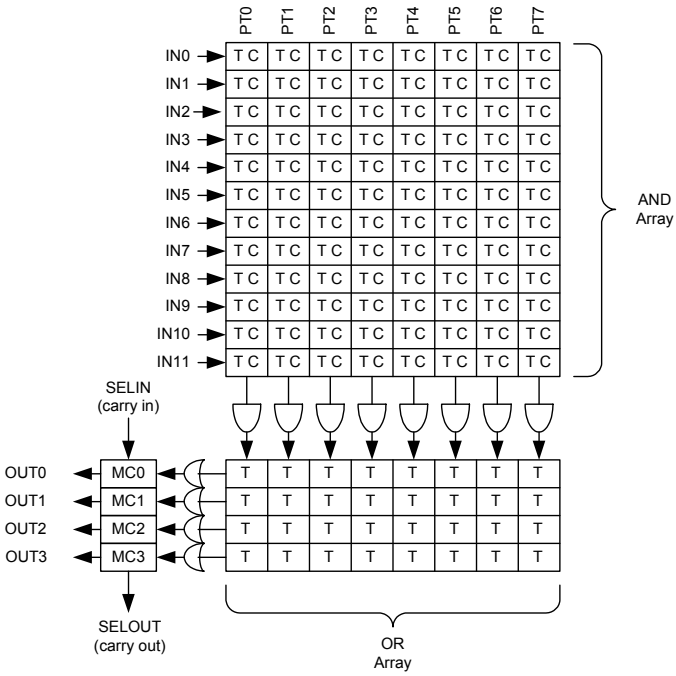

The main component blocks of the UDB are:

- 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, lookup 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**

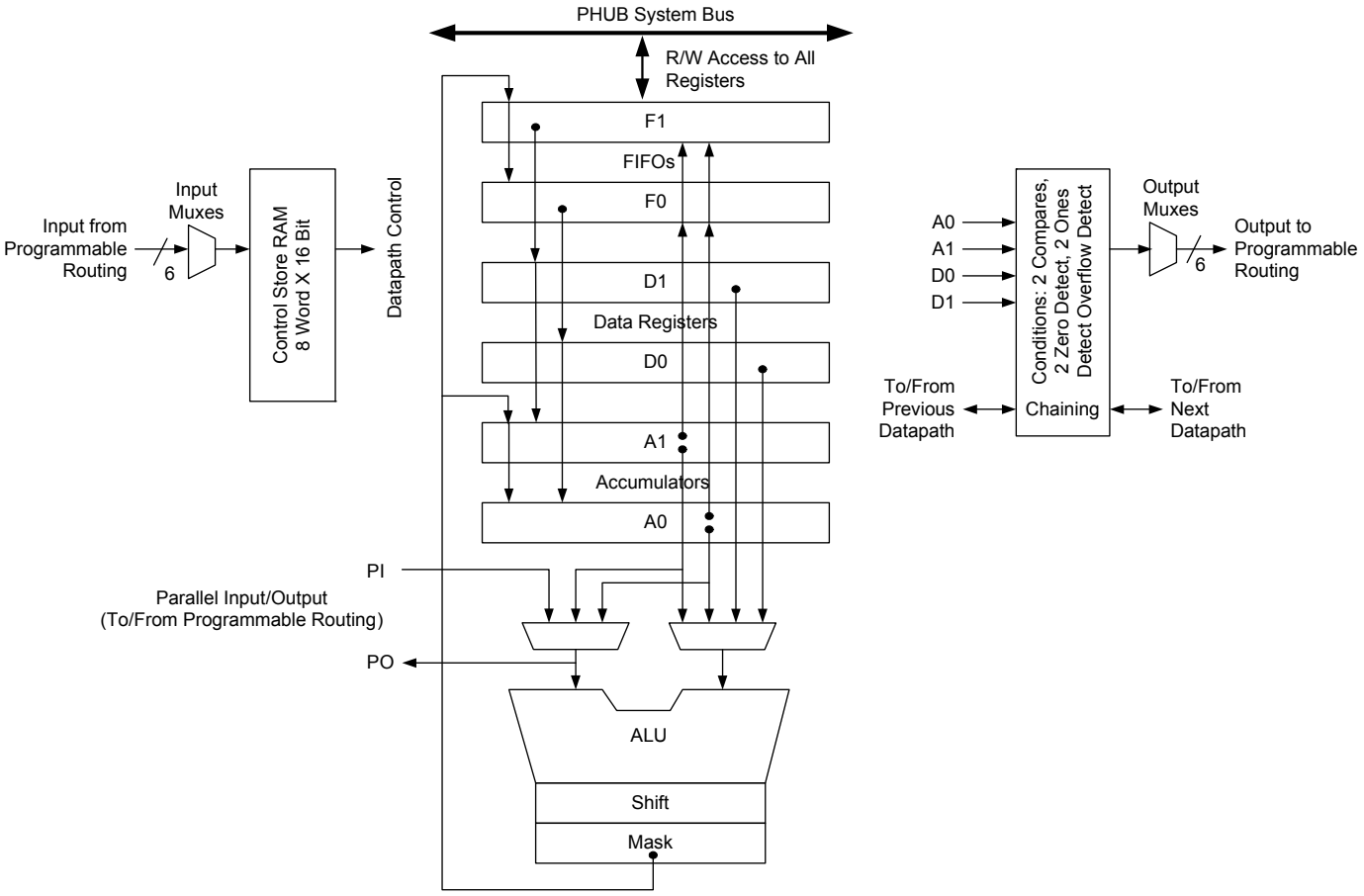



One 12C4 PLD block is shown in Figure 7-7. This PLD has 12 inputs, which feed across eight product terms. Each product term (AND function) can be from 1 to 12 inputs wide, and in a given product term, the true (T) or complement (C) of each input can be selected. The product terms are summed (OR function)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




#### 7.2.2.1 Working Registers

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

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

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

#### 7.2.2.2 Dynamic 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 × 16-bit configuration RAM, which stores eight unique 16-bit wide configurations. The address input to this RAM controls the sequence, and can be routed from any block connected to the UDB routing matrix, most typically PLD logic, I/O pins, or from the outputs of this or other datapath blocks.

**ALU**

The ALU performs eight general purpose functions. They are:

- Increment
- Decrement
- Add
- Subtract
- Logical AND

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

Independent of the ALU operation, these functions are available:

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

#### *7.2.2.3 Conditionals*

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

#### *7.2.2.4 Variable MSB*

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

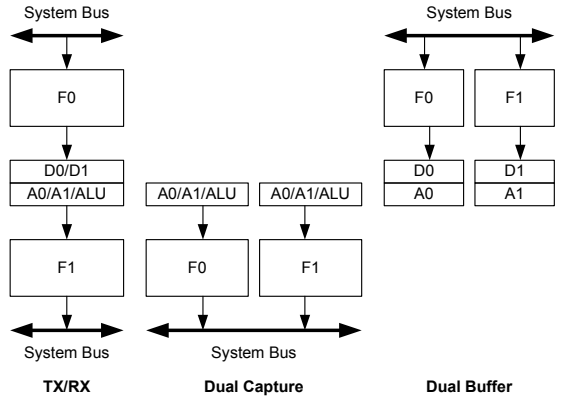
#### *7.2.2.5 Built in CRC/PRS*

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

#### *7.2.2.6 Input/Output FIFOs*

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

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




#### *7.2.2.7 Chaining*

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

#### *7.2.2.8 Time Multiplexing*

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

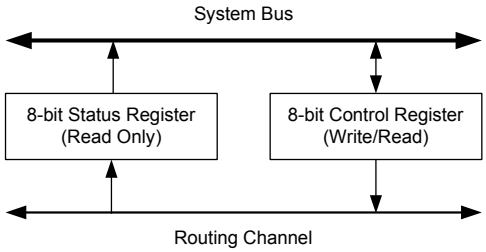
#### *7.2.2.9 Datapath I/O*

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

### *7.2.3 Status and Control Module*

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

**Figure 7-10. 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.

#### *7.2.3.1 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.

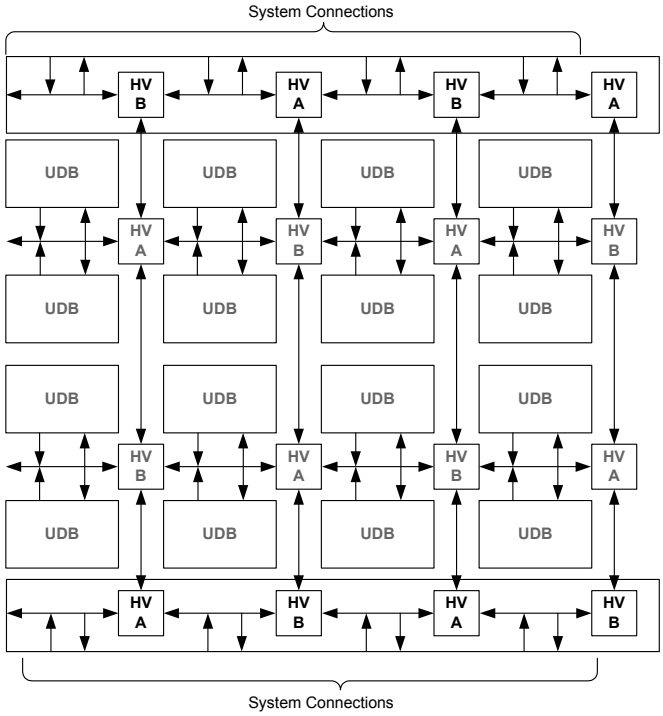
#### *7.2.3.2 Clock Generation*

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**

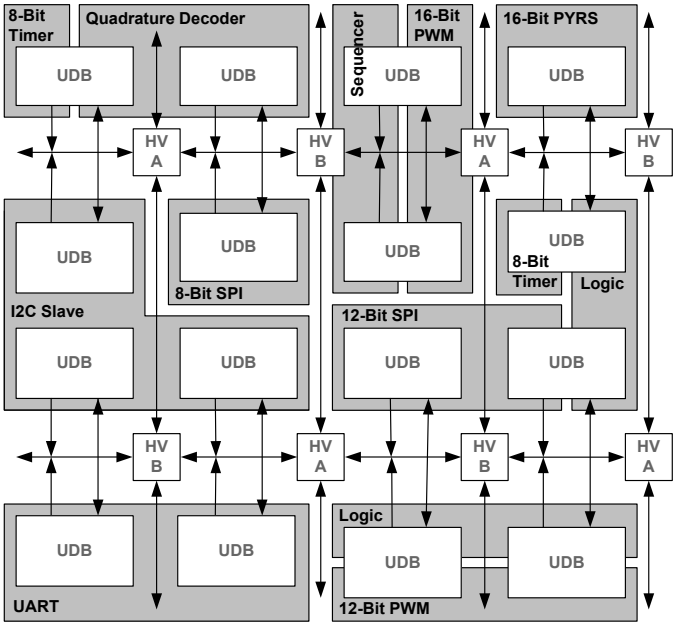
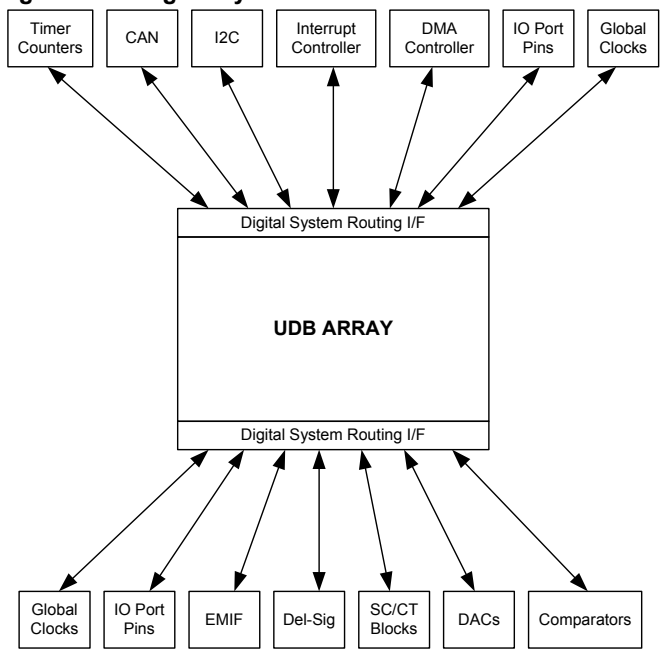



**Figure 7-13. Digital System Interconnect**




## 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.

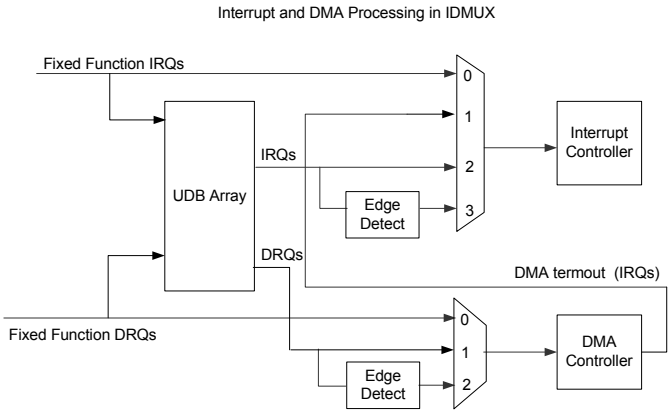
Figure 7-13 illustrates the concept of the digital system interconnect, which connects the UDB array routing matrix with other device peripherals. Any digital core or fixed function peripheral that needs programmable routing is connected to this interface.

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.

Interrupt and DMA routing is very flexible in the CY8C34 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**

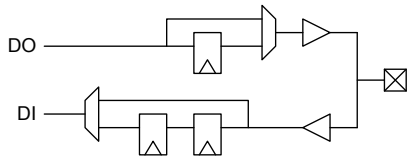
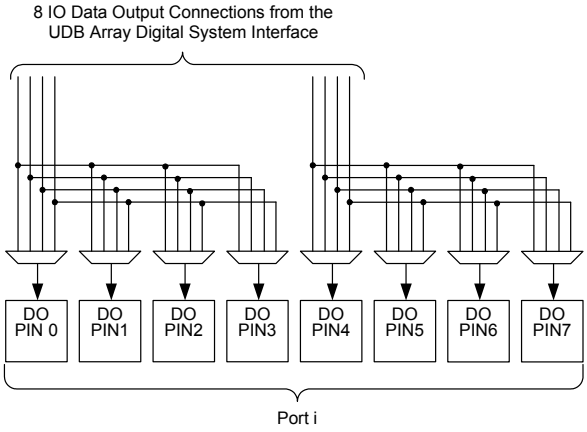


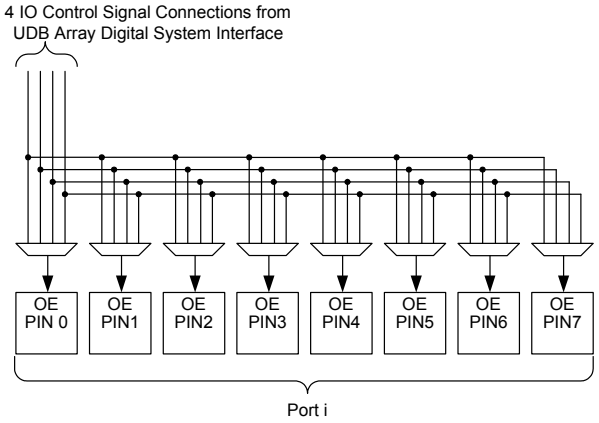

**Figure 7-16. I/O Pin Output Connectivity**




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 tri-state bidirectional pins and buses.

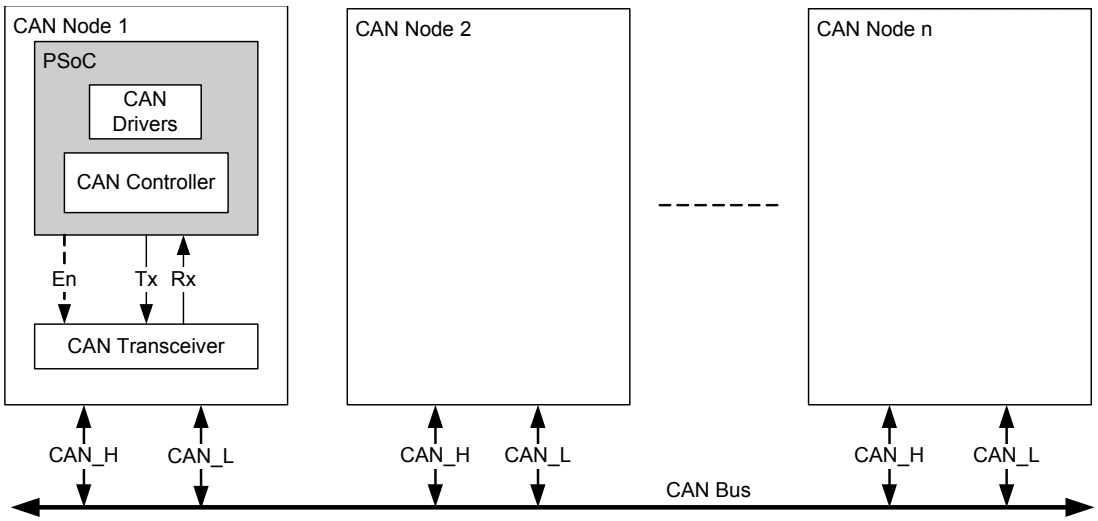
**Figure 7-17. I/O Pin Output Enable Connectivity**




## 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

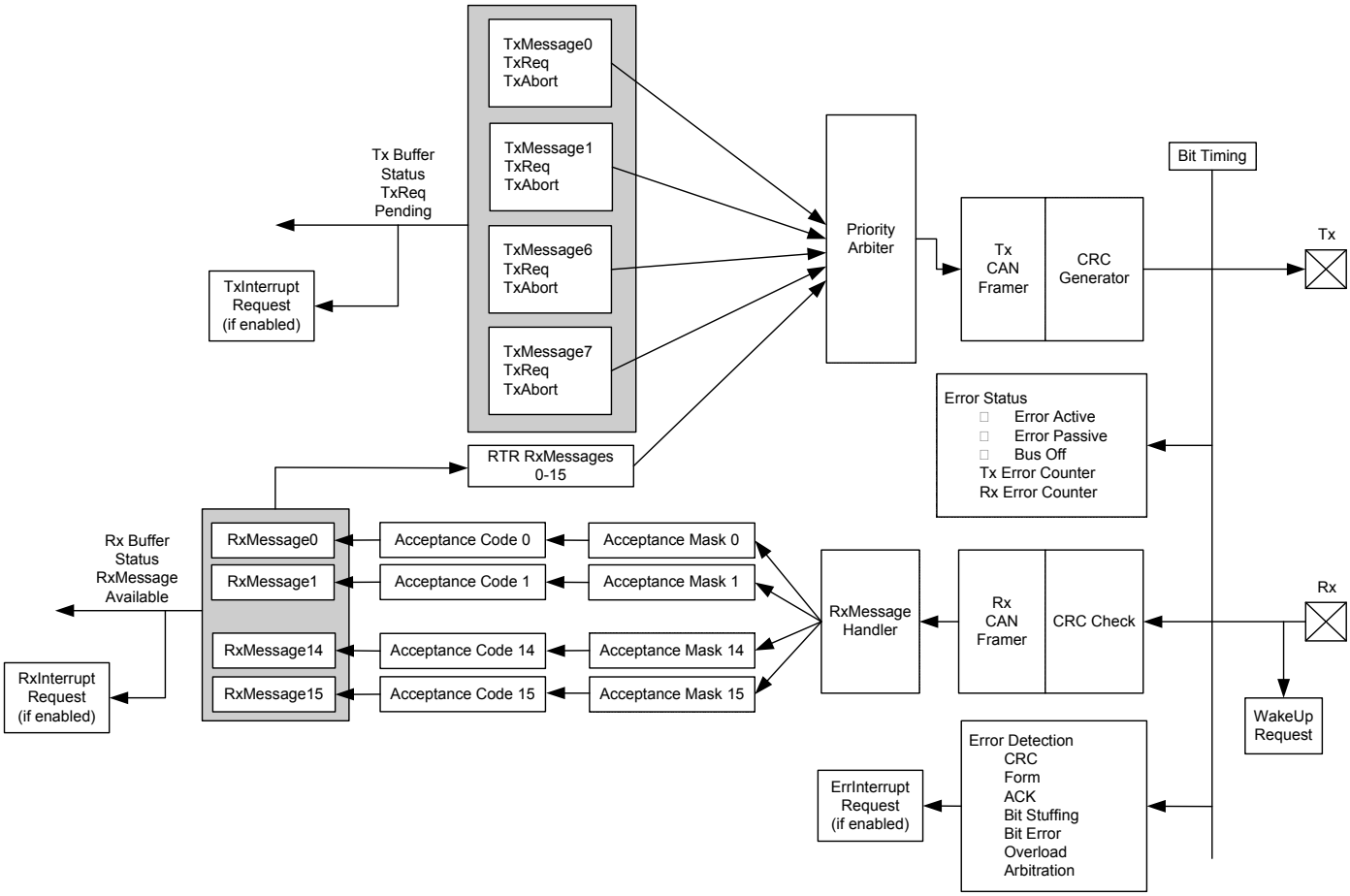
- CAN2.0A/B protocol implementation – ISO 11898 compliant
  - Standard and extended frames with up to 8 bytes of data per frame
  - Message filter capabilities
  - Remote Transmission Request (RTR) support
  - Programmable bit rate up to 1 Mbps
- Listen Only mode
- SW readable error counter and indicator
- 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**



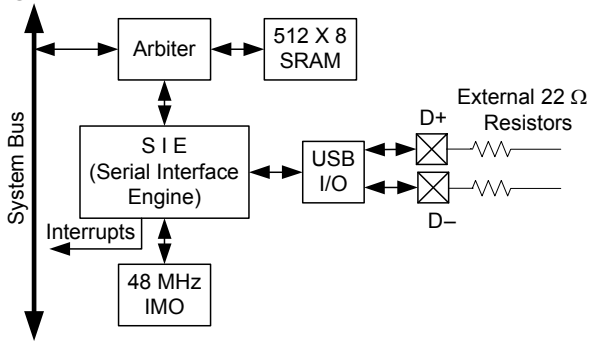
## 7.6 USB

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 32.

USB includes the following features:

- Eight unidirectional data endpoints
- One bidirectional control endpoint 0 (EP0)
- Shared 512-byte buffer for the eight data endpoints
- Dedicated 8-byte buffer for EP0
- Three memory modes
  - Manual Memory Management with No DMA Access
  - Manual Memory Management with Manual DMA Access
  - Automatic Memory Management with Automatic DMA Access
- Internal 3.3-V regulator for transceiver
- Internal 48 MHz main oscillator mode 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

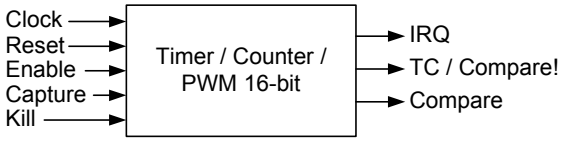
The Timer/Counter/PWM peripheral is a 16-bit dedicated peripheral providing three of the most common embedded peripheral features. As almost all embedded systems use some combination of timers, counters, and PWMs. Four of them have been included on this PSoC device family. Additional and more advanced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

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

Timer/Counter/PWM features include:

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

**Figure 7-21. Timer/Counter/PWM**




## 7.8 I²C

The I²C peripheral provides a synchronous two wire interface designed to interface the PSoC device with a two wire I²C serial communication bus. The bus is compliant with Philips 'The I²C Specification' version 2.1. Additional I²C interfaces can be instantiated using Universal Digital Blocks (UDBs) in PSoC Creator, as required.

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

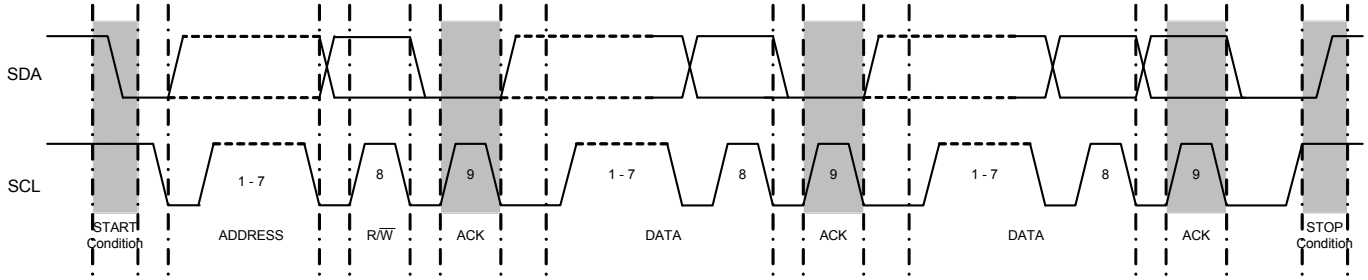
I²C provides hardware address detect of a 7-bit address without CPU intervention. Additionally the device can wake from low-power modes on a 7-bit hardware address match. If wakeup functionality is required, I²C pin connections are limited to the two special sets of SIO pins.

I²C features include:

- Slave and Master, Transmitter, and Receiver operation
- Byte processing for low CPU overhead
- Interrupt or polling CPU interface
- Support for bus speeds up to 1 Mbps (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)
- 7-bit hardware address compare
- Wake from low-power modes on address match

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. I²C Complete Transfer Timing**

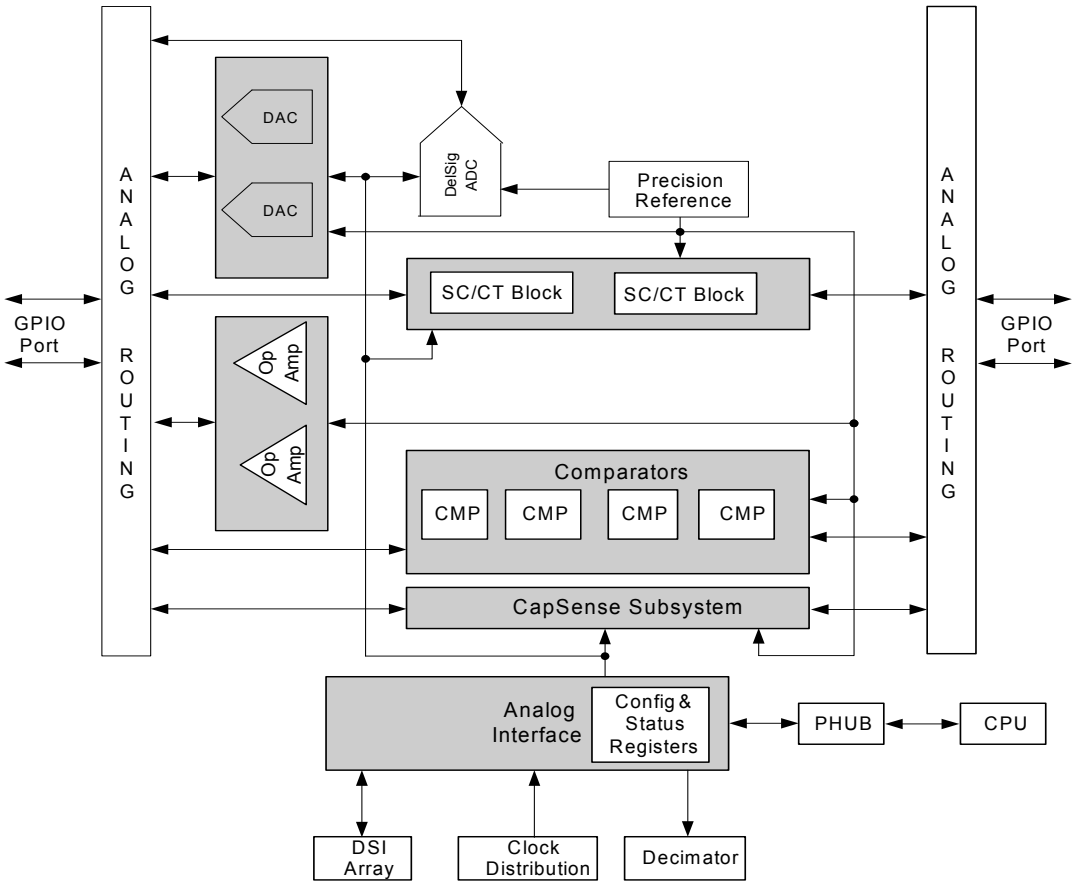



## 8. Analog Subsystem

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.
- High resolution Delta-Sigma ADC.
- Two 8-bit DACs that provide either voltage or current output.
- Four comparators with optional connection to configurable LUT outputs.
- Two configurable switched capacitor/continuous time (SC/CT) blocks for functions that include opamp, unity gain buffer, programmable gain amplifier, transimpedance amplifier, and mixer.
- Two opamps for internal use and connection to GPIO that can be used as high current output buffers.
- 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 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 (PGA, transimpedance amplifier, voltage DAC, current DAC, and so on). 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 CY8C34 family of devices has a flexible analog routing architecture that provides the capability to connect GPIOs and different analog blocks, and also route signals between different analog blocks. One of the strong points of this flexible routing architecture is that it allows dynamic routing of input and output connections to the different analog blocks.

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

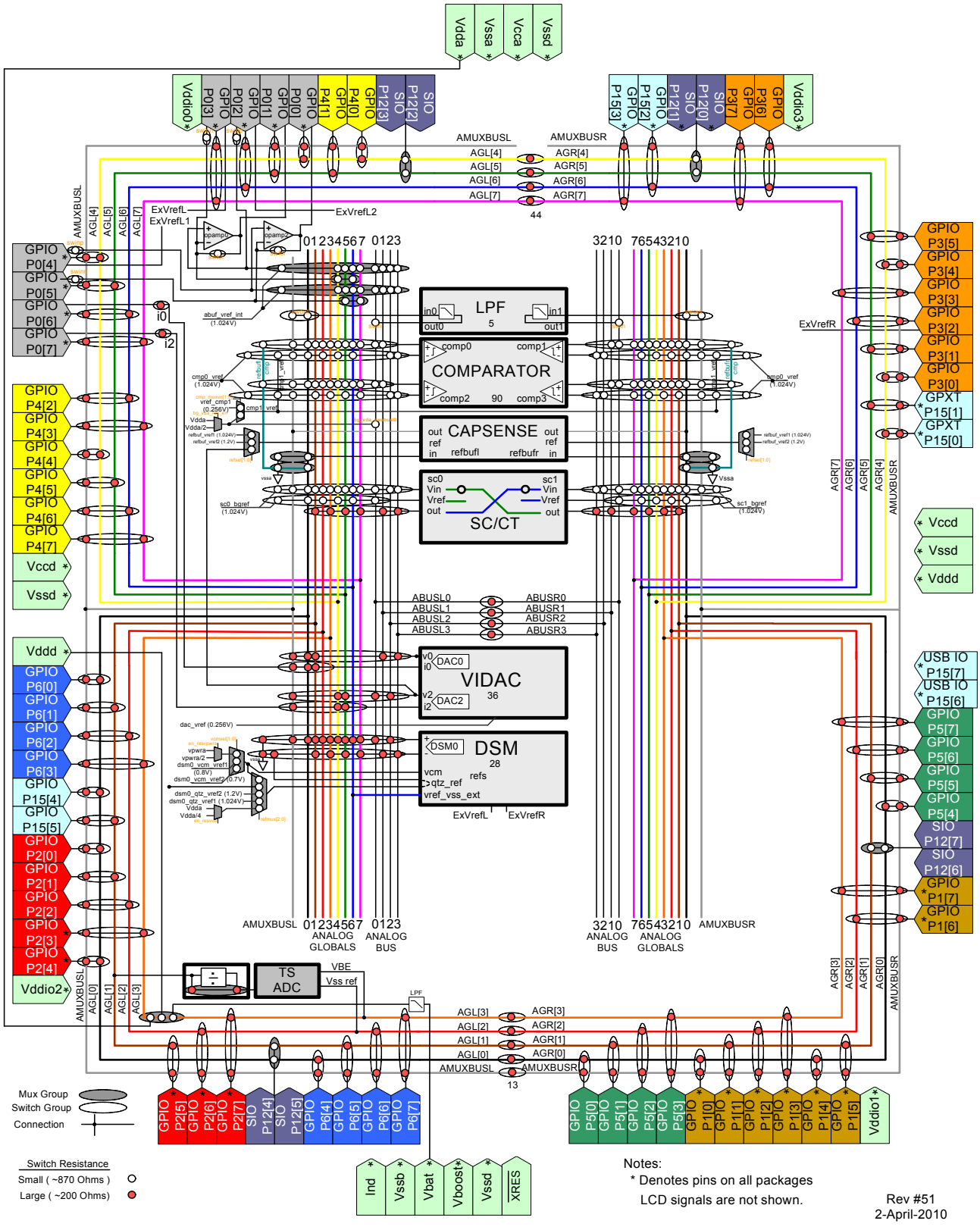
### *8.1.1 Features*

- Flexible, configurable analog routing architecture
- 16 Analog globals (AG) and two analog mux buses (AMUXBUS) to connect GPIOs and the analog blocks
- Each GPIO is connected to one analog global and one analog mux bus
- 8 Analog local buses (abus) to route signals between the different analog blocks
- Multiplexers and switches for input and output selection of the analog blocks

### *8.1.2 Functional Description*

Analog globals (AGs) and analog mux buses (AMUXBUS) provide analog connectivity between GPIOs and the various analog blocks. There are 16 AGs in the CY8C34 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 CY8C34, one in the left half (AMUXBUSL) and one in the right half (AMUXBUSR), as shown in Figure 8-2.

**Figure 8-2. CY8C34 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 CY8C34, 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.

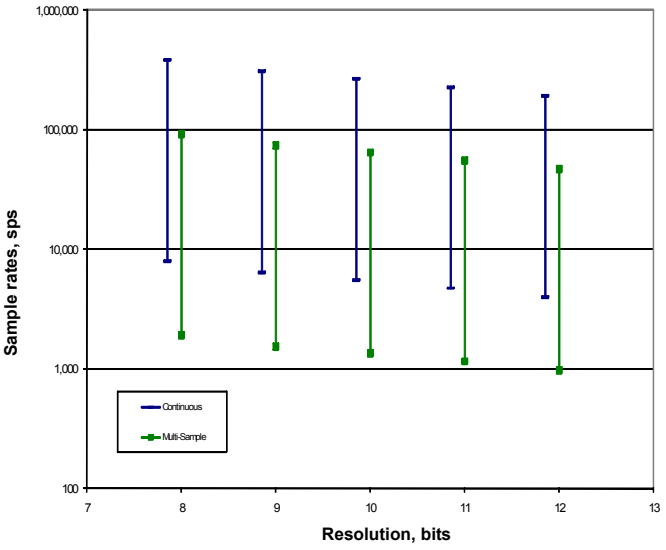
## 8.2 Delta-sigma ADC

The CY8C34 device contains one delta-sigma ADC. This ADC offers differential input, high resolution and excellent linearity, making it a good ADC choice for measurement applications. The converter can be configured to output 12-bit resolution at data rates of up to 192 ksps. At a fixed clock rate, resolution can be traded for faster data rates as shown in Table 8-1 and Figure 8-3.

**Table 8-1. Delta-sigma ADC Performance**

| Bits | Maximum Sample Rate (sps) | SINAD (dB) |
|---|---|---|
| 12 | 192 k | 66 |
| 8 | 384 k | 43 |

**Figure 8-3. Delta-sigma ADC Sample Rates, Range = ±1.024 V**




### 8.2.1 Functional Description

The ADC connects and configures three basic components, input buffer, delta-sigma modulator, and decimator. The basic block diagram is shown in Figure 8-4. The signal from the input muxes is delivered to the delta-sigma modulator either directly or through the input buffer. The delta-sigma modulator performs the actual analog to digital conversion. The modulator over-samples the input and generates a serial data stream output. This high speed data stream is not useful for most applications without some type of post processing, and so is passed to the decimator through the Analog Interface block. The decimator converts the high speed serial data stream into parallel ADC results. The modulator/decimator frequency response is $[(\sin x)/x]^4$; a typical frequency response is shown in Figure 8-5.

**Figure 8-4. Delta-sigma ADC Block Diagram**

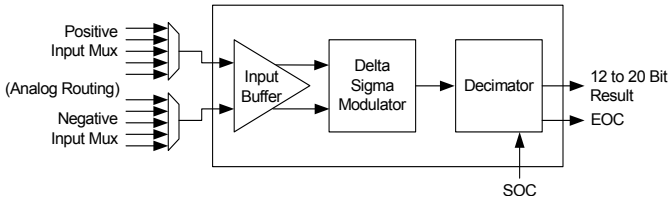
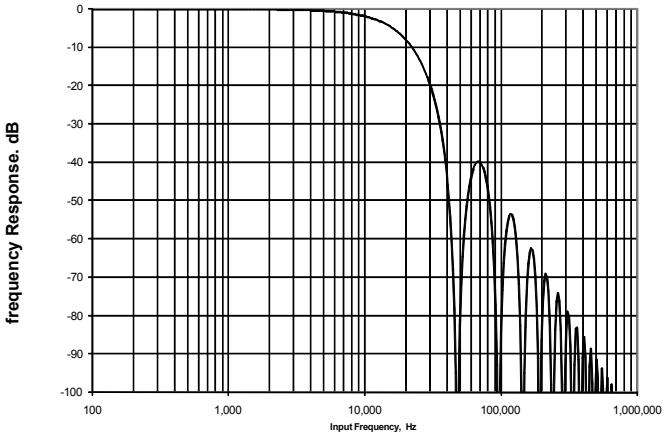



**Figure 8-5. Delta-sigma ADC Frequency Response, Normalized to Output, Sample Rate = 48 kHz**




Resolution and sample rate are controlled by the Decimator. Data is pipelined in the decimator; the output is a function of the last four samples. When the input multiplexer is switched, the output data is not valid until after the fourth sample after the switch.

### 8.2.2 Operational Modes

The ADC can be configured by the user to operate in one of four modes: Single Sample, Multi Sample, Continuous, or Multi Sample (Turbo). All four modes are started by either a write to the start bit in a control register or an assertion of the Start of Conversion (SoC) signal. When the conversion is complete, a status bit is set and the output signal End of Conversion (EoC) asserts high and remains high until the value is read by either the DMA controller or the CPU.

#### 8.2.2.1 Single Sample

In Single Sample mode, the ADC performs one sample conversion on a trigger. In this mode, the ADC stays in standby state waiting for the SoC signal to be asserted. When SoC is signaled the ADC performs four successive conversions. The first three conversions prime the decimator. The ADC result is valid and available after the fourth conversion, at which time the

EoC signal is generated. To detect the end of conversion, the system may poll a control register for status or configure the external EoC signal to generate an interrupt or invoke a DMA request. When the transfer is done the ADC reenters the standby state where it stays until another SoC event.

#### 8.2.2.2 Continuous

Continuous sample mode is used to take multiple successive samples of a single input signal. Multiplexing multiple inputs should not be done with this mode. There is a latency of three conversion times before the first conversion result is available. This is the time required to prime the decimator. After the first result, successive conversions are available at the selected sample rate.

#### 8.2.2.3 Multi Sample

Multi sample mode is similar to continuous mode except that the ADC is reset between samples. This mode is useful when the input is switched between multiple signals. The decimator is re-primed between each sample so that previous samples do not affect the current conversion. Upon completion of a sample, the next sample is automatically initiated. The results can be transferred using either firmware polling, interrupt, or DMA.

More information on output formats is provided in the Technical Reference Manual.

### 8.2.3 Start of Conversion Input

The SoC signal is used to start an ADC conversion. A digital clock or UDB output can be used to drive this input. It can be used when the sampling period must be longer than the ADC conversion time or when the ADC must be synchronized to other hardware. This signal is optional and does not need to be connected if ADC is running in a continuous mode.

### 8.2.4 End of Conversion Output

The EoC signal goes high at the end of each ADC conversion. This signal may be used to trigger either an interrupt or DMA request.

## 8.3 Comparators

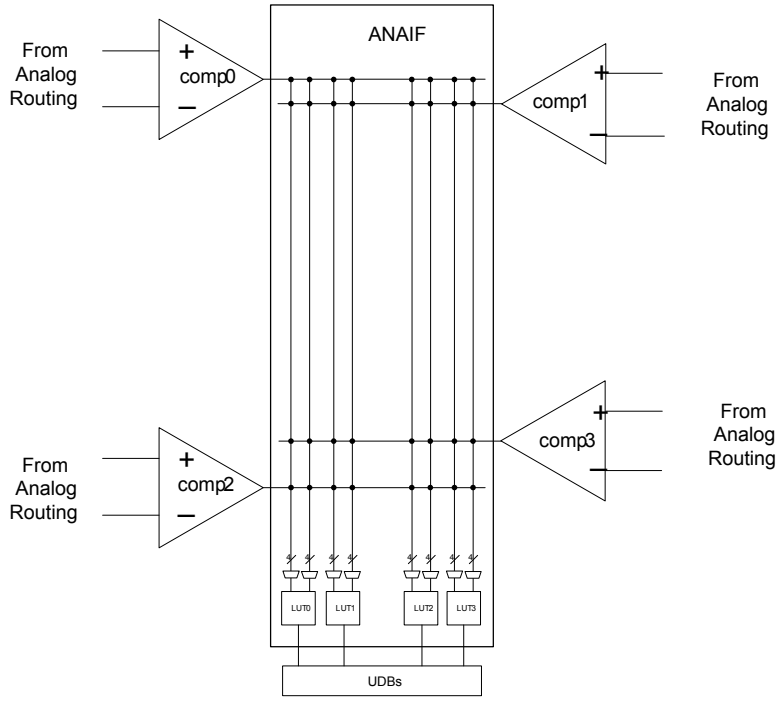
The CY8C34 family of devices contains four comparators in a device. Comparators have these features:

- Input offset factory trimmed to less than 5 mV
- Rail-to-rail common mode input range ($V_{SSA}$ to $V_{DDA}$)
- Speed and power can be traded off by using one of three modes: fast, slow, or ultra low-power
- Comparator outputs can be routed to lookup 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 connections to GPIO, DAC outputs and SC block outputs

### 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 8-6. Analog Comparator**



### 8.3.2 LUT

The CY8C34 family of devices contains four LUTs. The LUT is a two input, one output lookup table that is driven by any 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 8-2.

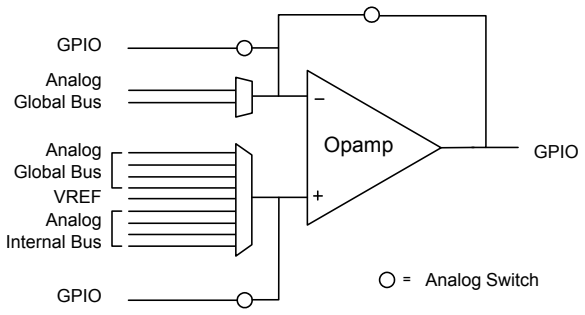
**Table 8-2. LUT Function vs. Program Word and Inputs**

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

## 8.4 Opamps

The CY8C34 family of devices contains two general purpose opamps in a device.

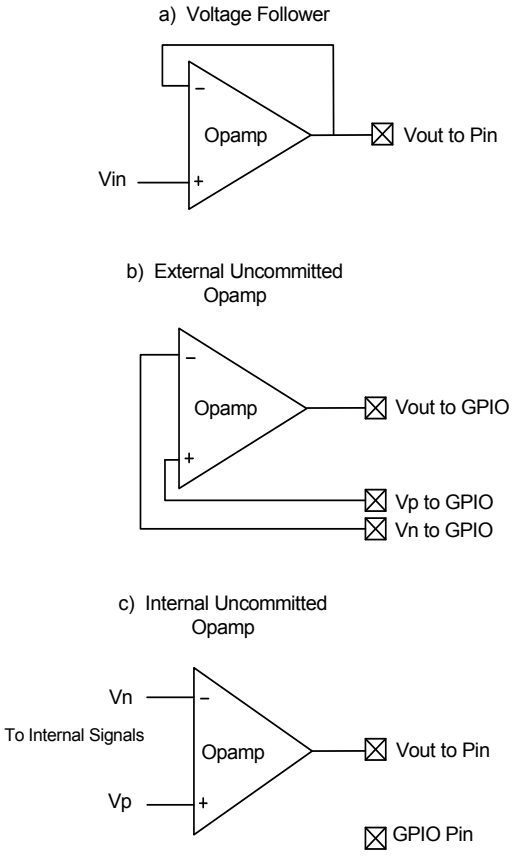
**Figure 8-7. Opamp**




The opamp is uncommitted and can be configured as a gain stage or voltage follower, or output buffer on external or internal signals.

See Figure 8-8. In any configuration, the input and output signals can all be connected to the internal global signals and monitored with an ADC, or comparator. The configurations are implemented with switches between the signals and GPIO pins.

**Figure 8-8. Opamp Configurations**




The opamp has three speed modes, slow, medium, and fast. The slow mode consumes the least amount of quiescent power and the fast mode consumes the most power. The inputs are able to swing rail-to-rail. The output swing is capable of rail-to-rail operation at low current output, within 50 mV of the rails. When driving high current loads (about 25 mA) the output voltage may only get within 500 mV of the rails.

## 8.5 Programmable SC/CT Blocks

The CY8C34 family of devices contains two switched capacitor/continuous time (SC/CT) blocks in a device. Each switched capacitor/continuous time block is built around a single rail-to-rail high bandwidth opamp.

Switched capacitor is a circuit design technique that uses capacitors plus switches instead of resistors to create analog functions. These circuits work by moving charge between capacitors by opening and closing different switches. Nonoverlapping in phase clock signals control the switches, so that not all switches are ON simultaneously.

The PSoC Creator tool offers a user friendly interface, which allows you to easily program the SC/CT blocks. Switch control and clock phase control configuration is done by PSoC Creator so users only need to determine the application use parameters such as gain, amplifier polarity, $V_{REF}$ connection, and so on.

The same opamps and block interfaces are also connectable to an array of resistors which allows the construction of a variety of continuous time functions.

The opamp and resistor array is programmable to perform various analog functions including

- Naked operational amplifier – Continuous mode
- Unity-gain buffer – Continuous mode
- Programmable gain amplifier (PGA) – Continuous mode
- Transimpedance amplifier (TIA) – Continuous mode
- Up/down mixer – Continuous mode
- Sample and hold mixer (NRZ S/H) – Switched cap mode
- First order analog to digital modulator – Switched cap mode

### *8.5.1 Naked Opamp*

The Naked Opamp presents both inputs and the output for connection to internal or external signals. The opamp has a unity gain bandwidth greater than 6.0 MHz and output drive current up to 650 µA. This is sufficient for buffering internal signals (such as DAC outputs) and driving external loads greater than 7.5 kohms.

### *8.5.2 Unity Gain*

The Unity Gain buffer is a Naked Opamp with the output directly connected to the inverting input for a gain of 1.00. It has a –3 dB bandwidth greater than 6.0 MHz.

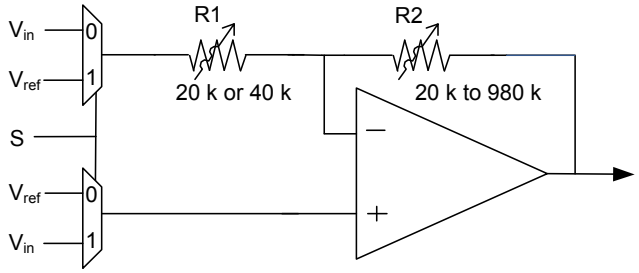
### *8.5.3 PGA*

The PGA amplifies an external or internal signal. The PGA can be configured to operate in inverting mode or noninverting mode. The PGA function may be configured for both positive and negative gains as high as 50 and 49 respectively. The gain is adjusted by changing the values of R1 and R2 as illustrated in Figure 8-9. The schematic in Figure 8-9 shows the configuration and possible resistor settings for the PGA. The gain is switched from inverting and non inverting by changing the shared select value of the both the input muxes. The bandwidth for each gain case is listed in Table 8-3.

**Table 8-3. Bandwidth**

| Gain | Bandwidth |
|---|---|
| 1 | 6.0 MHz |
| 24 | 340 kHz |
| 48 | 220 kHz |
| 50 | 215 kHz |

**Figure 8-9. PGA Resistor Settings**




The PGA is used in applications where the input signal may not be large enough to achieve the desired resolution in the ADC, or dynamic range of another SC/CT block such as a mixer. The gain is adjustable at runtime, including changing the gain of the PGA prior to each ADC sample.

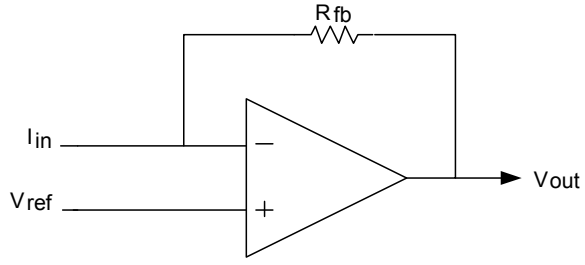
### *8.5.4 TIA*

The Transimpedance Amplifier (TIA) converts an internal or external current to an output voltage. The TIA uses an internal feedback resistor in a continuous time configuration to convert input current to output voltage. For an input current Iin, the output voltage is Iin x Rfb +$V_{REF}$, where $V_{REF}$ is the value placed on the non inverting input. The feedback resistor Rfb is programmable between 20 KΩ and 1 MΩ through a configuration register. Table 8-4 shows the possible values of Rfb and associated configuration settings.

**Table 8-4. Feedback Resistor Settings**

| Configuration Word | Nominal $R_{fb}$ (KΩ) |
|---|---|
| 000b | 20 |
| 001b | 30 |
| 010b | 40 |
| 011b | 60 |
| 100b | 120 |
| 101b | 250 |
| 110b | 500 |
| 111b | 1000 |

**Figure 8-10. Continuous Time TIA Schematic**



The TIA configuration is used for applications where an external sensor's output is current as a function of some type of stimulus such as temperature, light, magnetic flux etc. In a common application, the voltage DAC output can be connected to the $V_{REF}$ TIA input to allow calibration of the external sensor bias current by adjusting the voltage DAC output voltage.

## 8.6 LCD Direct Drive

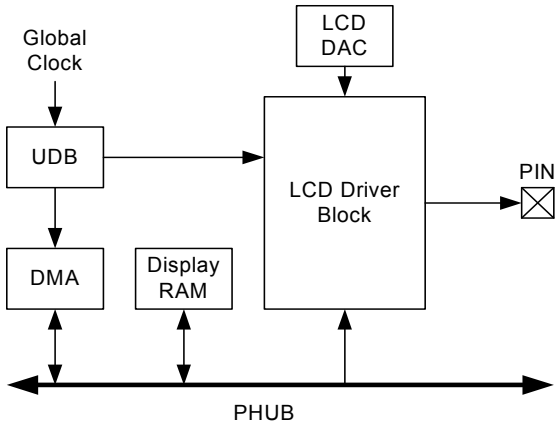
The PSoC Liquid Crystal Display (LCD) driver system is a highly configurable peripheral designed to allow PSoC to directly drive a broad range of LCD glass. All voltages are generated on chip, eliminating the need for external components. With a high multiplex ratio of up to 1/16, the CY8C34 family LCD driver system can drive a maximum of 736 segments. The PSoC LCD driver module was also designed with the conservative power budget of portable devices in mind, enabling different LCD drive modes and power down modes to conserve power.

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

Key features of the PSoC LCD segment system are:

- LCD panel direct driving
- Type A (standard) and Type B (low-power) waveform support
- Wide operating voltage range support (2 V to 5 V) for LCD panels
- Static, 1/2, 1/3, 1/4, 1/5 bias voltage levels
- Internal bias voltage generation through internal resistor ladder
- Up to 62 total common and segment outputs
- Up to 1/16 multiplex for a maximum of 16 backplane/common outputs
- Up to 62 front plane/segment outputs for direct drive
- Drives up to 736 total segments (16 backplane × 46 front plane)
- Up to 64 levels of software controlled contrast
- Ability to move display data from memory buffer to LCD driver through DMA (without CPU intervention)
- Adjustable LCD refresh rate from 10 Hz to 150 Hz
- Ability to invert LCD display for negative image
- Three LCD driver drive modes, allowing power optimization

**Figure 8-11. LCD System**




### *8.6.1 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.

### *8.6.2 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.

### *8.6.3 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.

### *8.6.4 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.

## 8.7 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.

## 8.8 Temp Sensor

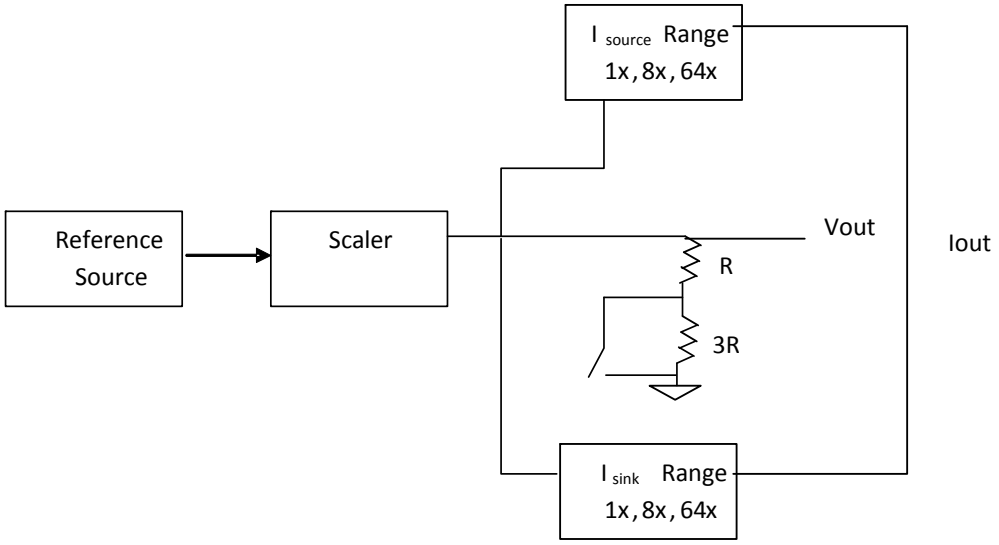
Die temperature is used to establish programming parameters for writing flash. Die temperature is measured using a dedicated sensor based on a forward biased transistor. The temperature sensor has its own auxiliary ADC.

## 8.9 DAC

The CY8C34 parts contain two Digital to Analog Convertors (DACs). Each DAC is 8-bit and can be configured for either voltage or current output. The DACs support CapSense, power supply regulation, and waveform generation. Each DAC has the following features:

- Adjustable voltage or current output in 255 steps
- Programmable step size (range selection)
- Eight bits of calibration to correct ± 25 percent 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 8-12. DAC Block Diagram**




### *8.9.1 Current DAC*

The current DAC (IDAC) can be configured for the ranges 0 to 32 µA, 0 to 256 µA, and 0 to 2.048 mA. The IDAC can be configured to source or sink current.

### *8.9.2 Voltage DAC*

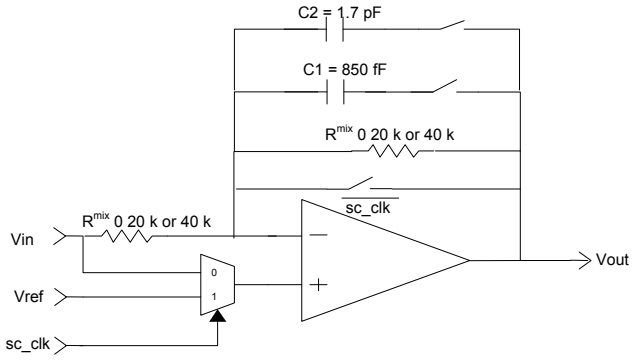
For the voltage DAC (VDAC), the current DAC output is routed through resistors. The two ranges available for the VDAC are 0 to 1.024 V and 0 to 4.096 V. In voltage mode any load connected to the output of a DAC should be purely capacitive (the output of the VDAC is not buffered).

## 8.10 Up/Down Mixer

In continuous time mode, the SC/CT block components are used to build an up or down mixer. Any mixing application contains an input signal frequency and a local oscillator frequency. The polarity of the clock, Fclk, switches the amplifier between inverting or noninverting gain. The output is the product of the input and the switching function from the local oscillator, with frequency components at the local oscillator plus and minus the signal frequency (Fclk + Fin and Fclk – Fin) and reduced-level frequency components at odd integer multiples of the local oscillator frequency. The local oscillator frequency is provided by the selected clock source for the mixer.

Continuous time up and down mixing works for applications with input signals and local oscillator frequencies up to 1 MHz.

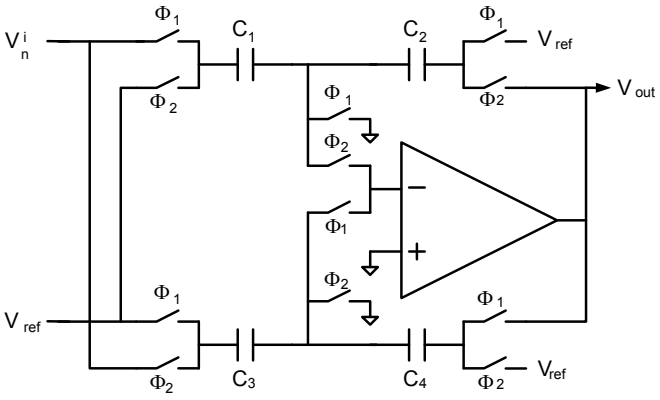
**Figure 8-13. Mixer Configuration**




## 8.11 Sample and Hold

The main application for a sample and hold, is to hold a value stable while an ADC is performing a conversion. Some applications require multiple signals to be sampled simultaneously, such as for power calculations (V and I).

**Figure 8-14. Sample and Hold Topology (Φ1 and Φ2 are opposite phases of a clock)**




### *8.11.1 Down Mixer*

The SC/CT block can be used as a mixer to down convert an input signal. This circuit is a high bandwidth passive sample network that can sample input signals up to 14 MHz. This sampled value is then held using the opamp with a maximum clock rate of 4 MHz. The output frequency is at the difference between the input frequency and the highest integer multiple of the Local Oscillator that is less than the input.

### *8.11.2 First Order Modulator – SC Mode*

A first order modulator is constructed by placing the SC/CT block in an integrator mode and using a comparator to provide a 1-bit feedback to the input. Depending on this bit, a reference voltage is either subtracted or added to the input signal. The block output is the output of the comparator and not the integrator in the modulator case. The signal is downshifted and buffered and then processed by a decimator to make a delta-sigma converter or a counter to make an incremental converter. The accuracy of the sampled data from the first-order modulator is determined from several factors.

The main application for this modulator is for a low-frequency ADC with high accuracy. Applications include strain gauges, thermocouples, precision voltage, and current measurement.

# 9. Programming, Debug Interfaces, Resources

PSoC devices include extensive support for programming, testing, debugging, and tracing both hardware and firmware. Three interfaces are available: JTAG, SWD, and SWV. JTAG and SWD support all programming and debug features of the device. JTAG also supports standard JTAG scan chains for board level test and chaining multiple JTAG devices to a single JTAG connection.

For more information on PSoC 3 Programming, refer to the application note AN62391 - In-System Programming for PSoC® 3.

Complete Debug on Chip (DoC) functionality enables full device debugging in the final system using the standard production device. It does not require special interfaces, debugging pods, simulators, or emulators. Only the standard programming connections are required to fully support debug.

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

All DOC circuits are disabled by default and can only be enabled in firmware. If not enabled, the only way to reenable them is to erase the entire device, clear flash protection, and reprogram the device with new firmware that enables DOC. Disabling DOC 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 you cannot access the device later. Because all programming, debug, and test interfaces are disabled when Device Security is enabled, PSoCs with Device Security enabled may not be returned for failure analysis.

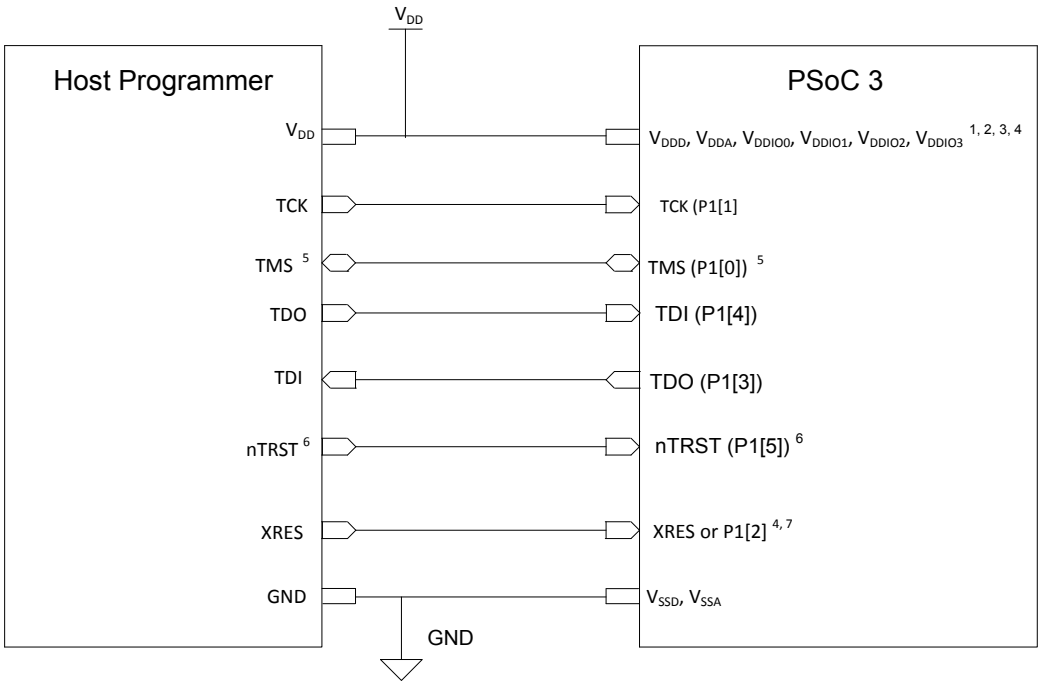
**Table 9-1. Debug Configurations**

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

## 9.1 JTAG Interface

The IEEE 1149.1 compliant JTAG interface exists on four or five pins (the nTRST pin is optional). The JTAG clock frequency can be up to 8 MHz, or 1/3 of the CPU clock frequency for 8 and 16-bit transfers, or 1/5 of the CPU clock frequency for 32-bit transfers, whichever is least. By default, the JTAG pins are enabled on new devices but the JTAG interface can be disabled, allowing these pins to be used as General Purpose I/O (GPIO) instead. The JTAG interface is used for programming the flash memory, debugging, I/O scan chains, and JTAG device chaining.

**Figure 9-1. JTAG Interface Connections between PSoC 3 and Programmer**




[1] The voltage levels of Host Programmer and the PSoC 3 voltage domains involved in Programming should be same. The Port 1 JTAG pins, XRES pin (XRES_N or P1[2]) are powered by $V_{DDIO1}$. So, $V_{DDIO1}$ of PSoC 3 should be at same voltage level as host $V_{DD}$. Rest of PSoC 3 voltage domains ( $V_{DDD}$, $V_{DDA}$, $V_{DDIO0}$, $V_{DDIO2}$, $V_{DDIO3}$) need not be at the same voltage level as host Programmer.

[2] Vdda must be greater than or equal to all other power supplies (Vddd, Vddio's) in PSoC 3.

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

[4] For JTAG Programming, Device reset can also be done without connecting to the XRES pin or Power cycle mode by using the TMS,TCK,TDI, TDO pins of PSoC 3, and writing to a specific register. But this requires that the DPS setting in NVL is not equal to "Debug Ports Disabled".

[5] By default, PSoC 3 is configured for 4-wire JTAG mode unless user changes the DPS setting. So the TMS pin is unidirectional. But if the DPS setting is changed to non-JTAG mode, the TMS pin in JTAG is bi-directional as the SWD Protocol has to be used for acquiring the PSoC 3 device initially. After switching from SWD to JTAG mode, the TMS pin will be uni-directional. In such a case, unidirectional buffer should not be used on TMS line.

[6] nTRST JTAG pin (P1[5]) cannot be used to reset the JTAG TAP controlller during first time programming of PSoC 3 as the default setting is 4-wire JTAG (nTRST disabled). Use the TMS, TCK pins to do a reset of JTAG TAP controller.

[7] If XRES pin is used by host, P1[2] will be configured as XRES by default only for 48-pin devices (without dedicated XRES pin). For devices with dedicated XRES pin, P1[2] is GPIO pin by default. So use P1[2] as Reset pin only for 48-pin devices, but use dedicated XRES pin for rest of devices.

## 9.2 Serial Wire Debug Interface

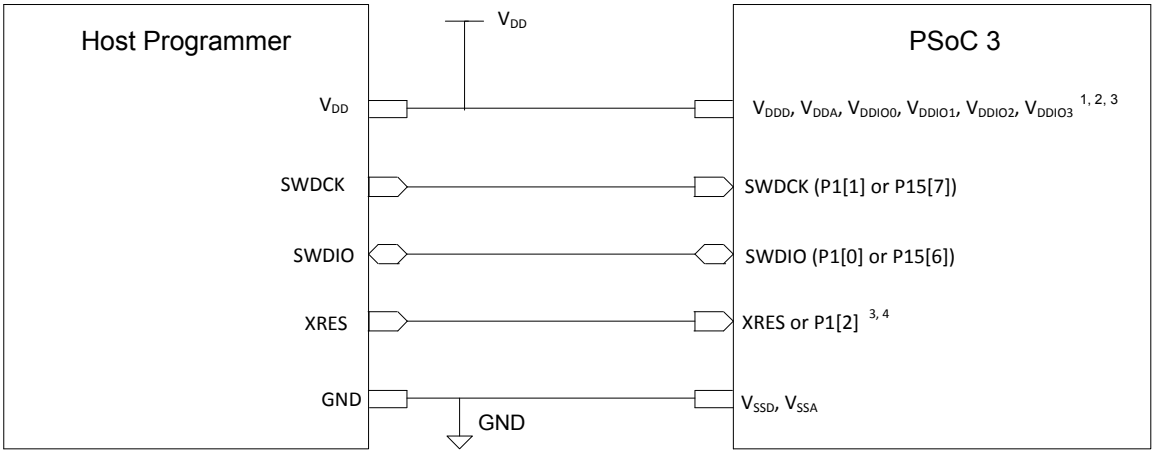
The SWD interface is the preferred alternative to the JTAG interface. It requires only two pins instead of the four or five needed by JTAG. SWD provides all of the programming and debugging features of JTAG at the same speed. SWD does not provide access to scan chains or device chaining. The SWD clock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

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

SWD can be enabled on only one of the pin pairs at a time. This only happens if, within 8 µs (key window) after reset, that pin pair (JTAG or USB) receives a predetermined sequence of 1s and 0s. SWD is used for debugging or for programming the flash memory.

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

**Figure 9-2. SWD Interface Connections between PSoC 3 and Programmer**




[1] The voltage levels of the Host Programmer and the PSoC 3 voltage domains involved in Programming should be the same. XRES pin (XRES_N or P1[2]) is powered by $V_{DDIO1}$. The USB SWD pins are powered by $V_{DDD}$. So for Programming using the USB SWD pins with XRES pin, the $V_{DDD}$, $V_{DDIO1}$ of PSoC 3 should be at the same voltage level as Host $V_{DD}$. Rest of PSoC 3 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 3 should be at same voltage level as host $V_{DD}$ for Port 1 SWD programming. Rest of PSoC 3 voltage domains ( $V_{DDD}$, $V_{DDA}$, $V_{DDIO0}$, $V_{DDIO2}$, $V_{DDIO3}$) need not be at the same voltage level as host Programmer.

[2] Vdda must be greater than or equal to all other power supplies (Vddd, Vddio's) in PSoC 3.

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

[4] P1[2] will be configured as XRES by default only for 48-pin devices (without dedicated XRES pin). For devices with dedicated XRES pin, P1[2] is GPIO pin by default. So use P1[2] as Reset pin only for 48-pin devices, but use dedicated XRES pin for rest of devices.

## 9.3 Debug Features

Using the JTAG or SWD interface, the CY8C34 supports the following debug features:

- Halt and single-step the CPU
- View and change CPU and peripheral registers, and RAM addresses
- Eight program address breakpoints
- One memory access breakpoint—break on reading or writing any memory address and data value
- Break on a sequence of breakpoints (non recursive)
- Debugging at the full speed of the CPU
- Debug operations are possible while the device is reset, or in low-power modes
- Compatible with PSoC Creator and MiniProg3 programmer and debugger
- Standard JTAG programming and debugging interfaces make CY8C34 compatible with other popular third-party tools (for example, ARM / Keil)

## 9.4 Trace Features

The CY8C34 supports the following trace features when using JTAG or SWD:

- Trace the 8051 program counter (PC), accumulator register (ACC), and one SFR / 8051 core RAM register
- Trace depth up to 1000 instructions if all registers are traced, or 2000 instructions if only the PC is traced (on devices that include trace memory)
- Program address trigger to start tracing
- Trace windowing, that is, only trace when the PC is within a given range
- Two modes for handling trace buffer full: continuous (overwriting the oldest trace data) or break when trace buffer is full

## 9.5 Single Wire Viewer Interface

The SWV interface is closely associated with SWD but can also be used independently. SWV data is output on the JTAG interface's TDO pin. If using SWV, you must configure the device for SWD, not JTAG. SWV is not supported with the JTAG interface.

SWV is ideal for application debug where it is helpful for the firmware to output data similar to 'printf' debugging on PCs. The SWV is ideal for data monitoring, because it requires only a single pin and can output data in standard UART format or Manchester encoded format. For example, it can be used to tune a PID control loop in which the output and graphing of the three error terms greatly simplifies coefficient tuning.

The following features are supported in SWV:

- 32 virtual channels, each 32 bits long
- Simple, efficient packing and serializing protocol
- Supports standard UART format (N81)

## 9.6 Programming Features

The JTAG and SWD interfaces provide full programming support. The entire device can be erased, programmed, and verified. You 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 3 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 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 (0×50536F43); 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 (0×50536F43) 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.

The user can write the key into the WOL to lock out external access only if no flash protection is set (see "Flash Security" on page 21). However, after setting the values in the WOL, a user still has access to the part until it is reset. Therefore, a user 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 SWD port to electrically identify protected parts. The user 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 3 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 CY8C34 family has a rich set of documentation, development tools, and online resources to assist you during your development process. Visit psoc.cypress.com/getting-started to find out more.

## 10.1 Documentation

A suite of documentation, supports the CY8C34 family to ensure that you can find answers to your questions quickly. 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**: The Technical Reference Manual (TRM) contains all the technical detail you need to use a PSoC device, including a complete description of all PSoC registers.

## 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 CY8C34 family is part of a development tool ecosystem. Visit us at 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 °C ≤ $T_A$ ≤ 85 °C and $T_J$ ≤ 100 °C, except where noted. Specifications are valid for 1.71 V to 5.5 V, except where noted. The unique flexibility of the PSoC UDBs and analog blocks enable many functions to be implemented in PSoC Creator components, see the component data sheets for full AC/DC specifications of individual functions. See the "Example Peripherals" section on page 39 for further explanation of PSoC Creator components.

## 11.1 Absolute Maximum Ratings

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{STG}$ | Storage temperature | Higher storage temperatures reduce NVL data retention time. Recommended storage temperature is +25 °C ±25 °C. Extended duration storage temperatures above 85 °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 |
| $V_{GPIO}$[17] | 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 |
| Ivddio | Current per $V_{DDIO}$ supply pin | | – | – | 100 | mA |
| Vextref | ADC external reference inputs | Pins P0[3], P3[2] | – | – | 2 | V |
| LU | Latch up current[18] | | –140 | – | 140 | mA |
| $ESD_{HBM}$ | Electrostatic discharge voltage | Human body model | 750 | – | – | 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**
17. The $V_{DDIO}$ supply voltage must be greater than the maximum analog voltage on the associated GPIO pins. Maximum analog voltage on GPIO pin ≤ $V_{DDIO}$ ≤ $V_{DDA}$.
18. Meets or exceeds JEDEC Spec EIA/JESD78 IC Latch-up Test.

## 11.2 Device Level Specifications

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

### *11.2.1 Device Level Specifications*

**Table 11-2. DC Specifications**

| Parameter | Description | Conditions | | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|---|
| $V_{DDA}$ | Analog supply voltage and input to analog core regulator | Analog core regulator enabled | | 1.8 | – | 5.5 | V |
| $V_{DDA}$ | Analog supply voltage, analog regulator bypassed | Analog core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| $V_{DDD}$ | Digital supply voltage relative to $V_{SSD}$ | Digital core regulator enabled | | 1.8 | – | $V_{DDA}$[19] | V |
| $V_{DDD}$ | Digital supply voltage, digital regulator bypassed | Digital core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| $V_{DDIO}$[20] | I/O supply voltage relative to $V_{SSIO}$ | | | 1.71 | – | $V_{DDA}$[19] | V |
| $V_{CCA}$ | Direct analog core voltage input (Analog regulator bypass) | Analog core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| $V_{CCD}$ | Direct digital core voltage input (Digital regulator bypass) | Digital core regulator disabled | | 1.71 | 1.8 | 1.89 | V |
| $I_{DD}$[21] | **Active Mode, $V_{DD}$ = 1.71 V – 5.5 V** | | | | | | |
| | Bus clock off. Execute from CPU instruction buffer. See "Flash Program Memory" on page 21. | CPU at 3 MHz | T = –40 °C | – | – | – | mA |
| | | | T = 25 °C | – | 0.8 | – | mA |
| | | | T = 85 °C | – | – | – | mA |
| | | CPU at 6 MHz | T = –40 °C | – | – | – | mA |
| | | | T = 25 °C | – | 1.2 | – | mA |
| | | | T = 85 °C | – | – | – | mA |
| | | CPU at 12 MHz | T = –40 °C | – | – | – | mA |
| | | | T = 25 °C | – | 2.0 | – | mA |
| | | | T = 85 °C | – | – | – | mA |
| | | CPU at 24 MHz | T = –40 °C | – | – | – | mA |
| | | | T = 25 °C | – | 3.5 | – | mA |
| | | | T = 85 °C | – | – | – | mA |
| | | CPU at 48 MHz | T = –40 °C | – | – | – | mA |
| | | | T = 25 °C | – | 6.6 | – | mA |
| | | | T = 85 °C | – | – | – | mA |
| | $V_{DD}$ = 3.3 V, T = 25 °C, IMO and bus clock enabled, ILO = 1 kHz, CPU executing from flash and accessing SRAM, all other blocks off, all I/Os tied low. | CPU at 3 MHz | | – | 1.4 | – | mA |
| | | CPU at 6 MHz | | – | 2.2 | – | mA |
| | | CPU at 12 MHz | | – | 3.6 | – | mA |
| | | CPU at 24 MHz | | – | 6.4 | – | mA |
| | | CPU at 48 MHz | | – | 11.8 | – | mA |

**Notes**
19. 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.
20. The $V_{DDIO}$ supply voltage must be greater than the maximum analog voltage on the associated GPIO pins. Maximum analog voltage on GPIO pin ≤ $V_{DDIO}$ ≤ $V_{DDA}$.
21. 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.

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

| Parameter | Description | Conditions | | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|---|
| | **Sleep Mode**[22] | | | | | | |
| | CPU = OFF<br>RTC = ON (= ECO32K ON, in low-power mode)<br>Sleep timer = ON (= ILO ON at 1 kHz)[23]<br>WDT = OFF<br>$I^2C$ Wake = OFF<br>Comparator = OFF<br>POR = ON<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO}$ = 4.5 – 5.5 V | T = –40 °C | – | – | – | µA |
| | | | T = 25 °C | – | – | – | µA |
| | | | T = 85 °C | – | – | – | µA |
| | | $V_{DD} = V_{DDIO}$ = 2.7 – 3.6 V | T = –40 °C | – | – | – | µA |
| | | | T = 25 °C | – | 1 | – | µA |
| | | | T = 85 °C | – | – | – | µA |
| | | $V_{DD} = V_{DDIO}$ = 1.71 – 1.95 V | T = –40 °C | – | – | – | µA |
| | | | T = 25 °C | – | – | – | µA |
| | | | T = 85 °C | – | – | – | µA |
| | Comparator = ON<br>CPU = OFF<br>RTC = OFF<br>Sleep timer = OFF<br>WDT = OFF<br>I2C Wake = OFF<br>POR = ON<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO}$ = 2.7 – 3.6V | T = 25 °C | – | – | – | µA |
| | I2C Wake = ON<br>CPU = OFF<br>RTC = OFF<br>Sleep timer = OFF<br>WDT = OFF<br>Comparator = OFF<br>POR = ON<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO}$ = 2.7 – 3.6V | T = 25 °C | – | – | – | µA |
| | **Hibernate Mode**[22] | | | | | | |
| | Hibernate mode current<br>All regulators and oscillators off.<br>SRAM retention<br>GPIO interrupts are active<br>Boost = OFF<br>SIO pins in single ended input, unregulated output mode | $V_{DD} = V_{DDIO}$ = 4.5 – 5.5 V | T = –40 °C | – | – | – | nA |
| | | | T = 25 °C | – | – | – | nA |
| | | | T = 85 °C | – | – | – | nA |
| | | $V_{DD} = V_{DDIO}$ = 2.7 – 3.6 V | T = –40 °C | – | – | – | nA |
| | | | T = 25 °C | – | 200 | – | nA |
| | | | T = 85 °C | – | – | – | nA |
| | | $V_{DD} = V_{DDIO}$ = 1.71 – 1.95 V | T = –40 °C | – | – | – | nA |
| | | | T = 25 °C | – | – | – | nA |
| | | | T = 85 °C | – | – | – | nA |

**Notes**
22. If $V_{CCD}$ and $V_{CCA}$ are externally regulated, the voltage difference between $V_{CCD}$ and $V_{CCA}$ must be less than 50 mV.
23. Sleep timer generates periodic interrupts to wake up the CPU. This specification applies only to those times that the CPU is off.

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

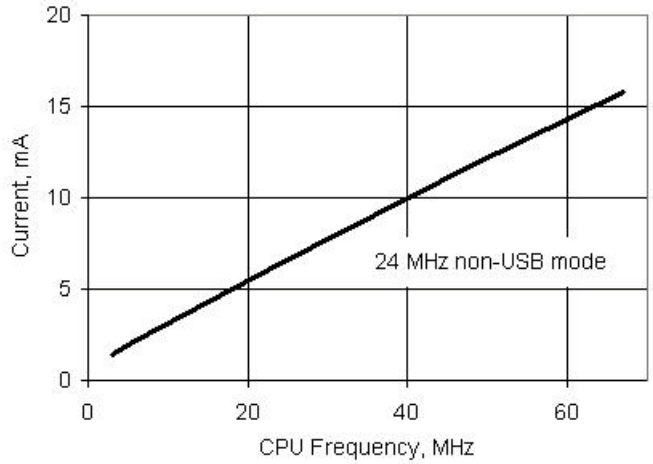



**Figure 11-2. Active Mode Current vs Temperature and $F_{CPU}$, $V_{DD}$ = 3.3 V**

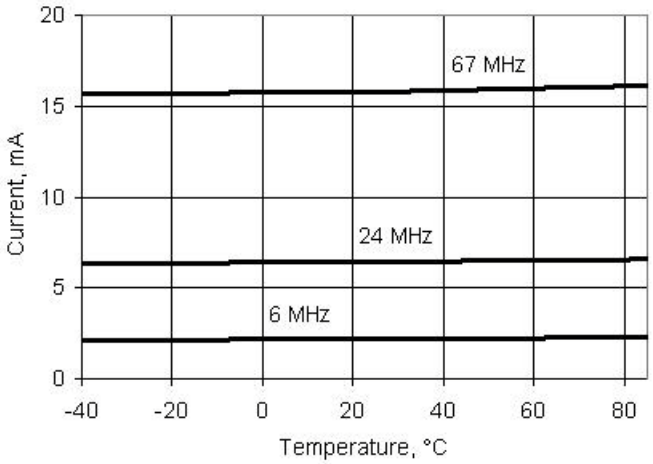



**Figure 11-3. Active Mode Current vs $V_{DD}$ and Temperature, $F_{CPU}$ = 24 MHz**

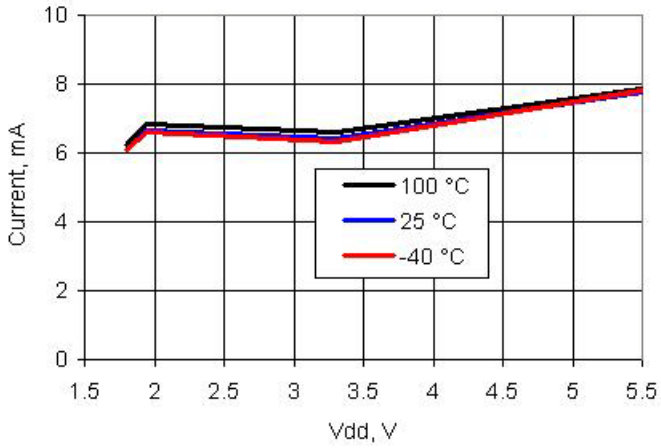
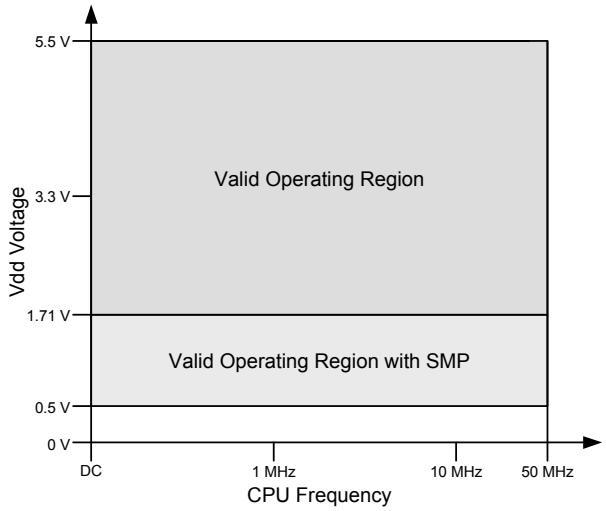


**Table 11-3. AC Specifications**[24]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{CPU}$ | CPU frequency | 1.71 V ≤ $V_{DDD}$ ≤ 5.5 V | DC | – | 50.01 | MHz |
| $F_{BUSCLK}$ | Bus frequency | 1.71 V ≤ $V_{DDD}$ ≤ 5.5 V | DC | – | 50.01 | MHz |
| Svdd | $V_{DD}$ ramp rate | – | – | – | 1 | V/ns |
| $T_{IO_INIT}$ | Time from $V_{DDD}$/$V_{DDA}$/$V_{CCD}$/$V_{CCA}$ ≥ IPOR to I/O ports set to their reset states | – | – | – | 10 | µs |
| $T_{STARTUP}$ | Time from $V_{DDD}$/$V_{DDA}$/$V_{CCD}$/$V_{CCA}$ ≥ PRES to CPU executing code at reset vector | $V_{CCA}$/$V_{CCD}$ = regulated from $V_{DDA}$/$V_{DDD}$, no PLL used, IMO boot mode (12 MHz typ.) | – | – | 66 | µs |
| $T_{SLEEP}$ | Wakeup from sleep mode – Application of non-LVD interrupt to beginning of execution of next CPU instruction | | – | – | 15 | µs |
| $T_{HIBERNATE}$ | Wakeup from hibernate mode – Application of external interrupt to beginning of execution of next CPU instruction | | – | – | 100 | µs |

**Figure 11-4. $F_{CPU}$ vs. $V_{DD}$**




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

## 11.3 Power Regulators

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

### *11.3.1 Digital Core Regulator*

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{DDD}$ | Input voltage | | 1.8 | – | 5.5 | V |
| $V_{CCD}$ | Output voltage | | – | 1.80 | – | V |
| | Regulator output capacitor | ±10%, X5R ceramic or better. The two $V_{CCD}$ pins must be shorted together, with as short a trace as possible, see Power System on page 28 | – | 1 | – | µF |

**Figure 11-5. Regulators $V_{CC}$ vs $V_{DD}$**

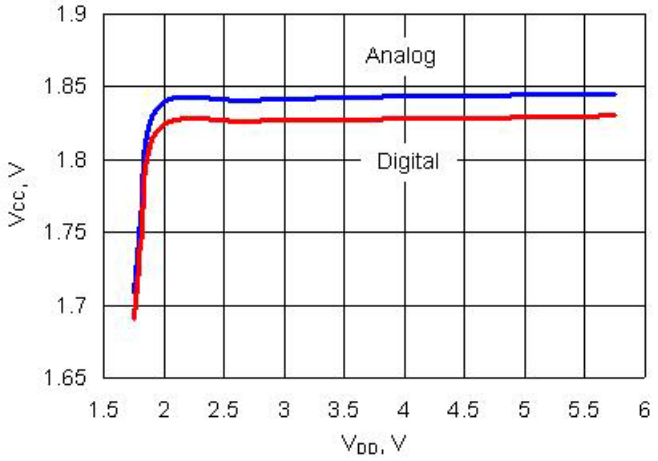
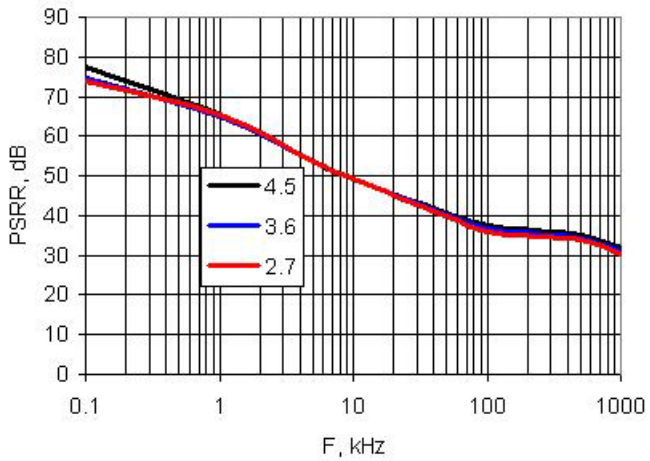



**Figure 11-6. Digital Regulator PSRR vs Frequency and $V_{DD}$**

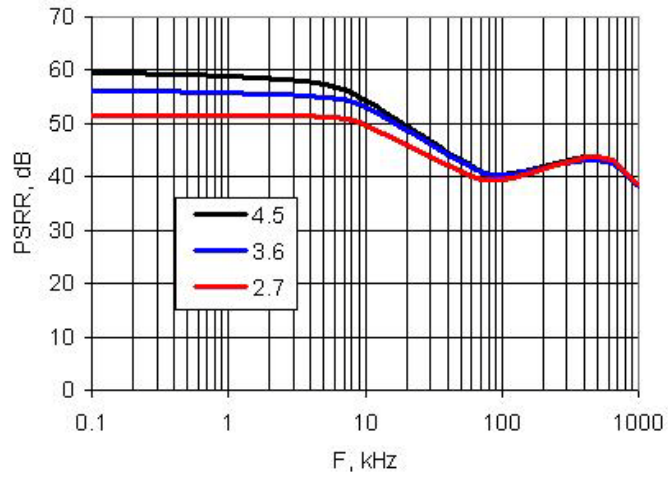



### *11.3.2 Analog Core Regulator*

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

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

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



*11.3.3 Inductive Boost Regulator.*

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

Unless otherwise specified, operating conditions are: $V_{BAT}$ = 2.4 V, $V_{OUT}$ = 2.7 V, $I_{OUT}$ = 40 mA, $F_{SW}$ = 400 kHz, $L_{BOOST}$ = 10 µH, $C_{BOOST}$ = 22 µF || 0.1 µF

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{BAT}$ | Input voltage Includes startup | T=-35 °C to +65 °C | 0.5 | – | 3.6 | V |
| | | Over entire temperature range | 0.68 | – | 3.6 | V |
| $I_{OUT}$ | Load current[25, 26] | $V_{BAT}$ = 1.6 – 3.6 V, $V_{OUT}$ = 3.6 – 5.0 V, external diode | – | – | 50 | mA |
| | | $V_{BAT}$ = 1.6 – 3.6 V, $V_{OUT}$ = 1.6 – 3.6 V, internal diode | – | – | 75 | mA |
| | | $V_{BAT}$ = 0.8 – 1.6 V, $V_{OUT}$ = 1.6 – 3.6 V, internal diode | – | – | 30 | mA |
| | | $V_{BAT}$ = 0.8 – 1.6 V, $V_{OUT}$ = 3.6 – 5.0 V, external diode | – | – | 20 | mA |
| | | $V_{BAT}$ = 0.5 – 0.8 V, $V_{OUT}$ = 1.6 – 3.6 V, internal diode | – | – | 15 | mA |
| $I_{LPK}$ | Inductor peak current | | – | – | 700 | mA |
| $I_Q$ | Quiescent current | Boost active mode | – | 200 | – | µA |
| | | Boost standby mode, 32 khz external crystal oscillator, $I_{OUT}$ < 1 µA | – | 12 | – | µA |
| $V_{OUT}$ | Boost voltage range[27, 28] | | | | | |
| | 1.8 V | | 1.71 | 1.80 | 1.89 | V |
| | 1.9 V | | 1.81 | 1.90 | 2.00 | V |
| | 2.0 V | | 1.90 | 2.00 | 2.10 | V |
| | 2.4 V | | 2.28 | 2.40 | 2.52 | V |
| | 2.7 V | | 2.57 | 2.70 | 2.84 | V |
| | 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 | % |
| η | Efficiency | $L_{BOOST}$ = 10 µH | 70 | 85 | – | % |
| | | $L_{BOOST}$ = 22 µH | 82 | 90 | – | % |

**Notes**
25. For output voltages above 3.6 V, an external diode is required.
26. Maximum output current applies for output voltages ≤ 4x input voltage.
27. Based on device characterization (Not production tested).
28. At boost frequency of 2 MHz, $V_{OUT}$ is limited to 2 x $V_{BAT}$. At 400 kHz, $V_{OUT}$ is limited to 4 x $V_{BAT}$.

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

Unless otherwise specified, operating conditions are: $V_{BAT}$ = 2.4 V, $V_{OUT}$ = 2.7 V, $I_{OUT}$ = 40 mA, $F_{SW}$ = 400 kHz, $L_{BOOST}$ = 10 µH, $C_{BOOST}$ = 22 µF || 0.1 µF.

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{RIPPLE}$ | Ripple voltage (peak-to-peak) | $V_{OUT}$ = 1.8 V, $F_{SW}$ = 400 kHz, $I_{OUT}$ = 10 mA | – | – | 100 | mV |
| $F_{SW}$ | Switching frequency | | – | 0.1, 0.4, or 2 | – | MHz |

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $L_{BOOST}$ | Boost inductor | | 4.7 | 10 | 47 | µH |
| $C_{BOOST}$ | Filter capacitor[29] | | 10 | 22 | 47 | µF |
| $I_F$ | External Schottky diode average forward current | External Schottky diode is required for $V_{OUT}$ > 3.6 V | 1 | – | – | A |
| $V_R$ | | | 20 | – | – | V |

**Figure 11-8. Efficiency vs $V_{OUT}$**
$I_{OUT}$ = 30 mA, $V_{BAT}$ ranges from 0.7 V to $V_{OUT}$, $L_{BOOST}$ = 22 µH

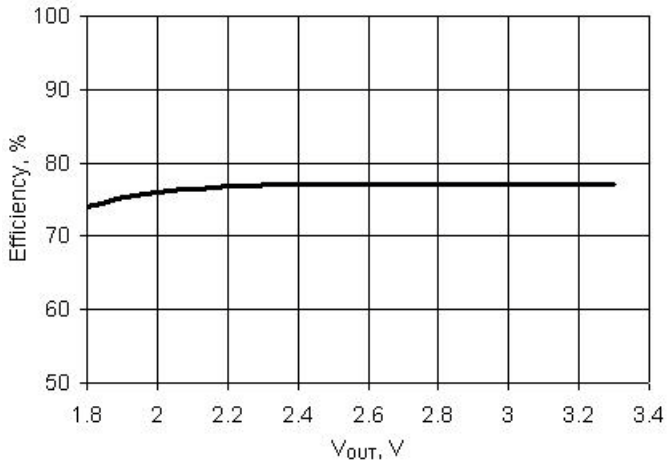
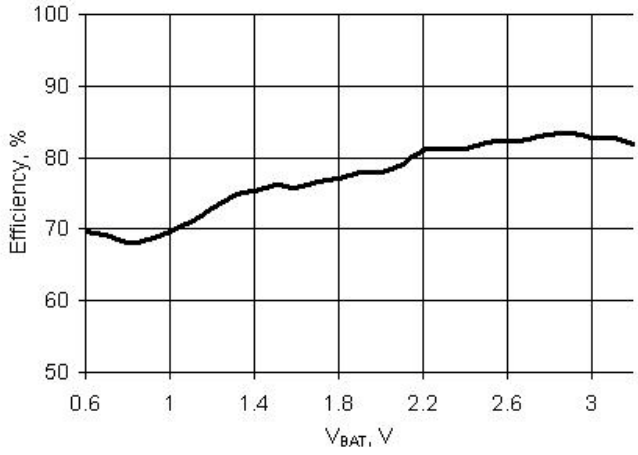



**Figure 11-9. Efficiency vs $V_{BAT}$**
$I_{OUT}$ = 30 mA, $V_{OUT}$ = 3.3 V, $L_{BOOST}$ = 22 µH




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

**Figure 11-10. Efficiency vs $I_{OUT}$**
$V_{BAT}$ = 2.4 V, $V_{OUT}$ = 3.3 V

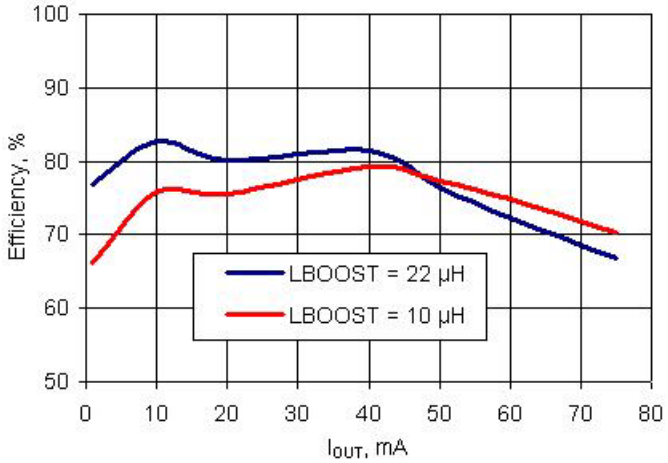



**Figure 11-11. Efficiency vs $I_{OUT}$**
$V_{BAT}$ ranges from 0.7 V to 3.3 V, $L_{BOOST}$ = 22 µH

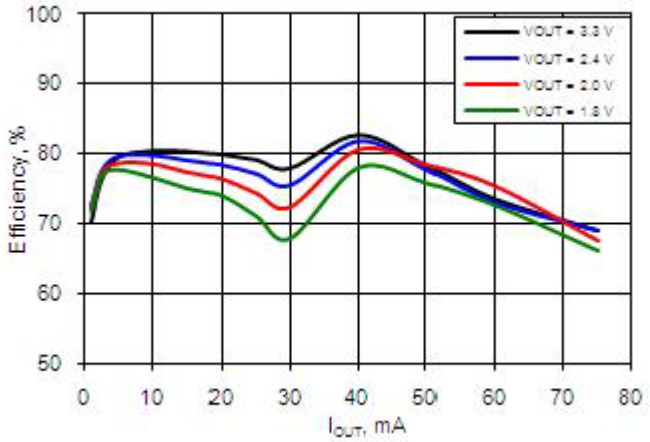
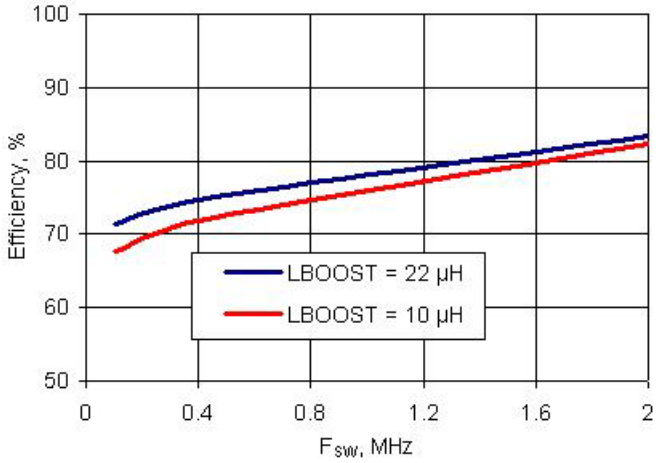



**Figure 11-12. Efficiency vs Switching Frequency**
$V_{OUT}$ = 3.3 V, $V_{BAT}$ = 2.4 V, $I_{OUT}$ = 40 mA



## 11.4 Inputs and Outputs

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

### *11.4.1 GPIO*

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IH}$ | Input voltage high threshold | CMOS Input, PRT[×]CTL = 0 | 0.7 × $V_{DDIO}$ | – | – | V |
| $V_{IL}$ | Input voltage low threshold | CMOS Input, PRT[×]CTL = 0 | – | – | 0.3 × $V_{DDIO}$ | V |
| $V_{IH}$ | Input voltage high threshold | LVTTL Input, PRT[×]CTL = 1,$V_{DDIO}$ < 2.7 V | 0.7 × $V_{DDIO}$ | – | – | V |
| $V_{IH}$ | Input voltage high threshold | LVTTL Input, PRT[×]CTL = 1, $V_{DDIO}$ ≥ 2.7V | 2.0 | – | – | V |
| $V_{IL}$ | Input voltage low threshold | LVTTL Input, PRT[×]CTL = 1,$V_{DDIO}$ < 2.7 V | – | – | 0.3 × $V_{DDIO}$ | V |
| $V_{IL}$ | Input voltage low threshold | LVTTL Input, PRT[×]CTL = 1, $V_{DDIO}$ ≥ 2.7V | – | – | 0.8 | V |
| $V_{OH}$ | Output voltage high | $I_{OH}$ = 4 mA at 3.3 $V_{DDIO}$ | $V_{DDIO}$ – 0.6 | – | – | V |
| | | $I_{OH}$ = 1 mA at 1.8 $V_{DDIO}$ | $V_{DDIO}$ – 0.5 | – | – | V |
| $V_{OL}$ | Output voltage low | $I_{OL}$ = 8 mA at 3.3 $V_{DDIO}$ | – | – | 0.6 | V |
| | | $I_{OL}$ = 4 mA at 1.8 $V_{DDIO}$ | – | – | 0.6 | V |
| Rpullup | Pull-up resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| Rpulldown | Pull-down resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| $I_{IL}$ | Input leakage current (absolute value)[30] | 25 °C, $V_{DDIO}$ = 3.0 V | – | – | 2 | nA |
| $C_{IN}$ | Input capacitance[30] | GPIOs without opamp outputs | – | – | 7 | pF |
| | | GPIOs with opamp outputs | – | – | 18 | pF |
| $V_H$ | Input voltage hysteresis (Schmitt-Trigger)[30] | | – | 40 | – | mV |
| Idiode | Current through protection diode to $V_{DDIO}$ and $V_{SSIO}$ | | – | – | 100 | µA |
| Rglobal | Resistance pin to analog global bus | 25 °C, $V_{DDIO}$ = 3.0 V | – | 320 | – | Ω |
| Rmux | Resistance pin to analog mux bus | 25 °C, $V_{DDIO}$ = 3.0 V | – | 220 | – | Ω |

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

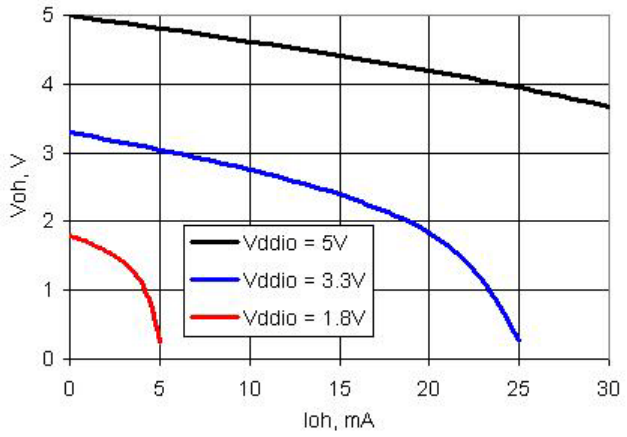
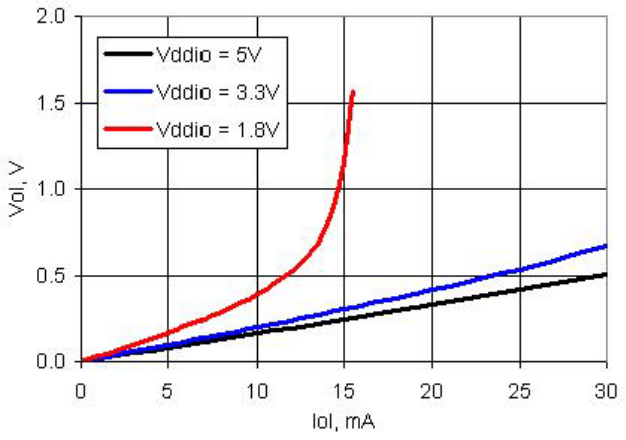



**Figure 11-14. GPIO Output Low Voltage and Current**




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

**Table 11-10. GPIO AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| TriseF | Rise time in Fast Strong Mode[31] | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 12 | ns |
| TfallF | Fall time in Fast Strong Mode[31] | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 12 | ns |
| TriseS | Rise time in Slow Strong Mode[31] | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 60 | ns |
| TfallS | Fall time in Slow Strong Mode[31] | 3.3 V $V_{DDIO}$ Cload = 25 pF | – | – | 60 | ns |
| Fgpioout | GPIO output operating frequency | | | | | |
| | 2.7 V ≤ $V_{DDIO}$ ≤ 5.5 V, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 33 | MHz |
| | 1.71 V ≤ $V_{DDIO}$ < 2.7 V, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 20 | MHz |
| | 3.3 V ≤ $V_{DDIO}$ ≤ 5.5 V, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 7 | MHz |
| | 1.71 V ≤ $V_{DDIO}$ < 3.3 V, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 3.5 | MHz |
| Fgpioin | GPIO input operating frequency | | | | | |
| | 1.71 V ≤ $V_{DDIO}$ ≤ 5.5 V | 90/10% $V_{DDIO}$ | – | – | 50 | MHz |

**Figure 11-15. GPIO Output Rise and Fall Times, Fast Strong Mode, $V_{DDIO}$ = 3.3 V, 25 pF Load**

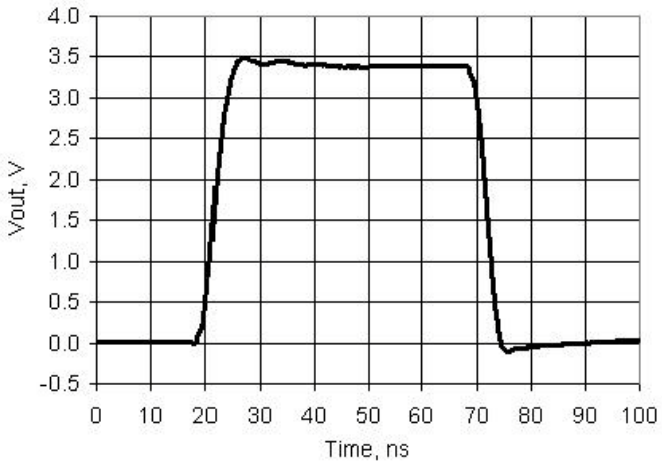
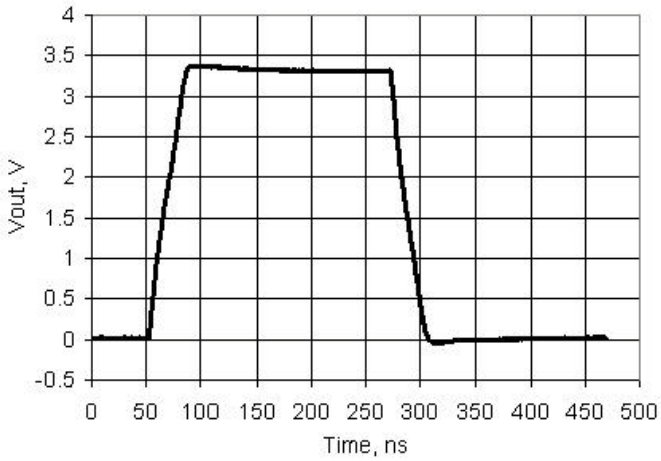



**Figure 11-16. GPIO Output Rise and Fall Times, Slow Strong Mode, $V_{DDIO}$ = 3.3 V, 25 pF Load**




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

### 11.4.2 SIO

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Vinmax | Maximum input voltage | All allowed values of Vddio and Vddd, see *Section 11.2.1* | – | – | 5.5 | V |
| Vinref | Input voltage reference (Differential input mode) | | 0.5 | – | $0.52 \times V_{DDIO}$ | V |
| Voutref | Output voltage reference (Regulated output mode) | | | | | |
| | | $V_{DDIO} > 3.7$ | 1 | – | $V_{DDIO} - 1$ | V |
| | | $V_{DDIO} < 3.7$ | 1 | – | $V_{DDIO} - 0.5$ | V |
| $V_{IH}$ | Input voltage high threshold | | | | | |
| | GPIO mode | CMOS input | $0.7 \times V_{DDIO}$ | – | – | V |
| | Differential input mode[32] | Hysteresis disabled | SIO_ref + 0.2 | – | – | V |
| $V_{IL}$ | Input voltage low threshold | | | | | |
| | GPIO mode | CMOS input | – | – | $0.3 \times V_{DDIO}$ | V |
| | Differential input mode[32] | Hysteresis disabled | – | – | SIO_ref – 0.2 | V |
| $V_{OH}$ | Output voltage high | | | | | |
| | Unregulated mode | $I_{OH}$ = 4 mA, $V_{DDIO}$ = 3.3 V | $V_{DDIO} - 0.4$ | – | – | V |
| | Regulated mode[32] | $I_{OH}$ = 1 mA | SIO_ref – 0.65 | – | SIO_ref + 0.2 | V |
| | Regulated mode[32] | $I_{OH}$ = 0.1 mA | SIO_ref – 0.3 | – | SIO_ref + 0.2 | V |
| $V_{OL}$ | Output voltage low | | | | | |
| | | $V_{DDIO}$ = 3.30 V, $I_{OL}$ = 25 mA | – | – | 0.8 | V |
| | | $V_{DDIO}$ = 1.80 V, $I_{OL}$ = 4 mA | – | – | 0.4 | V |
| Rpullup | Pull-up resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| Rpulldown | Pull-down resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| $I_{IL}$ | Input leakage current (absolute value)[33] | | | | | |
| | $V_{IH} \leq$ Vddsio | 25 °C, Vddsio = 3.0 V, $V_{IH}$ = 3.0 V | – | – | 14 | nA |
| | $V_{IH}$ > Vddsio | 25 °C, Vddsio = 0 V, $V_{IH}$ = 3.0 V | – | – | 10 | µA |
| $C_{IN}$ | Input Capacitance[33] | | – | – | 7 | pF |
| $V_H$ | Input voltage hysteresis (Schmitt-Trigger)[33] | Single ended mode (GPIO mode) | – | 40 | – | mV |
| | | Differential mode | – | 35 | – | mV |
| Idiode | Current through protection diode to $V_{SSIO}$ | | – | – | 100 | µA |

**Notes**
32. See Figure 6-9 on page 34 and Figure 6-12 on page 37 for more information on SIO reference.
33. Based on device characterization (Not production tested).

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

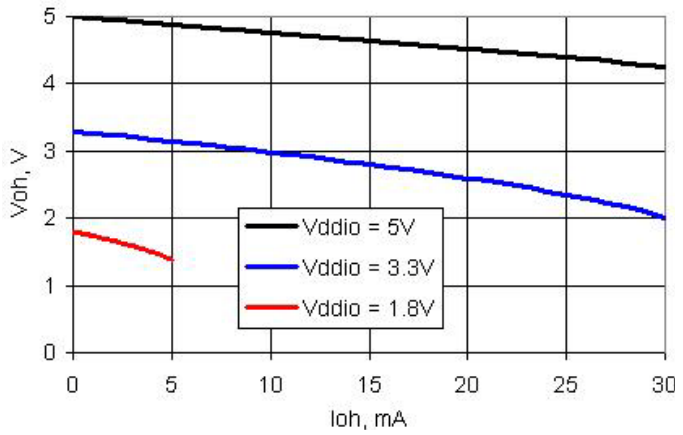



**Figure 11-18. SIO Output Low Voltage and Current, Unregulated Mode**

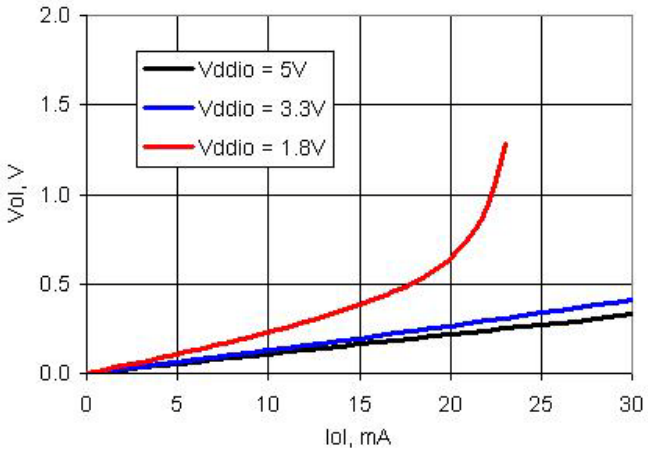



**Figure 11-19. SIO Output High Voltage and Current, Regulated Mode**

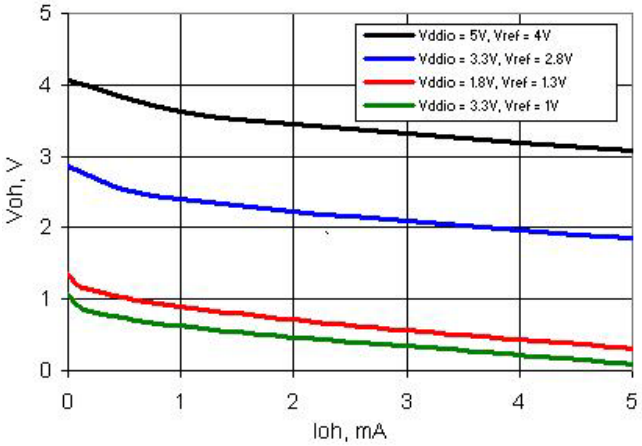



**Table 11-12. SIO AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| TriseF | Rise time in Fast Strong Mode (90/10%)[34] | Cload = 25 pF, $V_{DDIO}$ = 3.3 V | – | – | 12 | ns |
| TfallF | Fall time in Fast Strong Mode (90/10%)[34] | Cload = 25 pF, $V_{DDIO}$ = 3.3 V | – | – | 12 | ns |
| TriseS | Rise time in Slow Strong Mode (90/10%)[34] | Cload = 25 pF, $V_{DDIO}$ = 3.0 V | – | – | 75 | ns |
| TfallS | Fall time in Slow Strong Mode (90/10%)[34] | Cload = 25 pF, $V_{DDIO}$ = 3.0 V | – | – | 60 | ns |

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

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Fsioout | SIO output operating frequency | | | | | |
| | 2.7 V < $V_{DDIO}$ < 5.5 V, Unregulated output (GPIO) mode, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 33 | MHz |
| | 1.71 V < $V_{DDIO}$ < 2.7 V, Unregulated output (GPIO) mode, fast strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 16 | MHz |
| | 3.3 V < $V_{DDIO}$ < 5.5 V, Unregulated output (GPIO) mode, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 5 | MHz |
| | 1.71 V < $V_{DDIO}$ < 3.3 V, Unregulated output (GPIO) mode, slow strong drive mode | 90/10% $V_{DDIO}$ into 25 pF | – | – | 4 | MHz |
| | 2.7 V < $V_{DDIO}$ < 5.5 V, Regulated output mode, fast strong drive mode | Output continuously switching into 25 pF | – | – | 20 | MHz |
| | 1.71 V < $V_{DDIO}$ < 2.7 V, Regulated output mode, fast strong drive mode | Output continuously switching into 25 pF | – | – | 10 | MHz |
| | 1.71 V < $V_{DDIO}$ < 5.5 V, Regulated output mode, slow strong drive mode | Output continuously switching into 25 pF | – | – | 2.5 | MHz |
| Fsioin | SIO input operating frequency | | | | | |
| | 1.71 V ≤ $V_{DDIO}$ ≤ 5.5 V | 90/10% $V_{DDIO}$ | – | – | 50 | MHz |

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

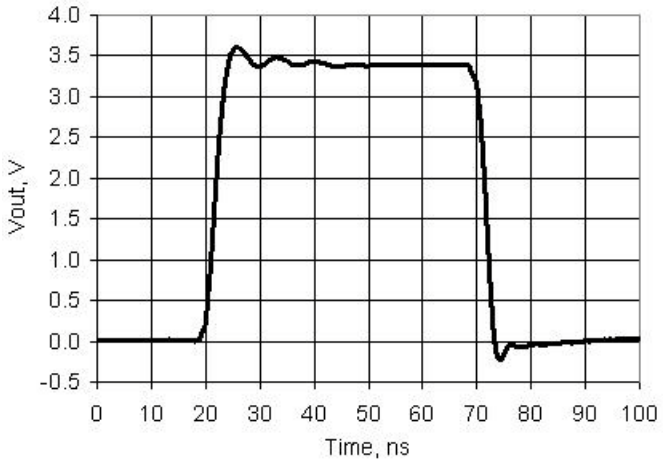
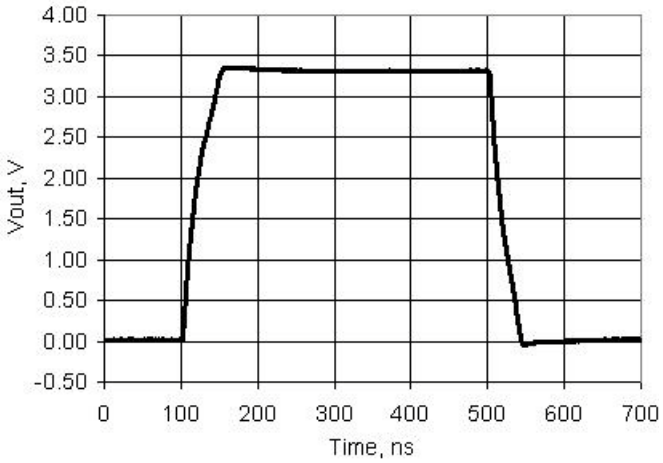



**Figure 11-21. SIO Output Rise and Fall Times, Slow Strong Mode, $V_{DDIO}$ = 3.3 V, 25 pF Load**




*11.4.3 USBIO*

For operation in GPIO mode, the standard range for $V_{DDD}$ applies, see Device Level Specifications on page 66.

**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 | kΩ |
| Rusba | USB D+ pull-up resistance | While receiving traffic | 1.425 | – | 3.090 | kΩ |
| Vohusb | Static output high | 15 kΩ ±5% to Vss, internal pull-up enabled | 2.8 | – | 3.6 | V |
| Volusb | Static output low | 15 kΩ ±5% to Vss, internal pull-up enabled | – | – | 0.3 | V |

**Table 11-13. USBIO DC Specifications** *(continued)*

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Vihgpio | Input voltage high, GPIO mode | $V_{DDD} \geq 3$ V | 2 | – | – | V |
| Vilgpio | Input voltage low, GPIO mode | $V_{DDD} \geq 3$ V | – | – | 0.8 | V |
| Vohgpio | Output voltage high, GPIO mode | $I_{OH}$ = 4 mA, $V_{DDD} \geq 3$ V | 2.4 | – | – | V |
| Volgpio | Output voltage low, GPIO mode | $I_{OL}$ = 4 mA, $V_{DDD} \geq 3$ 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 | kΩ |
| Rext | External USB series resistor | In series with each USB pin | 21.78 (–1%) | 22 | 22.22 (+1%) | Ω |
| Zo | USB driver output impedance | Including Rext | 28 | – | 44 | Ω |
| $C_{IN}$ | USB transceiver input capacitance | | – | – | 20 | pF |
| $I_{IL}$ | Input leakage current (absolute value) | 25 °C, $V_{DDD}$ = 3.0 V | – | – | 2 | nA |

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

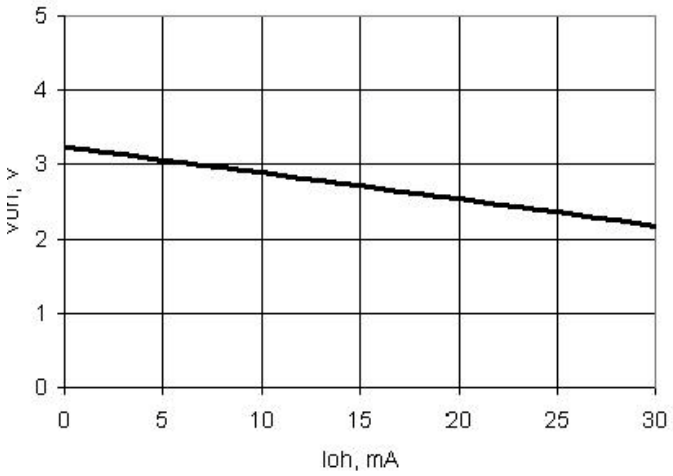



**Figure 11-23. USBIO Output Low Voltage and Current, GPIO Mode**

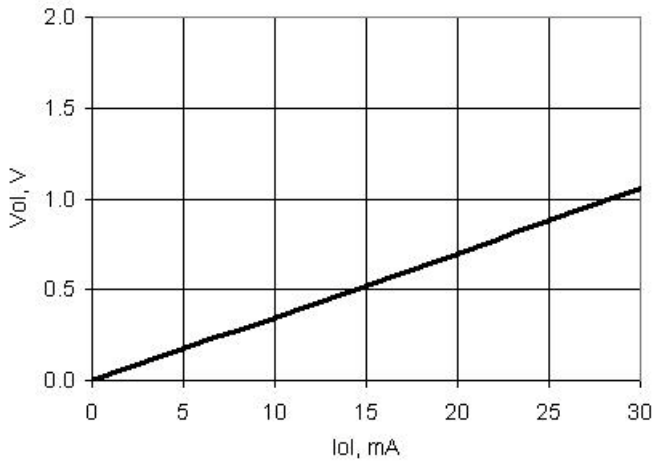


**Table 11-14. USBIO AC Specifications**

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

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

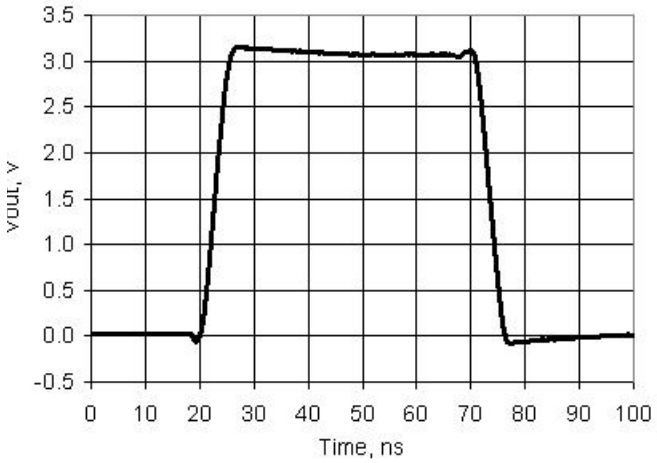



**Table 11-15. USB Driver AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Tr | Transition rise time | | – | – | 20 | ns |
| Tf | Transition fall time | | – | – | 20 | ns |
| TR | Rise/fall time matching | $V_{USB_5}$, $V_{USB_3.3}$, see USB DC Specifications on page 101 | 90% | – | 111% | |
| Vcrs | Output signal crossover voltage | | 1.3 | – | 2 | V |

### 11.4.4 XRES

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IH}$ | Input voltage high threshold | | $0.7 \times V_{DDIO}$ | – | – | V |
| $V_{IL}$ | Input voltage low threshold | | – | – | $0.3 \times V_{DDIO}$ | V |
| Rpullup | Pull-up resistor | | 3.5 | 5.6 | 8.5 | kΩ |
| $C_{IN}$ | Input capacitance[35] | | – | 3 | – | pF |
| $V_H$ | Input voltage hysteresis (Schmitt-Trigger)[35] | | – | 100 | – | mV |
| Idiode | Current through protection diode to $V_{DDIO}$ and $V_{SSIO}$ | | – | – | 100 | µA |

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

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

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

## 11.5 Analog Peripherals

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

### *11.5.1 Opamp*

**Table 11-18. Opamp DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IOFF}$ | Input offset voltage | | – | – | 2 | mV |
| Vos | Input offset voltage | | – | – | 2.5 | mV |
| | | Operating temperature –40 °C to 70 °C | – | – | 2 | mV |
| TCVos | Input offset voltage drift with temperature | Power mode = high | – | – | ±30 | µV / °C |
| Ge1 | Gain error, unity gain buffer mode | Rload = 1 kΩ | – | – | ±0.1 | % |
| Cin | Input capacitance | Routing from pin | – | – | 18 | pF |
| Vo | Output voltage range | 1 mA, source or sink, power mode = high | $V_{SSA}$ + 0.05 | – | $V_{DDA}$ – 0.05 | V |
| Iout | Output current, source or sink | $V_{SSA}$ + 500 mV ≤ Vout ≤ $V_{DDA}$ –500 mV, $V_{DDA}$ > 2.7 V | 25 | – | – | mA |
| | | $V_{SSA}$ + 500 mV ≤ Vout ≤ $V_{DDA}$ –500 mV, 1.7 V = $V_{DDA}$ ≤ 2.7 V | 16 | – | – | mA |
| Idd | Quiescent current | Power mode = min | – | 200 | 270 | uA |
| | | Power mode = low | – | 250 | 400 | uA |
| | | Power mode = med | – | 330 | 950 | uA |
| | | Power mode = high | – | 1000 | 2500 | uA |
| CMRR | Common mode rejection ratio | | 80 | – | – | dB |
| PSRR | Power supply rejection ratio | Vdda ≥ 2.7 V | 85 | – | – | dB |
| | | Vdda < 2.7 V | 70 | – | – | dB |

**Figure 11-25. Opamp Voffset Histogram, 3388 samples/847 parts, 25 °C, Vdda = 5 V**

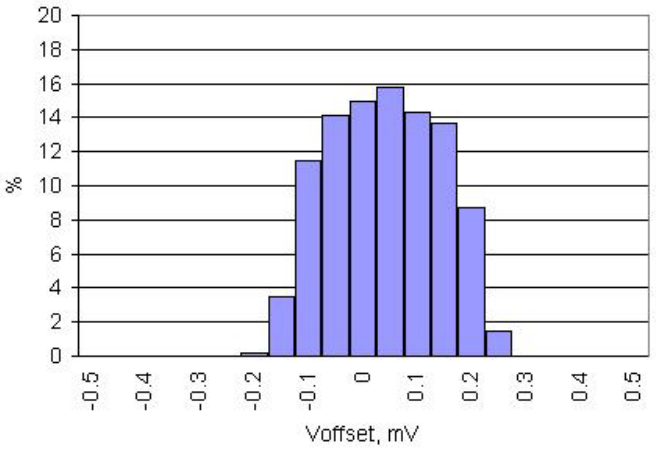



**Figure 11-26. Opamp Voffset vs Temperature, Vdda = 5V**

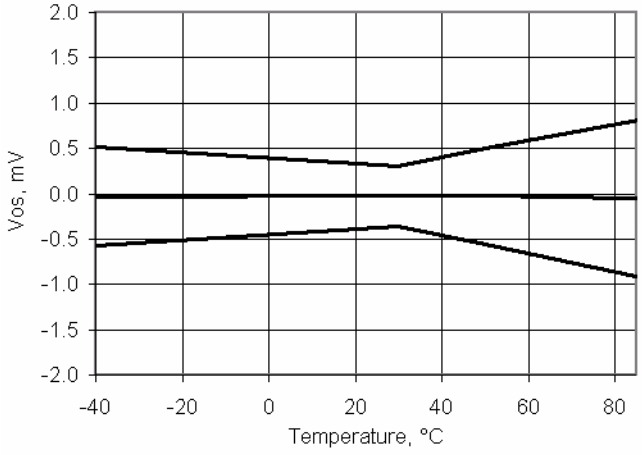


**Figure 11-27. Opamp Voffset vs Vcommon and Vdda, 25 °C**

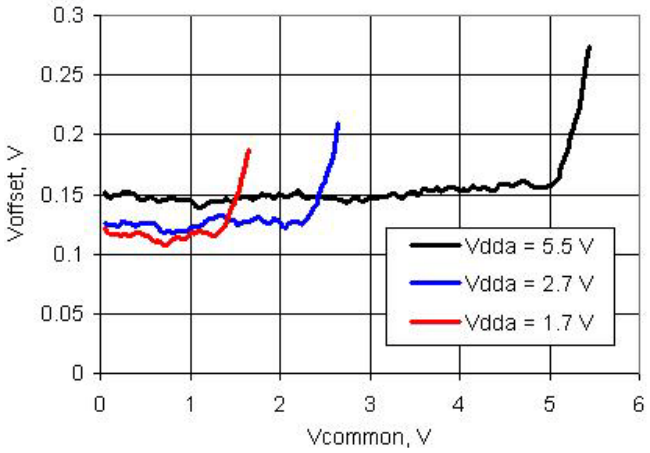



**Figure 11-28. Opamp Output Voltage vs Load Current and Temperature, High Power Mode, 25 °C, Vdda = 2.7 V**

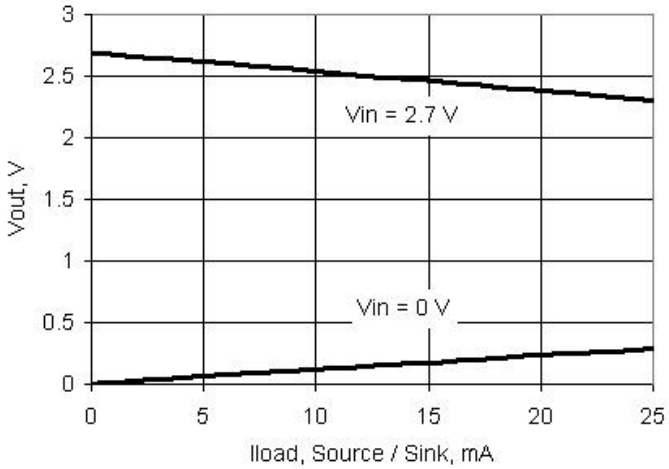



**Figure 11-29. Opamp Operating Current vs Vdda and Power Mode**

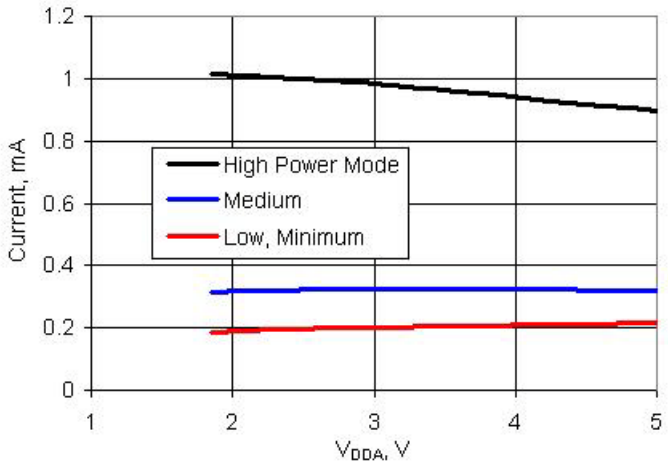



**Table 11-19. Opamp AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| GBW | Gain-bandwidth product | Power mode = minimum, 200 pF load | 1 | – | – | MHz |
| | | Power mode = low, 200 pF load | 2 | – | – | MHz |
| | | Power mode = medium, 200 pF load | 1 | – | – | MHz |
| | | Power mode = high, 200 pF load | 3 | – | – | MHz |
| SR | Slew rate, 20% - 80% | Power mode = low, 200 pF load | 1.1 | – | – | V/µs |
| | | Power mode = medium, 200 pF load | 0.9 | – | – | V/µs |
| | | Power mode = high, 200 pF load | 3 | – | – | V/µs |
| $e_n$ | Input noise density | Power mode = high, Vdda = 5 V, at 100 kHz | – | 45 | – | nV/sqrtHz |

**Figure 11-30. Opamp Noise vs Frequency, Power Mode = High, Vdda = 5V**

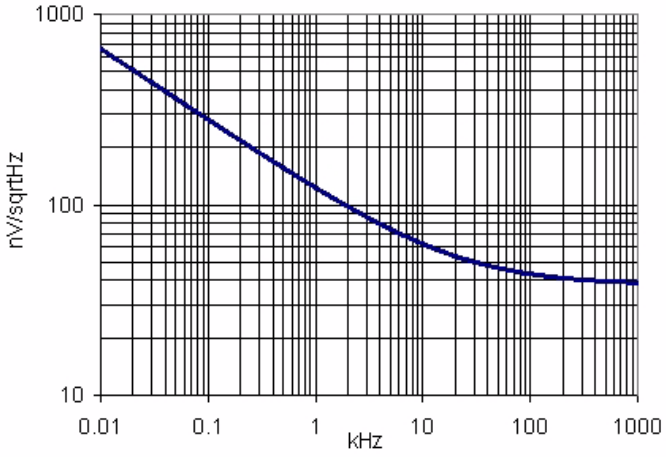



**Figure 11-31. Opamp Step Response, Rising**

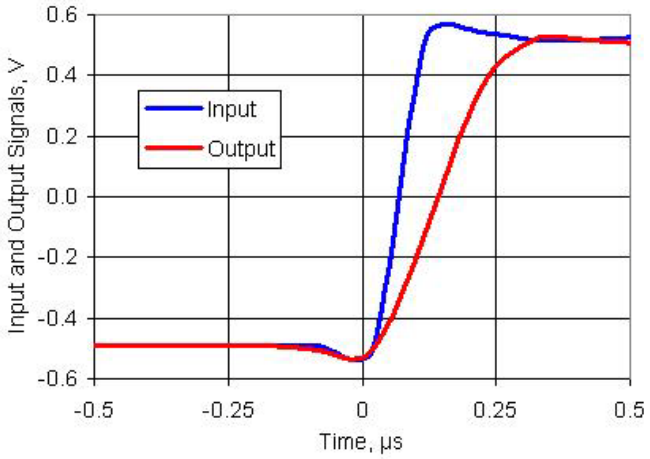
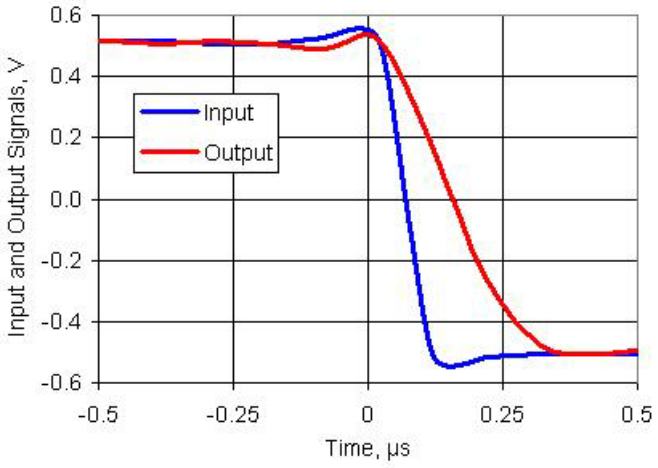



**Figure 11-32. Opamp Step Response, Falling**



### 11.5.2 Delta-sigma ADC

Unless otherwise specified, operating conditions are:

- Operation in continuous sample mode
- fclk = 6.144 MHz
- Reference = 1.024 V internal reference bypassed on P3.2 or P0.3
- Unless otherwise specified, all charts and graphs show typical values

**Table 11-20. 12-bit Delta-sigma ADC DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Resolution | | 8 | – | 12 | bits |
| | Number of channels, single ended | | – | – | No. of GPIO | – |
| | Number of channels, differential | Differential pair is formed using a pair of GPIOs. | – | – | No. of GPIO/2 | – |
| | Monotonic | Yes | – | – | – | – |
| Ge | Gain error | Buffered, buffer gain = 1, Range = ±1.024 V, 25 °C | – | – | ±0.2 | % |
| Gd | Gain drift | Buffered, buffer gain = 1, Range = ±1.024 V | – | – | 50 | ppm/°C |
| Vos | Input offset voltage | Buffered, 16-bit mode, $V_{DDA}$ = 2.7 V, 25 °C | – | – | ±0.1 | mV |
| TCVos | Temperature coefficient, input offset voltage | Buffer gain = 1, 16-bit, Range = ±1.024 V | – | – | 55 | µV/°C |
| | Input voltage range, single ended[36] | | $V_{SSA}$ | – | $V_{DDA}$ | V |
| | Input voltage range, differential unbuffered[36] | | $V_{SSA}$ | – | $V_{DDA}$ | V |
| | Input voltage range, differential, buffered[36] | | $V_{SSA}$ | – | $V_{DDA}$ – 1 | V |
| INL12 | Integral non linearity[36] | Range = ±1.024 V, unbuffered | – | – | ±1 | LSB |
| DNL12 | Differential non linearity[36] | Range = ±1.024 V, unbuffered | – | – | ±1 | LSB |
| INL8 | Integral non linearity[36] | Range = ±1.024 V, unbuffered | – | – | ±1 | LSB |
| DNL8 | Differential non linearity[36] | Range = ±1.024 V, unbuffered | – | – | ±1 | LSB |
| Rin_Buff | ADC input resistance | Input buffer used | 10 | – | – | MΩ |
| Rin_ADC12 | ADC input resistance | Input buffer bypassed, 12 bit, Range = ±1.024 V | – | 148[37] | – | kΩ |
| Vextref | ADC external reference input voltage, see also internal reference in Voltage Reference on page 87 | Pins P0[3], P3[2] | 0.9 | – | 1.3 | V |
| **Current Consumption** | | | | | | |
| $I_{DD_12}$ | Current consumption, 12 bit[36] | 192 ksps, unbuffered | – | – | 1.4 | mA |
| $I_{BUFF}$ | Buffer current consumption[36] | | – | – | 2.5 | mA |

**Notes**
36. Based on device characterization (Not production tested).
37. By using switched capacitors at the ADC input an effective input resistance is created. Holding the gain and number of bits constant, the resistance is proportional to the inverse of the clock frequency. This value is calculated, not measured. For more information see the Technical Reference Manual.

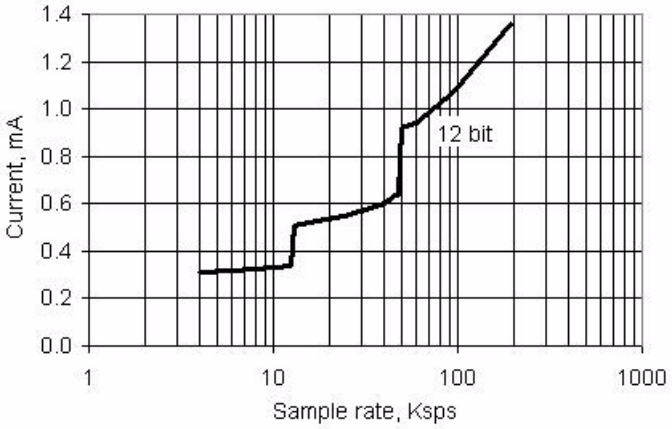
**Table 11-21. Delta-sigma ADC AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Startup time | | – | – | 4 | Samples |
| THD | Total harmonic distortion[38] | Buffer gain = 1, 16 bit, Range = ±1.024 V | – | – | 0.0032 | % |
| **12-Bit Resolution Mode** | | | | | | |
| SR12 | Sample rate, continuous, high power[38] | Range = ±1.024 V, unbuffered | 4 | – | 192 | ksps |
| BW12 | Input bandwidth at max sample rate[38] | Range = ±1.024 V, unbuffered | – | 44 | – | kHz |
| SINAD12int | Signal to noise ratio, 12-bit, internal reference[38] | Range = ±1.024 V, unbuffered | 66 | – | – | dB |
| **8-Bit Resolution Mode** | | | | | | |
| SR8 | Sample rate, continuous, high power[38] | Range = ±1.024 V, unbuffered | 8 | – | 384 | ksps |
| BW8 | Input bandwidth at max sample rate[38] | Range = ±1.024 V, unbuffered | – | 88 | – | kHz |
| SINAD8int | Signal to noise ratio, 8-bit, internal reference[38] | Range = ±1.024 V, unbuffered | 43 | – | – | dB |

**Table 11-22. Delta-sigma ADC Sample Rates, Range = ±1.024 V**

| Resolution, Bits | Continuous | | Multi-Sample | |
|---|---|---|---|---|
| | Min | Max | Min | Max |
| 8 | 8000 | 384000 | 1911 | 91701 |
| 9 | 6400 | 307200 | 1543 | 74024 |
| 10 | 5566 | 267130 | 1348 | 64673 |
| 11 | 4741 | 227555 | 1154 | 55351 |
| 12 | 4000 | 192000 | 978 | 46900 |

**Figure 11-33. Delta-sigma ADC IDD vs sps, Range = ±1.024 V, Continuous Sample Mode, Input Buffer Bypassed**




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

### 11.5.3 Voltage Reference

**Table 11-23. Voltage Reference Specifications**

See also ADC external reference specifications in *Section 11.5.2*.

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{REF}$ | Precision reference voltage | Initial trim | 1.014 (–1%) | 1.024 | 1.034 (+1%) | V |

### 11.5.4 Analog Globals

**Table 11-24. Analog Globals Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Rppag | Resistance pin-to-pin through analog global[39] | $V_{DDA}$ = 3.0 V | – | 939 | 1461 | Ω |
| Rppmuxbus | Resistance pin-to-pin through analog mux bus[39] | $V_{DDA}$ = 3.0 V | – | 721 | 1135 | Ω |

### 11.5.5 Comparator

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{OS}$ | Input offset voltage in fast mode | Factory trim, Vdda > 2.7 V, Vin ≥ 0.5 V | – | | 10 | mV |
| | Input offset voltage in slow mode | Factory trim, Vin ≥ 0.5 V | – | | 9 | mV |
| $V_{OS}$ | Input offset voltage in fast mode[40] | Custom trim | – | – | 4 | mV |
| | Input offset voltage in slow mode[40] | Custom trim | – | – | 4 | mV |
| $V_{OS}$ | Input offset voltage in ultra low-power mode | | – | ±12 | – | mV |
| $V_{HYST}$ | Hysteresis | Hysteresis enable mode | – | 10 | 32 | mV |
| $V_{ICM}$ | Input common mode voltage | High current / fast mode | $V_{SSA}$ | – | $V_{DDA}$ – 0.1 | V |
| | | Low current / slow mode | $V_{SSA}$ | – | $V_{DDA}$ | V |
| | | Ultra low power mode | $V_{SSA}$ | – | $V_{DDA}$ – 0.9 | |
| CMRR | Common mode rejection ratio | | – | 50 | – | dB |
| $I_{CMP}$ | High current mode/fast mode[41] | | – | – | 400 | µA |
| | Low current mode/slow mode[41] | | – | – | 100 | µA |
| | Ultra low-power mode[41] | | – | 6 | – | µA |

**Table 11-26. Comparator AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{RESP}$ | Response time, high current mode[41] | 50 mV overdrive, measured pin-to-pin | – | 75 | 110 | ns |
| | Response time, low current mode[41] | 50 mV overdrive, measured pin-to-pin | – | 155 | 200 | ns |
| | Response time, ultra low-power mode[41] | 50 mV overdrive, measured pin-to-pin | – | 55 | – | µs |

**Notes**
39. The resistance of the analog global and analog mux bus is high if $V_{DDA} \leq 2.7$ V, and the chip is in either sleep or hibernate mode. Use of analog global and analog mux bus under these conditions is not recommended.
40. The recommended procedure for using a custom trim value for the on-chip comparators can be found in the TRM.
41. Based on device characterization (Not production tested).

### *11.5.6 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 11-27. IDAC DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Resolution | | – | – | 8 | bits |
| $I_{OUT}$ | Output current at code = 255 | Range = 2.048 mA, code = 255, $V_{DDA} \geq 2.7$ V, Rload = 600 Ω | – | 2.048 | – | mA |
| | | Range = 2.048 mA, High mode, code = 255, $V_{DDA} \leq 2.7$ V, Rload = 300 Ω | – | 2.048 | – | mA |
| | | Range = 255 µA, code = 255, Rload = 600 Ω | – | 255 | – | µA |
| | | Range = 31.875 µA, code = 255, Rload = 600 Ω | – | 31.875 | – | µA |
| | Monotonicity | | – | – | Yes | |
| Ezs | Zero scale error | | – | 0 | ±1 | LSB |
| Eg | Gain error | Range = 2.048 mA, 25 °C | – | – | ±2.5 | % |
| | | Range = 255 µA, 25 ° C | – | – | ±2.5 | % |
| | | Range = 31.875 µA, 25 ° C | – | – | ±3.5 | % |
| TC_Eg | Temperature coefficient of gain error | Range = 2.048 mA | – | – | 0.04 | % / °C |
| | | Range = 255 µA | – | – | 0.04 | % / °C |
| | | Range = 31.875 µA | – | – | 0.05 | % / °C |
| INL | Integral nonlinearity | Sink mode, range = 255 µA, Codes 8 – 255, Rload = 2.4 kΩ, Cload = 15 pF | – | ±0.9 | ±1 | LSB |
| | | Source mode, range = 255 µA, Codes 8 – 255, Rload = 2.4 kΩ, Cload = 15 pF | – | ±1.2 | ±1.5 | LSB |
| DNL | Differential nonlinearity | Sink mode, range = 255 µA, Rload = 2.4 kΩ, Cload = 15 pF | – | ±0.3 | ±1 | LSB |
| | | Source mode, range = 255 µA, Rload = 2.4 kΩ, Cload = 15 pF | – | ±0.3 | ±1 | LSB |
| Vcompliance | Dropout voltage, source or sink mode | Voltage headroom at max current, Rload to Vdda or Rload to Vssa, Vdiff from Vdda | 1 | – | – | V |

**Table 11-27. IDAC DC Specifications** *(continued)*

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Operating current, code = 0 | Slow mode, source mode, range = 31.875 µA | – | 44 | 100 | µA |
| | | Slow mode, source mode, range = 255 µA, | – | 33 | 100 | µA |
| | | Slow mode, source mode, range = 2.04 mA | – | 33 | 100 | µA |
| | | Slow mode, sink mode, range = 31.875 µA | – | 36 | 100 | µA |
| | | Slow mode, sink mode, range = 255 µA | – | 33 | 100 | µA |
| | | Slow mode, sink mode, range = 2.04 mA | – | 33 | 100 | µA |
| | | Fast mode, source mode, range = 31.875 µA | – | 310 | 500 | µA |
| | | Fast mode, source mode, range = 255 µA | – | 305 | 500 | µA |
| | | Fast mode, source mode, range = 2.04 mA | – | 305 | 500 | µA |
| | | Fast mode, sink mode, range = 31.875 µA | – | 310 | 500 | µA |
| | | Fast mode, sink mode, range = 255 µA | – | 300 | 500 | µA |
| | | Fast mode, sink mode, range = 2.04 mA | – | 300 | 500 | µA |

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

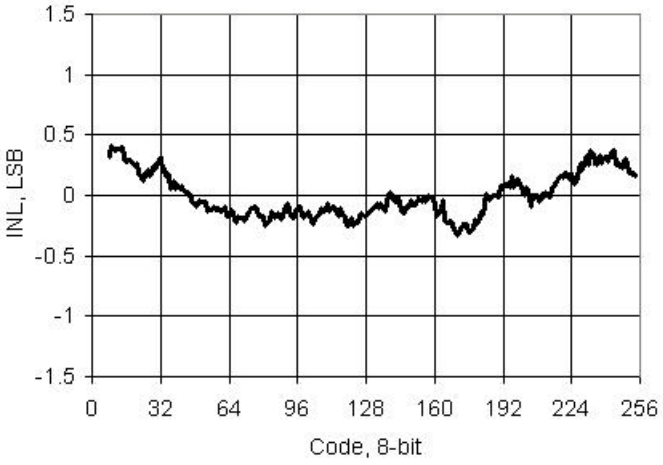



**Figure 11-35. IDAC INL vs Input Code, Range = 255 µA, Sink Mode**

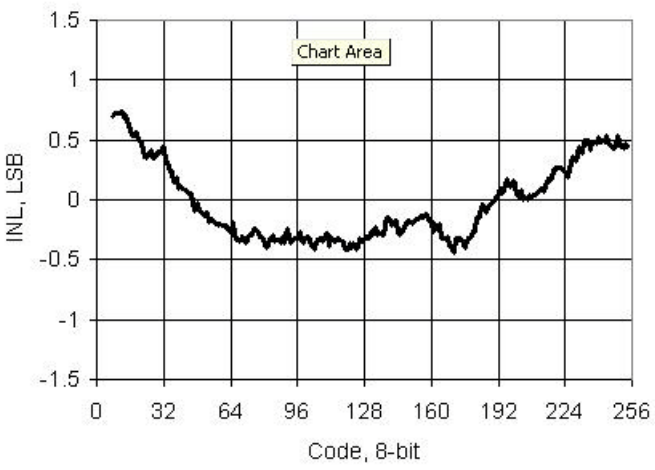


**Figure 11-36. IDAC DNL vs Input Code, Range = 255 µA, Source Mode**

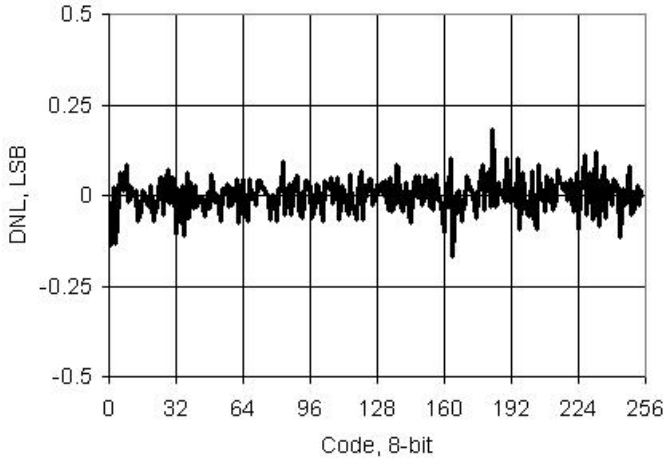



**Figure 11-37. IDAC DNL vs Input Code, Range = 255 µA, Sink Mode**

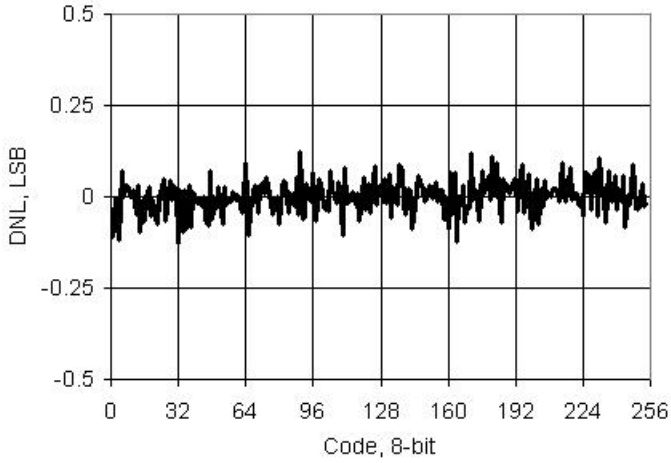



**Figure 11-38. IDAC INL vs Temperature, Range = 255 µA, Fast Mode**

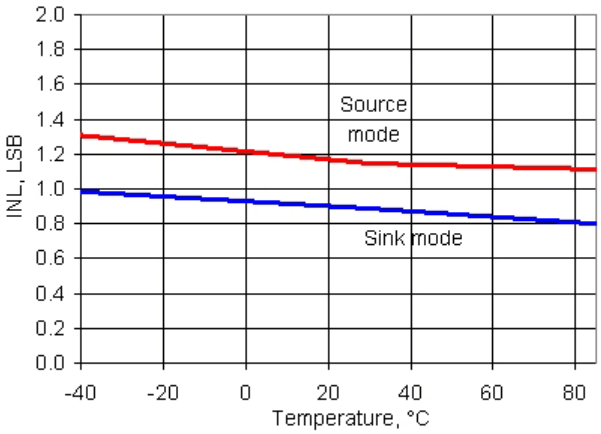



**Figure 11-39. IDAC DNL vs Temperature, Range = 255 µA, Fast Mode**

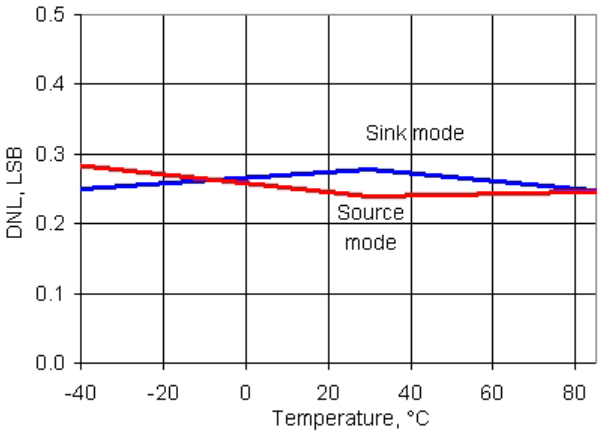


**Figure 11-40. IDAC Full Scale Error vs Temperature, Range = 255 µA, Source Mode**

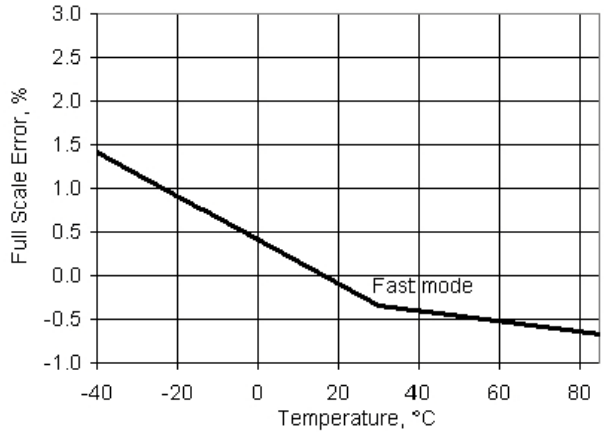



**Figure 11-41. IDAC Full Scale Error vs Temperature, Range = 255 µA, Sink Mode**

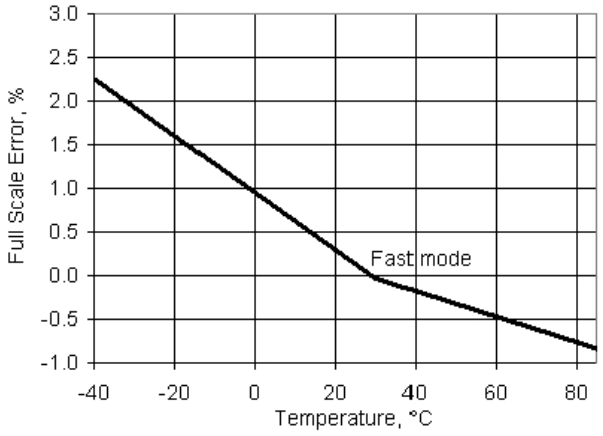



**Figure 11-42. IDAC Operating Current vs Temperature, Range = 255 µA, Code = 0, Source Mode**

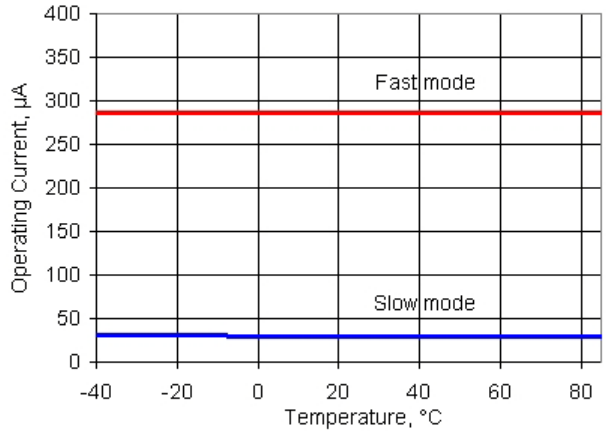



**Figure 11-43. IDAC Operating Current vs Temperature, Range = 255 µA, Code = 0, Sink Mode**

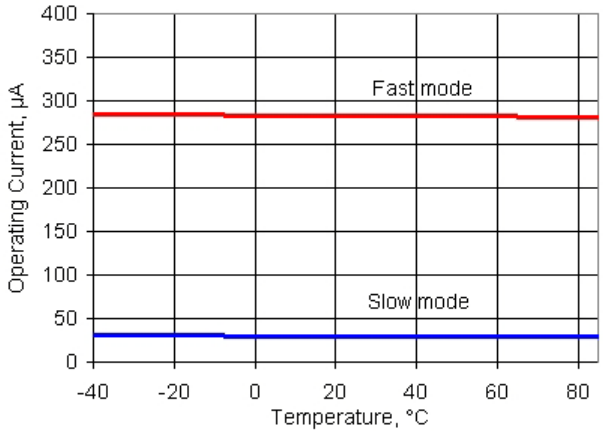


**Table 11-28. IDAC AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{DAC}$ | Update rate | | – | – | 8 | Msps |
| $T_{SETTLE}$ | Settling time to 0.5 LSB | Range = 31.875 µA or 255 µA, full scale transition, fast mode, 600 Ω 15-pF load | – | – | 125 | ns |

**Figure 11-44. IDAC Step Response, Codes 0x40 - 0xC0, 255 µA Mode, Source Mode, Fast Mode, Vdda = 5 V**

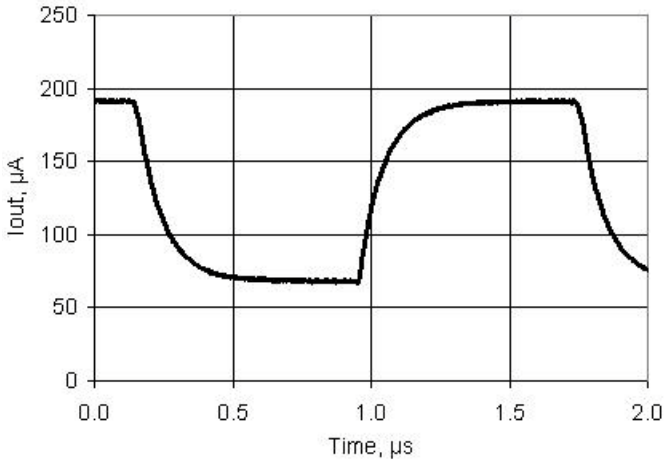



**Figure 11-45. IDAC Glitch Response, Codes 0x7F - 0x80, 255 µA Mode, Source Mode, Fast Mode, Vdda = 5 V**

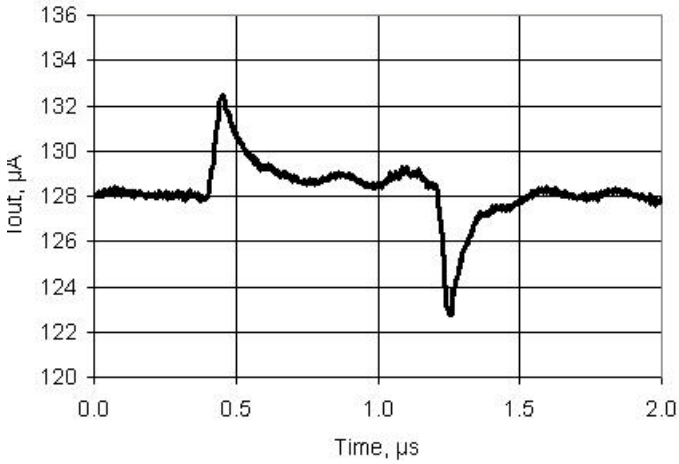
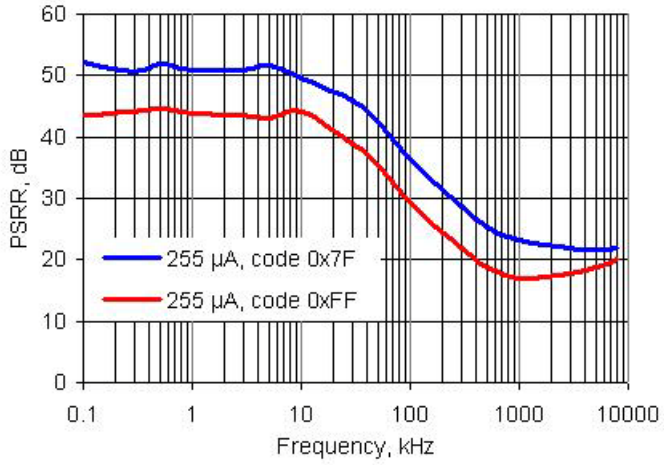



**Figure 11-46. IDAC PSRR vs Frequency**



### *11.5.7 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 11-29. 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 |
| $V_{OUT}$ | Output voltage range, code = 255 | 1 V scale | – | 1 | – | V |
| | | 4 V scale, Vdda = 5 V | – | 4 | – | V |
| | Monotonicity | | – | – | Yes | – |
| $V_{OS}$ | 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 |
| $I_{DD}$ | Operating current | Slow mode | – | – | 100 | µA |
| | | Fast mode | – | – | 500 | µA |

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

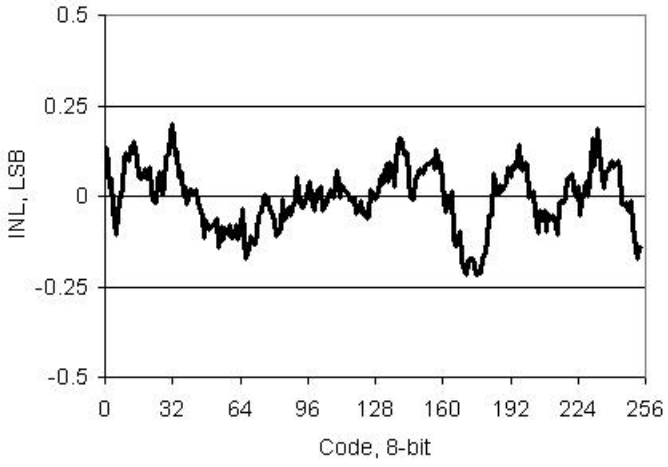



**Figure 11-48. VDAC DNL vs Input Code, 1 V Mode**

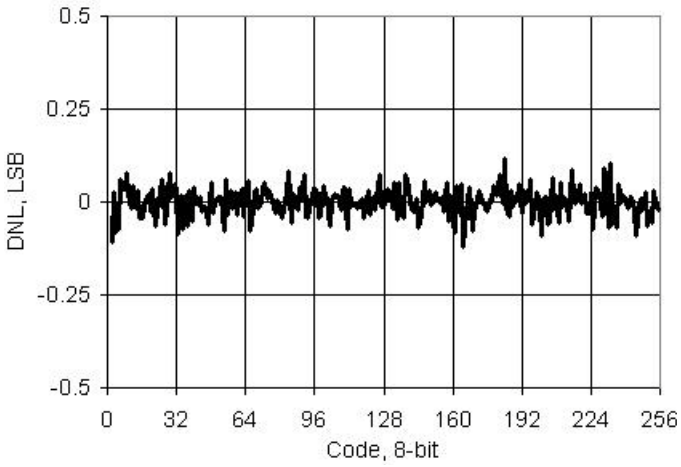


**Figure 11-49. VDAC INL vs Temperature, 1 V Mode**

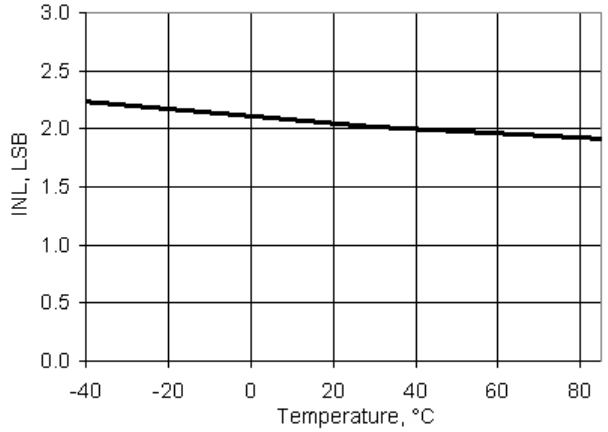



**Figure 11-50. VDAC DNL vs Temperature, 1 V Mode**

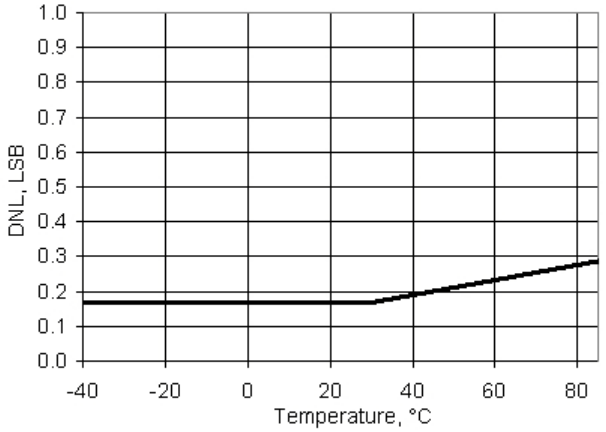



**Figure 11-51. VDAC Full Scale Error vs Temperature, 1 V Mode**

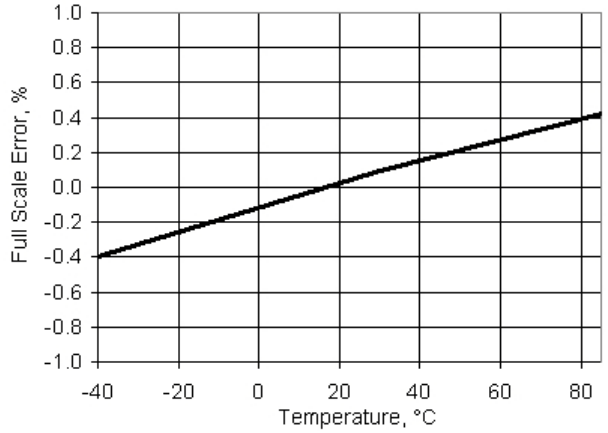



**Figure 11-52. VDAC Full Scale Error vs Temperature, 4 V Mode**

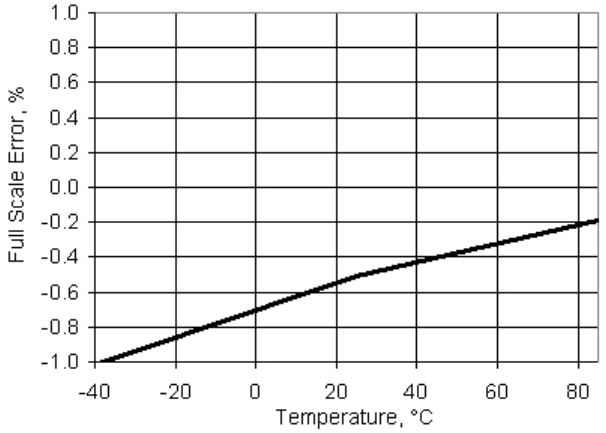



**Figure 11-53. VDAC Operating Current vs Temperature, 1V Mode, Slow Mode**

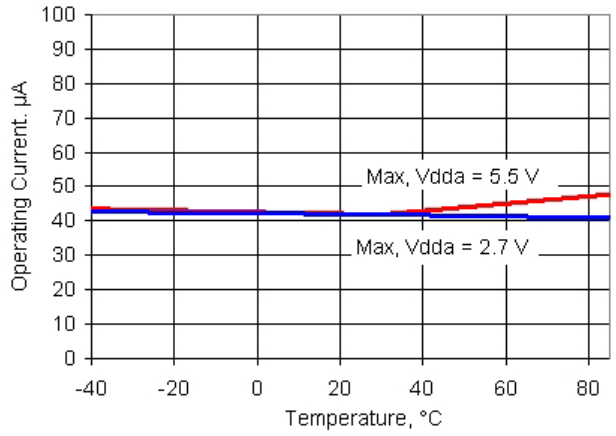



**Figure 11-54. VDAC Operating Current vs Temperature, 1 V Mode, Fast Mode**

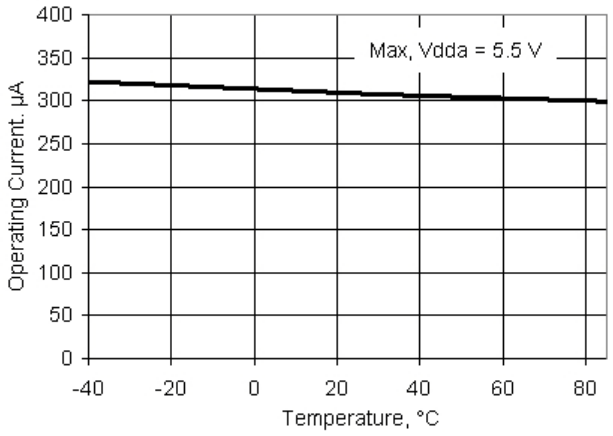


**Table 11-30. VDAC AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{DAC}$ | Update rate | 1 V scale | – | – | 1000 | ksps |
| | | 4 V scale | – | – | 250 | ksps |
| TsettleP | Settling time to 0.1%, step 25% to 75% | 1 V scale, Cload = 15 pF | – | 0.45 | 1 | µs |
| | | 4 V scale, Cload = 15 pF | – | 0.8 | 3.2 | µs |
| TsettleN | Settling time to 0.1%, step 75% to 25% | 1 V scale, Cload = 15 pF | – | 0.45 | 1 | µs |
| | | 4 V scale, Cload = 15 pF | – | 0.7 | 3 | µs |

**Figure 11-55. VDAC Step Response, Codes 0x40 - 0xC0, 1 V Mode, Fast Mode, Vdda = 5 V**

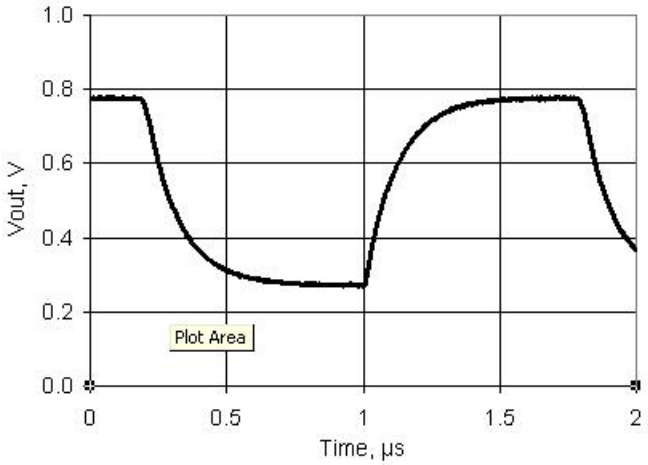



**Figure 11-56. VDAC Glitch Response, Codes 0x7F - 0x80, 1 V Mode, Fast Mode, Vdda = 5 V**

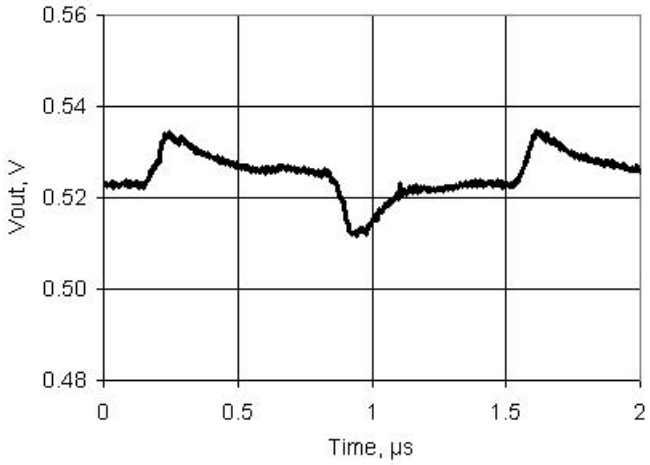
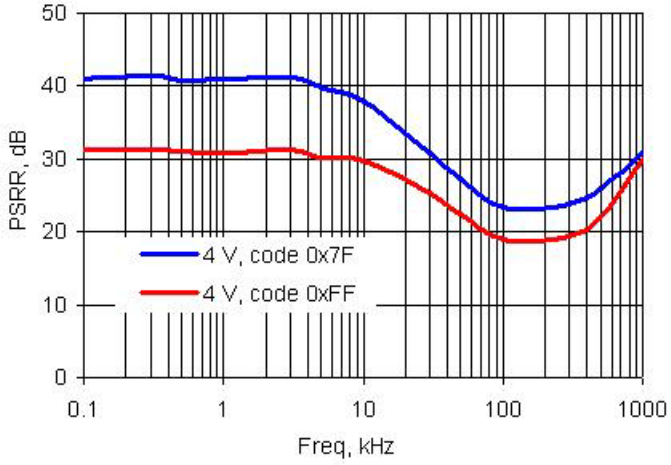



**Figure 11-57. VDAC PSRR vs Frequency**



### *11.5.8 Mixer*

The mixer is created using a SC/CT analog block; see the Mixer component data sheet in PSoC Creator for full electrical specifications and APIs.

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{OS}$ | Input offset voltage | | – | – | 10 | mV |
| | Quiescent current | | – | 0.9 | 2 | mA |
| G | Gain | | – | 0 | – | dB |

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{LO}$ | Local oscillator frequency | Down mixer mode | – | – | 4 | MHz |
| $f_{in}$ | Input signal frequency | Down mixer mode | – | – | 14 | MHz |
| $f_{LO}$ | Local oscillator frequency | Up mixer mode | – | – | 1 | MHz |
| $f_{in}$ | Input signal frequency | Up mixer mode | – | – | 1 | MHz |
| SR | Slew rate | | 3 | – | – | V/µs |

### *11.5.9 Transimpedance Amplifier*

The TIA is created using a SC/CT analog block; see the TIA component data sheet in PSoC Creator for full electrical specifications and APIs.

**Table 11-33. Transimpedance Amplifier (TIA) DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{IOFF}$ | Input offset voltage | | – | – | 10 | mV |
| Rconv | Conversion resistance[42] | R = 20K; 40 pF load | –25 | – | +35 | % |
| | | R = 30K; 40 pF load | –25 | – | +35 | % |
| | | R = 40K; 40 pF load | –25 | – | +35 | % |
| | | R = 80K; 40 pF load | –25 | – | +35 | % |
| | | R = 120K; 40 pF load | –25 | – | +35 | % |
| | | R = 250K; 40 pF load | –25 | – | +35 | % |
| | | R= 500K; 40 pF load | –25 | – | +35 | % |
| | | R = 1M; 40 pF load | –25 | – | +35 | % |
| | Quiescent current | | – | 1.1 | 2 | mA |

**Table 11-34. Transimpedance Amplifier (TIA) AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| BW | Input bandwidth (–3 dB) | R = 20K; –40 pF load | 1500 | – | – | kHz |
| | | R = 120K; –40 pF load | 240 | – | – | kHz |
| | | R = 1M; –40 pF load | 25 | – | – | kHz |

**Note**
42. Conversion resistance values are not calibrated. Calibrated values and details about calibration are provided in PSoC Creator component data sheets. External precision resistors can also be used.

### *11.5.10 Programmable Gain Amplifier*

The PGA is created using a SC/CT analog block; see the PGA component data sheet in PSoC Creator for full electrical specifications and APIs.

Unless otherwise specified, operating conditions are:

- Operating temperature = 25 °C for typical values
- Unless otherwise specified, all charts and graphs show typical values

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Vin | Input voltage range | Power mode = minimum | Vssa | – | Vdda | V |
| Vos | Input offset voltage | Power mode = high, gain = 1 | – | – | 10 | mV |
| TCVos | Input offset voltage drift with temperature | Power mode = high, gain = 1 | – | – | ±30 | µV/°C |
| Ge1 | Gain error, gain = 1 | | – | – | ±0.15 | % |
| Ge16 | Gain error, gain = 16 | | – | – | ±2.5 | % |
| Ge50 | Gain error, gain = 50 | | – | – | ±5 | % |
| Vonl | DC output nonlinearity | Gain = 1 | – | – | ±0.01 | % of FSR |
| Cin | Input capacitance | | – | – | 7 | pF |
| Voh | Output voltage swing | Power mode = high, gain = 1, Rload = 100 kΩ to $V_{DDA}$ / 2 | $V_{DDA}$ – 0.15 | – | – | V |
| Vol | Output voltage swing | Power mode = high, gain = 1, Rload = 100 kΩ to $V_{DDA}$ / 2 | – | – | $V_{SSA}$ + 0.15 | V |
| Vsrc | Output voltage under load | Iload = 250 µA, Vdda ≥ 2.7V, power mode = high | – | – | 300 | mV |
| Idd | Operating current | Power mode = high | – | 1.5 | 1.65 | mA |
| PSRR | Power supply rejection ratio | | 48 | – | – | dB |

**Figure 11-58. PGA Voffset Histogram, 4096 samples/ 1024 parts**

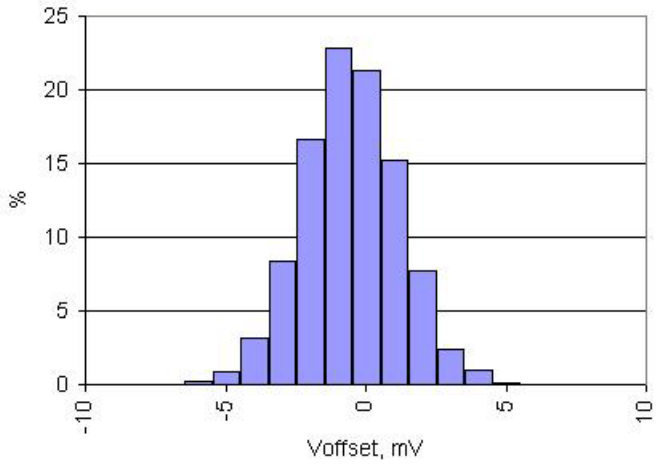


**Table 11-36. PGA AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| BW1 | –3 dB bandwidth | Power mode = high, gain = 1, input = 100 mV peak-to-peak | 6.7 | 8 | – | MHz |
| SR1 | Slew rate | Power mode = high, gain = 1, 20% to 80% | 3 | – | – | V/µs |
| $e_n$ | Input noise density | Power mode = high, Vdda = 5 V, at 100 kHz | – | 43 | – | nV/sqrtHz |

**Figure 11-59. Bandwidth vs. Temperature, at Different Gain Settings, Power Mode = High**

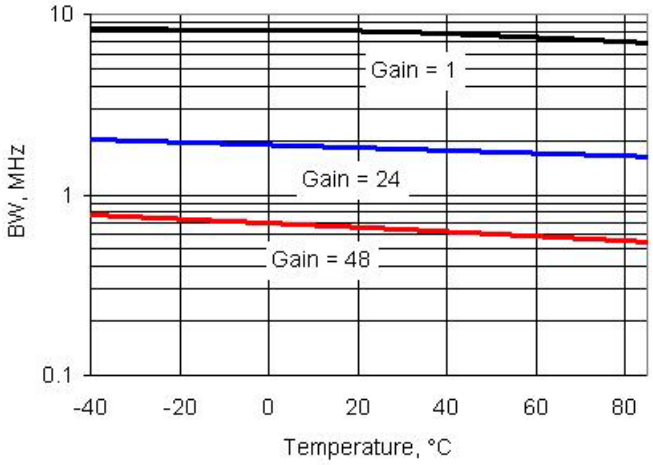
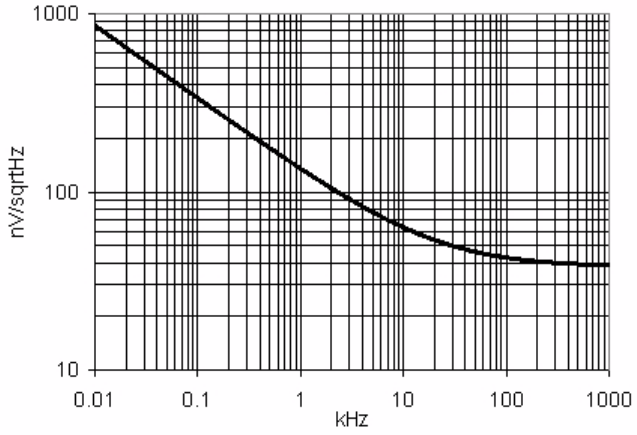



**Figure 11-60. Noise vs. Frequency, Vdda = 5 V, Power Mode = High**




*11.5.11 Temperature Sensor*

**Table 11-37. Temperature Sensor Specifications**

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

*11.5.12 LCD Direct Drive*

**Table 11-38. LCD Direct Drive DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | LCD system operating current | Device sleep mode with wakeup at 400-Hz rate to refresh LCDs, bus clock = 3 Mhz, Vddio = Vdda = 3 V, 4 commons, 16 segments, 1/4 duty cycle, 50 Hz frame rate, no glass connected | – | 38 | – | µA |
| $I_{CC_SEG}$ | Current per segment driver | Strong drive mode | – | 260 | – | µA |
| $V_{BIAS}$ | LCD bias range ($V_{BIAS}$ refers to the main output voltage(V0) of LCD DAC) | $V_{DDA} \geq 3$ V and $V_{DDA} \geq V_{BIAS}$ | 2 | – | 5 | V |
| | LCD bias step size | $V_{DDA} \geq 3$ V and $V_{DDA} \geq V_{BIAS}$ | – | 9.1 × $V_{DDA}$ | – | mV |
| | LCD capacitance per segment/common driver | Drivers may be combined | – | 500 | 5000 | pF |
| | Long term segment offset | | – | – | 20 | mV |
| $I_{OUT}$ | Output drive current per segment driver) | Vddio = 5.5V, strong drive mode | 355 | – | 710 | µA |

**Table 11-39. LCD Direct Drive AC Specifications**

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

## 11.6 Digital Peripherals

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

### *11.6.1 Timer*

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

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Block current consumption | 16-bit timer, at listed input clock frequency | – | – | – | µA |
| | 3 MHz | | – | 15 | – | µA |
| | 12 MHz | | – | 60 | – | µA |
| | 50 MHz | | – | 260 | – | µA |

**Table 11-41. Timer AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Operating frequency | | DC | – | 50.01 | MHz |
| | Capture pulse width (Internal) | | 21 | – | – | ns |
| | Capture pulse width (external) | | 42 | – | – | ns |
| | Timer resolution | | 21 | – | – | ns |
| | Enable pulse width | | 21 | – | – | ns |
| | Enable pulse width (external) | | 42 | – | – | ns |
| | Reset pulse width | | 21 | – | – | ns |
| | Reset pulse width (external) | | 42 | – | – | 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 11-42. Counter DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Block current consumption | 16–bit counter, at listed input clock frequency | – | – | – | µA |
| | 3 MHz | | – | 15 | – | µA |
| | 12 MHz | | – | 60 | – | µA |
| | 50 MHz | | – | 260 | – | µA |

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Operating frequency | | DC | – | 50.01 | MHz |
| | Capture pulse | | 21 | – | – | ns |
| | Resolution | | 21 | – | – | ns |
| | Pulse width | | 21 | – | – | ns |
| | Pulse width (external) | | 42 | | | ns |
| | Enable pulse width | | 21 | – | – | ns |
| | Enable pulse width (external) | | 42 | – | – | ns |
| | Reset pulse width | | 21 | – | – | ns |
| | Reset pulse width (external) | | 42 | – | – | 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 11-44. 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 |
| | 50 MHz | | – | 260 | – | µA |

**Table 11-45. Pulse Width Modulation (PWM) AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Operating frequency | | DC | – | 50.01 | MHz |
| | Pulse width | | 21 | – | – | ns |
| | Pulse width (external) | | 42 | – | – | ns |
| | Kill pulse width | | 21 | – | – | ns |
| | Kill pulse width (external) | | 42 | | | ns |
| | Enable pulse width | | 21 | – | – | ns |
| | Enable pulse width (external) | | 42 | – | – | ns |
| | Reset pulse width | | 21 | – | – | ns |
| | Reset pulse width (external) | | 42 | – | – | ns |

### 11.6.4 $I^2C$

**Table 11-46. Fixed $I^2C$ 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 |
| | | Wake from sleep mode | – | – | 30 | µA |

**Table 11-47. Fixed $I^2C$ AC Specifications**

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

### 11.6.5 Controller Area Network[43]

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

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Block current consumption | | – | – | 200 | µA |

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

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

**Note**
43. Refer to ISO 11898 specification for details.

### 11.6.6 USB

**Table 11-50. USB DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $V_{USB_5}$ | Device supply for USB operation | USB configured, USB regulator enabled | 4.35 | – | 5.25 | V |
| $V_{USB_3.3}$ | | USB configured, USB regulator bypassed | 3.15 | – | 3.6 | V |
| $V_{USB_3}$ | | USB configured, USB regulator bypassed[44] | 2.85 | – | 3.6 | V |
| $I_{USB_Configured}$ | Device supply current in device active mode, bus clock and IMO = 24 MHz | $V_{DDD}$ = 5 V, $F_{CPU}$ = 1.5 MHz | – | 10 | – | mA |
| | | $V_{DDD}$ = 3.3 V, $F_{CPU}$ = 1.5 MHz | – | 8 | – | mA |
| $I_{USB_Suspended}$ | Device supply current in device sleep mode | $V_{DDD}$ = 5 V, connected to USB host, PICU configured to wake on USB resume signal | – | 0.5 | – | mA |
| | | $V_{DDD}$ = 5 V, disconnected from USB host | – | 0.3 | – | mA |
| | | $V_{DDD}$ = 3.3 V, connected to USB host, PICU configured to wake on USB resume signal | – | 0.5 | – | mA |
| | | $V_{DDD}$ = 3.3 V, disconnected from USB host | – | 0.3 | – | mA |

### 11.6.7 Universal Digital Blocks (UDBs)

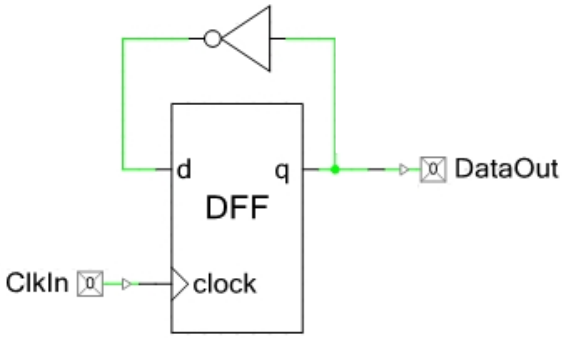
PSoC Creator provides a library of pre-built and tested standard digital peripherals (UART, SPI, LIN, PRS, CRC, timer, counter, PWM, AND, OR, and so on) that are mapped to the UDB array. See the component data sheets in PSoC Creator for full AC/DC specifications, APIs, and example code.

**Table 11-51. UDB AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Datapath Performance | | | | | | |
| $F_{MAX_TIMER}$ | Maximum frequency of 16-bit timer in a UDB pair | | – | – | 50.01 | MHz |
| $F_{MAX_ADDER}$ | Maximum frequency of 16-bit adder in a UDB pair | | – | – | 50.01 | MHz |
| $F_{MAX_CRC}$ | Maximum frequency of 16-bit CRC/PRS in a UDB pair | | – | – | 50.01 | MHz |
| PLD Performance | | | | | | |
| $F_{MAX_PLD}$ | Maximum frequency of a two-pass PLD function in a UDB pair | | – | – | 50.01 | MHz |
| Clock to Output Performance | | | | | | |
| $t_{CLK_OUT}$ | Propagation delay for clock in to data out, see Figure 11-61. | 25 °C, Vddd ≥ 2.7 V | – | 20 | 25 | ns |
| $t_{CLK_OUT}$ | Propagation delay for clock in to data out, see Figure 11-61. | Worst-case placement, routing, and pin selection | – | – | 55 | ns |

**Note**
44. Rise/fall time matching (TR) not guaranteed, see USB Driver AC Specifications on page 80.

**Figure 11-61. Clock to Output Performance**




## 11.7 Memory

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

*11.7.1 Flash*

**Table 11-52. Flash DC Specifications**

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

**Table 11-53. Flash AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{WRITE}$ | Row write time (erase + program) | | – | 15 | 20 | ms |
| $T_{ERASE}$ | Row erase time | | – | 10 | 13 | ms |
| | Row program time | | – | 5 | 7 | ms |
| $T_{BULK}$ | Bulk erase time (16 KB to 64 KB) | | – | – | 35 | ms |
| | Sector erase time (8 KB to 16 KB) | | – | – | 15 | ms |
| | Total device program time, including JTAG or SWD, and other overhead | | – | – | 5 | Seconds |
| | Flash data retention time, retention period measured from last erase cycle | Average ambient temp. $T_A$ ≤ 55 °C, 100 K erase/program cycles | 20 | – | – | years |
| | | Average ambient temp. $T_A$ ≤ 85 °C, 10 K erase/program cycles | 10 | – | – | years |

*11.7.2 EEPROM*

**Table 11-54. EEPROM DC Specifications**

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

**Table 11-55. EEPROM AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $T_{WRITE}$ | Single row erase/write cycle time | | – | 2 | 20 | ms |
| | EEPROM data retention time, retention period measured from last erase cycle | Average ambient temp, $T_A \le 25$ °C, 1M erase/program cycles | 20 | – | – | years |
| | | Average ambient temp, $T_A \le 55$ °C, 100 K erase/program cycles | 20 | – | – | |
| | | Average ambient temp. $T_A \le 85$ °C, 10 K erase/program cycles | 10 | – | – | |

*11.7.3 Nonvolatile Latches (NVL))*

**Table 11-56. NVL DC Specifications**

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

**Table 11-57. NVL AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | NVL endurance | Programmed at 25 °C | 1K | – | – | program/ erase cycles |
| | | Programmed at 0 °C to 70 °C | 100 | – | – | program/ erase cycles |
| | NVL data retention time | Programmed at 25 °C | 20 | – | – | years |
| | | Programmed at 0 °C to 70 °C | 20 | – | – | years |

*11.7.4 SRAM*

**Table 11-58. SRAM DC Specifications**

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

**Table 11-59. SRAM AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{SRAM}$ | SRAM operating frequency | | DC | – | 50.01 | MHz |

### 11.7.5 External Memory Interface

**Figure 11-62. Asynchronous Read Cycle Timing**

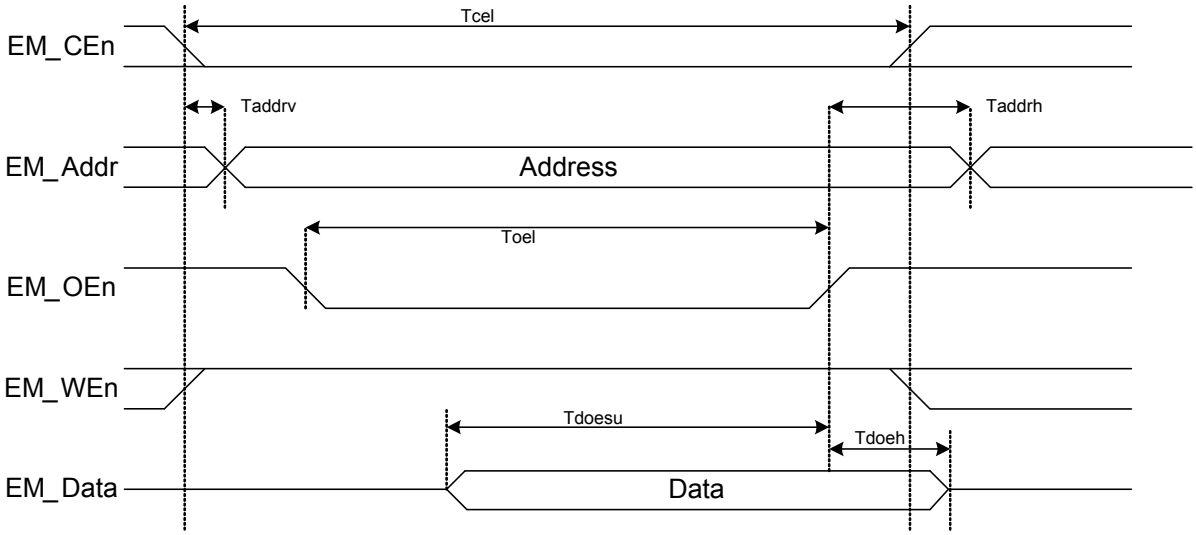



**Table 11-60. Asynchronous Read Cycle Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| T | EMIF clock period[45] | Vdda ≥ 3.3 V | 30.3 | – | – | nS |
| Tcel | EM_CEn low time | | 2T – 5 | – | 2T+ 5 | nS |
| Taddrv | EM_CEn low to EM_Addr valid | | – | – | 5 | nS |
| Taddrh | Address hold time after EM_Wen high | | T | – | – | nS |
| Toel | EM_OEn low time | | 2T – 5 | – | 2T + 5 | nS |
| Tdoesu | Data to EM_OEn high setup time | | T + 15 | – | – | nS |
| Tdoeh | Data hold time after EM_OEn high | | 3 | – | – | nS |

**Note**
45. Limited by GPIO output frequency, see Table 11-10 on page 75.

**Figure 11-63. Asynchronous Write Cycle Timing**

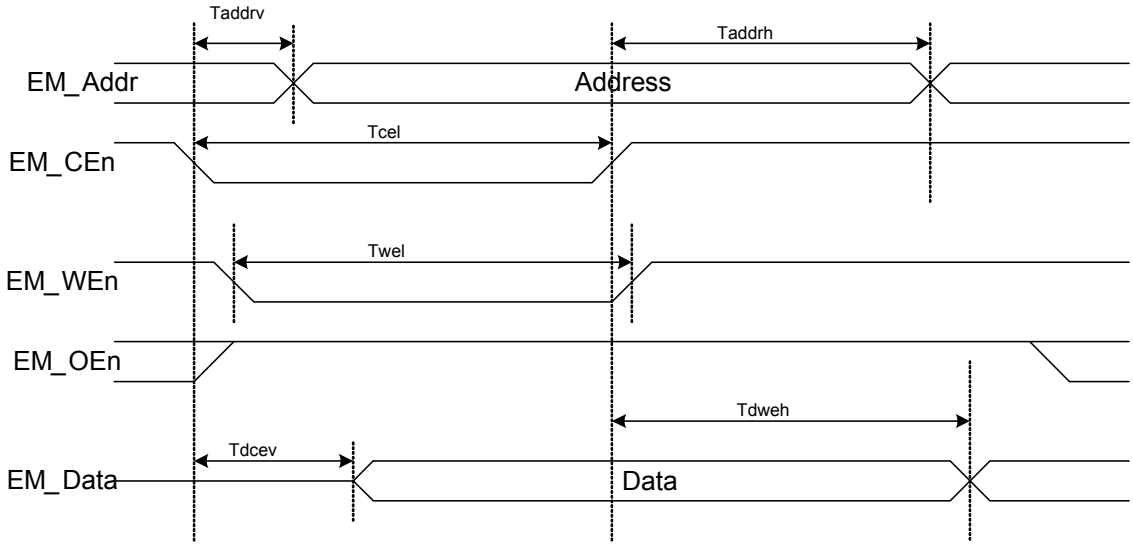



**Table 11-61. Asynchronous Write Cycle Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| T | EMIF clock period[46] | Vdda ≥ 3.3 V | 30.3 | – | – | nS |
| Tcel | EM_CEn low time | | T – 5 | – | T + 5 | nS |
| Taddrv | EM_CEn low to EM_Addr valid | | – | – | 5 | nS |
| Taddrh | Address hold time after EM_WEn high | | T | – | – | nS |
| Twel | EM_WEn low time | | T – 5 | – | T + 5 | nS |
| Tdcev | EM_CEn low to data valid | | – | – | 7 | nS |
| Tdweh | Data hold time after EM_WEn high | | T | – | – | nS |

**Note**
46. Limited by GPIO output frequency, see Table 11-10 on page 75.

**Figure 11-64. Synchronous Read Cycle Timing**

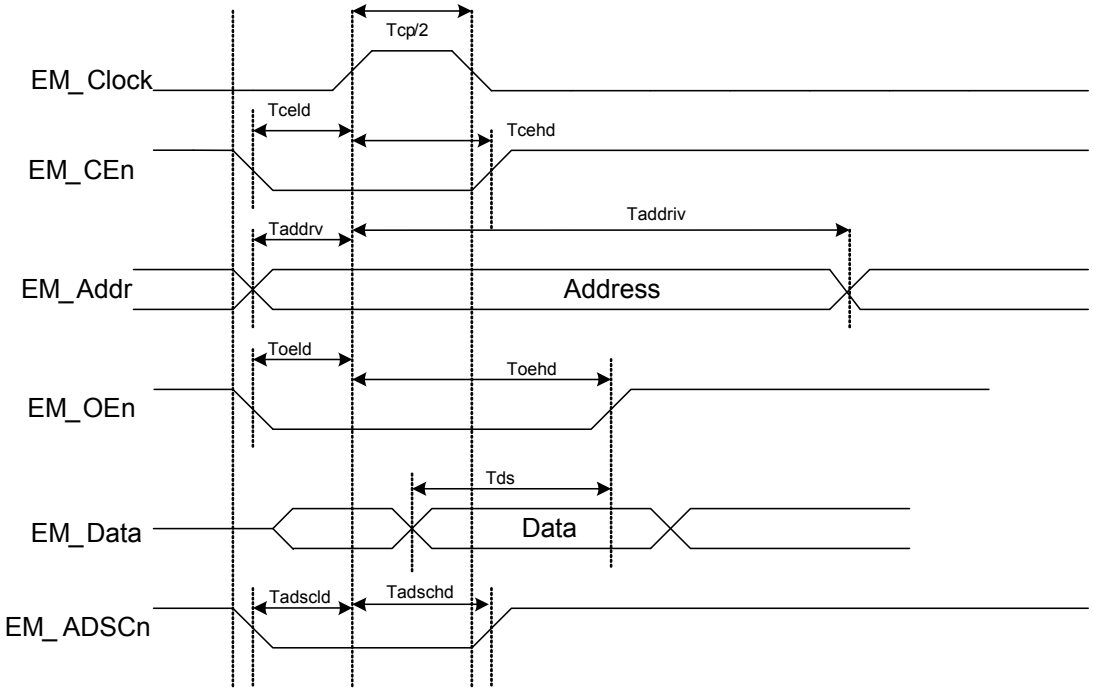



**Table 11-62. Synchronous Read Cycle Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| T | EMIF clock period[47] | Vdda ≥ 3.3 V | 30.3 | – | – | nS |
| Tcp/2 | EM_Clock pulse high | | T/2 | – | – | nS |
| Tceld | EM_CEn low to EM_Clock high | | 5 | – | – | nS |
| Tcehd | EM_Clock high to EM_CEn high | | T/2 – 5 | – | – | nS |
| Taddrv | EM_Addr valid to EM_Clock high | | 5 | – | – | nS |
| Taddriv | EM_Clock high to EM_Addr invalid | | T/2 – 5 | – | – | nS |
| Toeld | EM_OEn low to EM_Clock high | | 5 | – | – | nS |
| Toehd | EM_Clock high to EM_OEn high | | T | – | – | nS |
| Tds | Data valid before EM_OEn high | | T + 15 | – | – | nS |
| Tadscld | EM_ADSCn low to EM_Clock high | | 5 | – | – | nS |
| Tadschd | EM_Clock high to EM_ADSCn high | | T/2 – 5 | – | – | nS |

**Note**
47. Limited by GPIO output frequency, see Table 11-10 on page 75.

**Figure 11-65. Synchronous Write Cycle Timing**

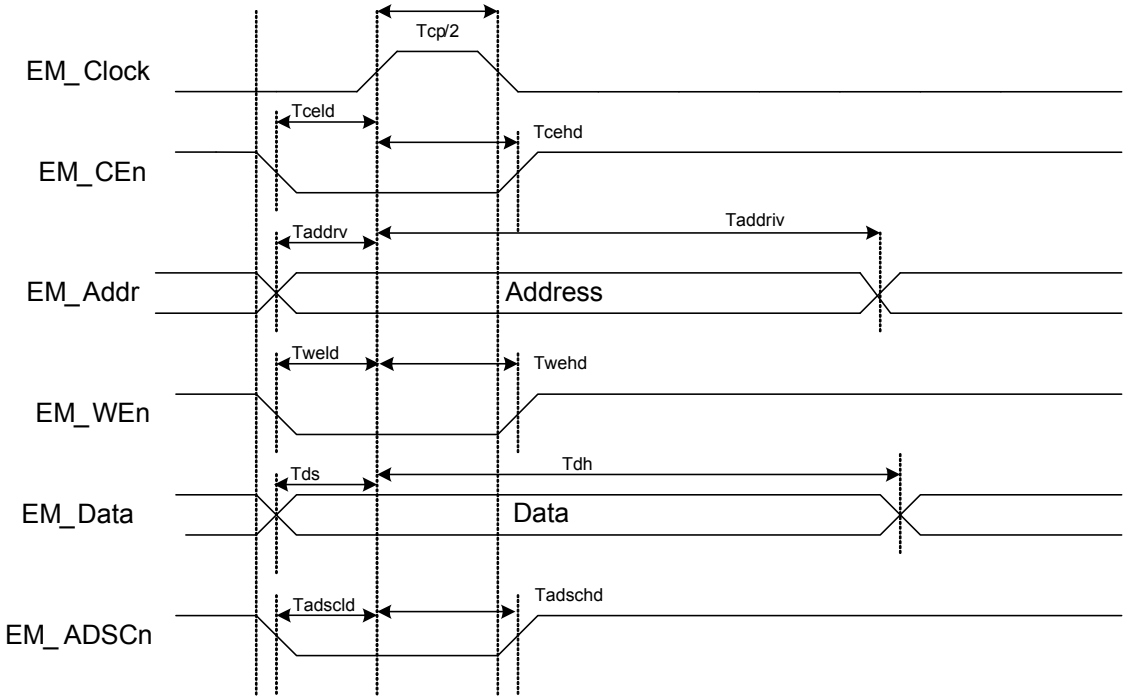



**Table 11-63. Synchronous Write Cycle Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| T | EMIF clock Period[48] | Vdda ≥ 3.3 V | 30.3 | – | – | nS |
| Tcp/2 | EM_Clock pulse high | | T/2 | – | – | nS |
| Tceld | EM_CEn low to EM_Clock high | | 5 | – | – | nS |
| Tcehd | EM_Clock high to EM_CEn high | | T/2 – 5 | – | – | nS |
| Taddrv | EM_Addr valid to EM_Clock high | | 5 | – | – | nS |
| Taddriv | EM_Clock high to EM_Addr invalid | | T/2 – 5 | – | – | nS |
| Tweld | EM_WEn low to EM_Clock high | | 5 | – | – | nS |
| Twehd | EM_Clock high to EM_WEn high | | T/2 – 5 | – | – | nS |
| Tds | Data valid before EM_Clock high | | 5 | – | – | nS |
| Tdh | Data invalid after EM_Clock high | | T | – | – | nS |
| Tadscld | EM_ADSCn low to EM_Clock high | | 5 | – | – | nS |
| Tadschd | EM_Clock high to EM_ADSCn high | | T/2 – 5 | – | – | nS |

**Note**
48. Limited by GPIO output frequency, see Table 11-10 on page 75.

## 11.8 PSoC System Resources

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

### *11.8.1 POR with Brown Out*

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

**Table 11-64. Precise power-on Reset (PRES) with Brown Out DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | **Precise POR (PPOR)** | | | | | |
| PRESR | Rising trip voltage | Factory trim | 1.64 | – | 1.68 | V |
| PRESF | Falling trip voltage | | 1.62 | – | 1.66 | V |

**Table 11-65. Power-on Reset (POR) with Brown Out AC Specifications**

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

### *11.8.2 Voltage Monitors*

**Table 11-66. Voltage Monitors DC Specifications**

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

**Table 11-67. Voltage Monitors AC Specifications**

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

### 11.8.3 Interrupt Controller

**Table 11-68. Interrupt Controller AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Delay from interrupt signal input to ISR code execution from ISR code | Includes worse case completion of longest instruction DIV with 6 cycles | – | – | 25 | Tcy CPU |

### 11.8.4 JTAG Interface

**Figure 11-66. JTAG Interface Timing**

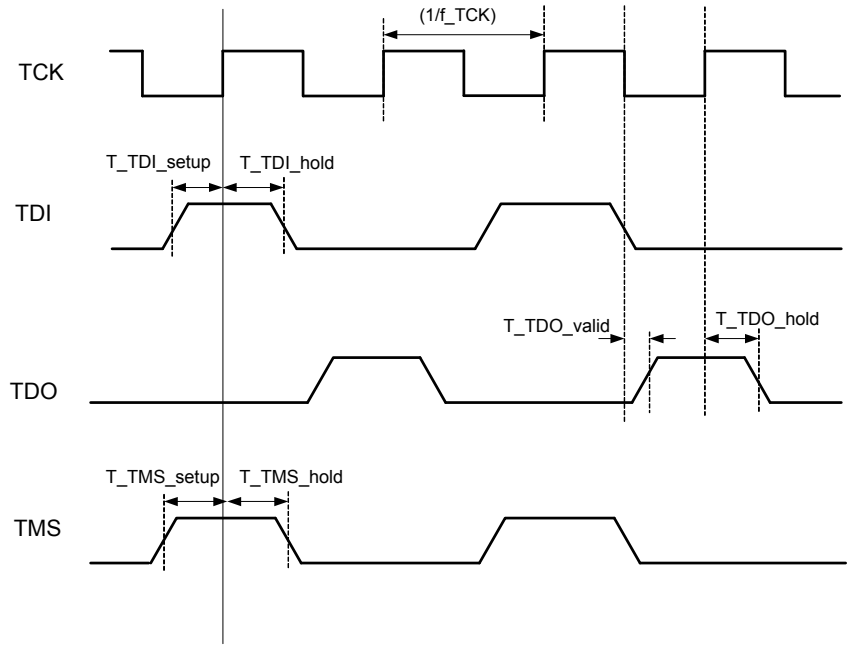



**Table 11-69. JTAG Interface AC Specifications**[49]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| f_TCK | TCK frequency | 3.3 V ≤ $V_{DDD}$ ≤ 5 V | – | – | 14[50] | MHz |
| | | 1.71 V ≤ $V_{DDD}$ < 3.3 V | – | – | 7[50] | MHz |
| T_TDI_setup | TDI setup before TCK high | | (T/10) – 5 | – | – | ns |
| T_TMS_setup | TMS setup before TCK high | | T/4 | – | – | |
| T_TDI_hold | TDI, TMS hold after TCK high | T = 1/f_TCK max | T/4 | – | – | |
| T_TDO_valid | TCK low to TDO valid | T = 1/f_TCK max | – | – | 2T/5 | |
| T_TDO_hold | TDO hold after TCK high | T = 1/f_TCK max | T/4 | – | – | |

**Notes**
49. Based on device characterization (Not production tested).
50. f_TCK must also be no more than 1/3 CPU clock frequency.

### 11.8.5 SWD Interface

**Figure 11-67. SWD Interface Timing**

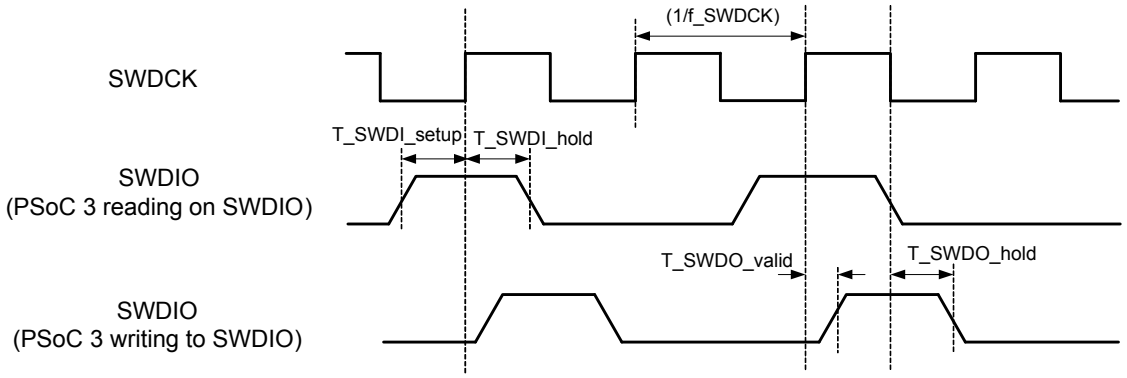



**Table 11-70. SWD Interface AC Specifications**[51]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| f_SWDCK | SWDCLK frequency | 3.3 V ≤ $V_{DDD}$ ≤ 5 V | – | – | 14[52] | MHz |
| | | 1.71 V ≤ $V_{DDD}$ < 3.3 V | – | – | 7[52] | MHz |
| | | 1.71 V ≤ $V_{DDD}$ < 3.3 V, SWD over USBIO pins | – | – | 5.5[52] | MHz |
| T_SWDI_setup | SWDIO input setup before SWDCK high | T = 1/f_SWDCK max | T/4 | – | – | – |
| T_SWDI_hold | SWDIO input hold after SWDCK high | T = 1/f_SWDCK max | T/4 | – | – | – |
| T_SWDO_valid | SWDCK high to SWDIO output | T = 1/f_SWDCK max | – | – | 2T/5 | – |
| T_SWDO_hold | SWDIO output hold after SWDCK low | T = 1/f_SWDCK max | T/4 | – | – | – |

### 11.8.6 SWV Interface

**Table 11-71. SWV Interface AC Specifications**[51]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | SWV mode SWV bit rate | | – | – | 33 | Mbit |

**Notes**
51. Based on device characterization (Not production tested).
52. ff_SWDCK must also be no more than 1/3 CPU clock frequency.

## 11.9 Clocking

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

### *11.9.1 32 kHz External Crystal*

**Table 11-72. 32 kHz External Crystal DC Specifications**[53]

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Operating current | Low-power mode | – | 0.25 | 1.0 | µA |
| CL | External crystal capacitance | | – | 6 | – | pF |
| DL | Drive level | | – | – | 1 | µW |

**Table 11-73. 32 kHz External Crystal AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| F | Frequency | | – | 32.768 | – | kHz |
| $T_{ON}$ | Startup time | High-power mode | – | 1 | – | s |

### *11.9.2 Internal Main Oscillator*

**Table 11-74. IMO DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Supply current | | | | | |
| | 24 MHz – USB mode | With oscillator locking to USB bus | – | – | 500 | µA |
| | 24 MHz – non USB mode | | – | – | 300 | µA |
| | 12 MHz | | – | – | 200 | µA |
| | 6 MHz | | – | – | 180 | µA |
| | 3 MHz | | – | – | 150 | µA |

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

**Figure 11-68. IMO Current vs. Frequency**

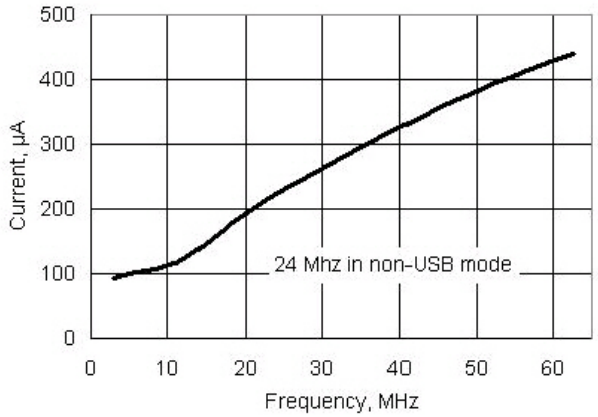



**Table 11-75. IMO AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $F_{IMO}$ | IMO frequency stability (with factory trim) | | | | | |
| | 24 MHz – Non USB mode | | –4 | – | 4 | % |
| | 24 MHz – USB mode | With oscillator locking to USB bus | –0.25 | – | 0.25 | % |
| | 12 MHz | | –3 | – | 3 | % |
| | 6 MHz | | –2 | – | 2 | % |
| | 3 MHz | | –1 | – | 1 | % |
| | Startup time[54] | From enable (during normal system operation) or wakeup from low-power state | – | – | 12 | µs |
| Jp-p | Jitter (peak to peak)[54] | | | | | |
| | F = 24 MHz | | – | 0.9 | – | ns |
| | F = 3 MHz | | – | 1.6 | – | ns |
| Jperiod | Jitter (long term)[54] | | | | | |
| | F = 24 MHz | | – | 0.9 | – | ns |
| | F = 3 MHz | | – | 12 | – | ns |

**Figure 11-69. IMO Frequency Variation vs. Temperature**

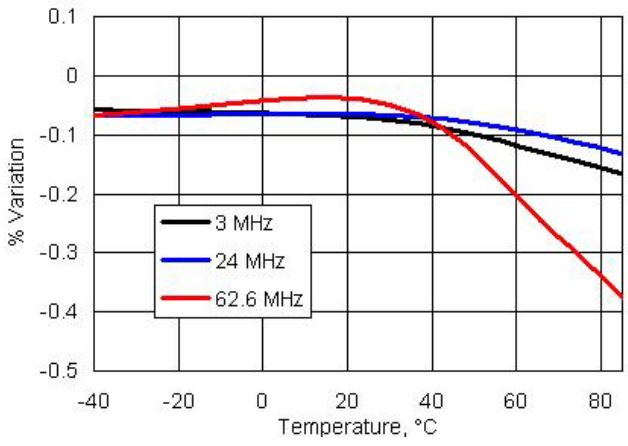
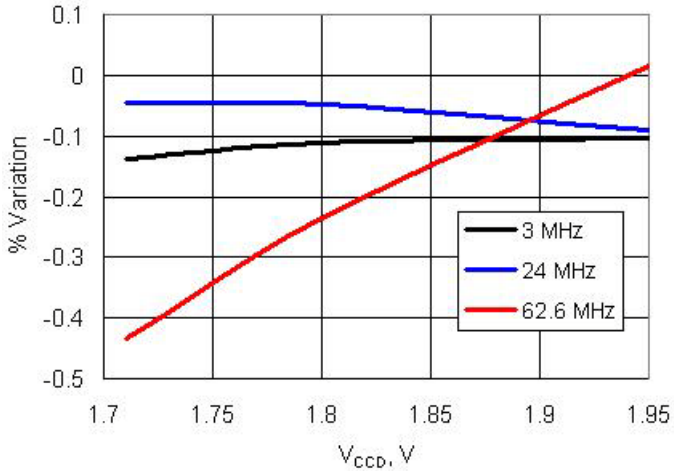



**Figure 11-70. IMO Frequency Variation vs. $V_{CC}$**




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

### 11.9.3 Internal Low-Speed Oscillator

**Table 11-76. ILO DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{CC}$ | Operating current | $F_{OUT}$ = 1 kHz | – | 0.3 | 1.7 | µA |
| | | $F_{OUT}$ = 33 kHz | – | 1.0 | 2.6 | µA |
| | | $F_{OUT}$ = 100 kHz | – | 1.0 | 2.6 | µA |
| | Leakage current | Power down mode | – | 2.0 | 15 | nA |

**Table 11-77. ILO AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | Startup time, all frequencies | Turbo mode | – | – | 2 | ms |
| $F_{ILO}$ | 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 11-71. ILO Frequency Variation vs. Temperature**

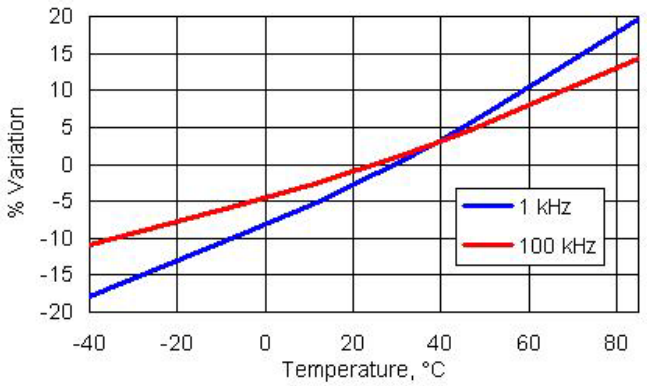
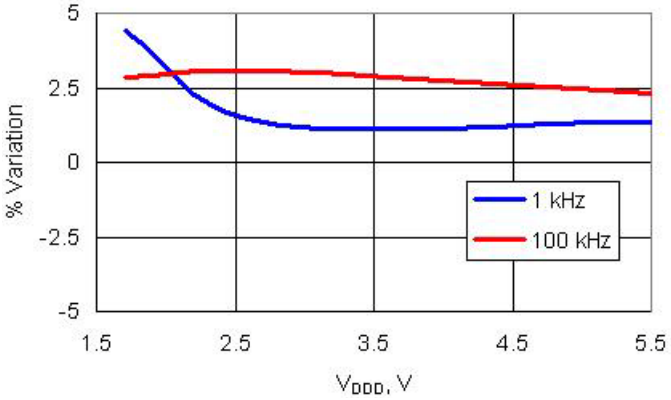



**Figure 11-72. ILO Frequency Variation vs. $V_{DD}$**



### 11.9.4 External Crystal Oscillator

**Table 11-78. ECO AC Specifications**

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

### 11.9.5 External Clock Reference

**Table 11-79. External Clock Reference AC Specifications[55]**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| | External frequency range | | 0 | – | 33 | MHz |
| | Input duty cycle range | Measured at $V_{DDIO}/2$ | 30 | 50 | 70 | % |
| | Input edge rate | $V_{IL}$ to $V_{IH}$ | 0.1 | – | – | V/ns |

### 11.9.6 Phase-Locked Loop

**Table 11-80. PLL DC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | PLL operating current | In = 3 MHz, Out = 24 MHz | – | 200 | – | µA |

**Table 11-81. PLL AC Specifications**

| Parameter | Description | Conditions | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| Fpllin | PLL input frequency[56] | | 1 | – | 48 | MHz |
| | PLL intermediate frequency[57] | Output of prescaler | 1 | – | 3 | MHz |
| Fpllout | PLL output frequency[56] | | 24 | – | 50 | MHz |
| | Lock time at startup | | – | – | 250 | µs |
| Jperiod-rms | Jitter (rms)[55] | | – | – | 250 | ps |

**Notes**
55. Based on device characterization (Not production tested).
56. This specification is guaranteed by testing the PLL across the specified range using the IMO as the source for the PLL.
57. PLL input divider, Q, must be set so that the input frequency is divided down to the intermediate frequency range. Value for Q ranges from 1 to 16.

## 12. Ordering Information

In addition to the features listed in Table 12-1, every CY8C34 device includes: a precision on-chip voltage reference, precision oscillators, flash, ECC, DMA, a fixed function $I^2C$, 4 KB trace RAM, JTAG/SWD programming and debug, external memory interface, 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 CY8C34 derivatives incorporate device and flash security in user-selectable security levels; see the TRM for details.

**Table 12-1. CY8C34 Family with Single Cycle 8051**

| Part Number | MCU Core | | | | Analog | | | | | | | | Digital | | | | I/O[60] | | | | Package | JTAG ID[61] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CPU Speed (MHz) | Flash (KB) | SRAM (KB) | EEPROM (KB) | LCD Segment Drive | ADC | DAC | Comparator | SC/CT Analog Blocks[58] | Opamps | DFB | CapSense | UDBs[59] | 16-bit Timer/PWM | FS USB | CAN 2.0b | Total I/O | GPIO | SIO | USBIO | | |
| **16 KB Flash** | | | | | | | | | | | | | | | | | | | | | | |
| CY8C3444LTI-110 | 50 | 16 | 2 | 0.5 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 16 | 4 | – | – | 46 | 38 | 8 | 0 | 68-pin QFN | 0×0E06E069 |
| CY8C3444AXI-116 | 50 | 16 | 2 | 0.5 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 16 | 4 | – | – | 70 | 62 | 8 | 0 | 100-pin TQFP | 0×0E074069 |
| CY8C3444LTI-119 | 50 | 16 | 2 | 0.5 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 16 | 4 | – | – | 29 | 25 | 4 | 0 | 48-pin QFN | 0×0E077069 |
| CY8C3444PVI-100 | 50 | 16 | 2 | 0.5 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 16 | 4 | – | – | 29 | 25 | 4 | 0 | 48-pin SSOP | 0×0E064069 |
| **32 KB Flash** | | | | | | | | | | | | | | | | | | | | | | |
| CY8C3445AXI-104 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | – | – | 70 | 62 | 8 | 0 | 100-pin TQFP | 0×0E068069 |
| CY8C3445LTI-079 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | – | – | 46 | 38 | 8 | 0 | 68-pin QFN | 0×0E04F069 |
| CY8C3445LTI-078 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | – | – | 29 | 25 | 4 | 0 | 48-pin QFN | 0×0E04E069 |
| CY8C3445PVI-094 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | – | – | 29 | 25 | 4 | 0 | 48-pin SSOP | 0×1E05E069 |
| CY8C3445AXI-108 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | ✔ | – | 72 | 62 | 8 | 2 | 100-pin TQFP | 0×1E06C069 |
| CY8C3445LTI-081 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | ✔ | – | 48 | 38 | 8 | 2 | 68-pin QFN | 0×1E051069 |
| CY8C3445LTI-089 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | ✔ | – | 31 | 25 | 4 | 2 | 48-pin QFN | 0×1E059069 |
| CY8C3445PVI-090 | 50 | 32 | 4 | 1 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 20 | 4 | ✔ | – | 31 | 25 | 4 | 2 | 48-pin SSOP | 0×1E05A069 |
| **64 KB Flash** | | | | | | | | | | | | | | | | | | | | | | |
| CY8C3446LTI-073 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | ✔ | – | 31 | 25 | 4 | 2 | 48-pin QFN | 0×1E049069 |
| CY8C3446AXI-105 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | – | – | 70 | 62 | 8 | 0 | 100-pin TQFP | 0×1E069069 |
| CY8C3446LTI-074 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | – | – | 46 | 38 | 8 | 0 | 68-pin QFN | 0×1E04A069 |
| CY8C3446LTI-083 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | – | – | 29 | 25 | 4 | 0 | 48-pin QFN | 0×1E053069 |
| CY8C3446PVI-091 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | – | – | 29 | 25 | 4 | 0 | 48-pin SSOP | 0×1E05B069 |
| CY8C3446AXI-099 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | ✔ | – | 72 | 62 | 8 | 2 | 100-pin TQFP | 0×1E063069 |
| CY8C3446LTI-085 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | ✔ | – | 48 | 38 | 8 | 2 | 68-pin QFN | 0×1E055069 |
| CY8C3446PVI-076 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | ✔ | – | 31 | 25 | 4 | 2 | 48-pin SSOP | 0×1E04C069 |
| CY8C3446PVI-102 | 50 | 64 | 8 | 2 | ✔ | 12-bit Del-Sig | 2 | 4 | 2 | 2 | – | ✔ | 24 | 4 | – | ✔ | 29 | 25 | 4 | 0 | 48-pin SSOP | 0×1E066069 |

**Notes**

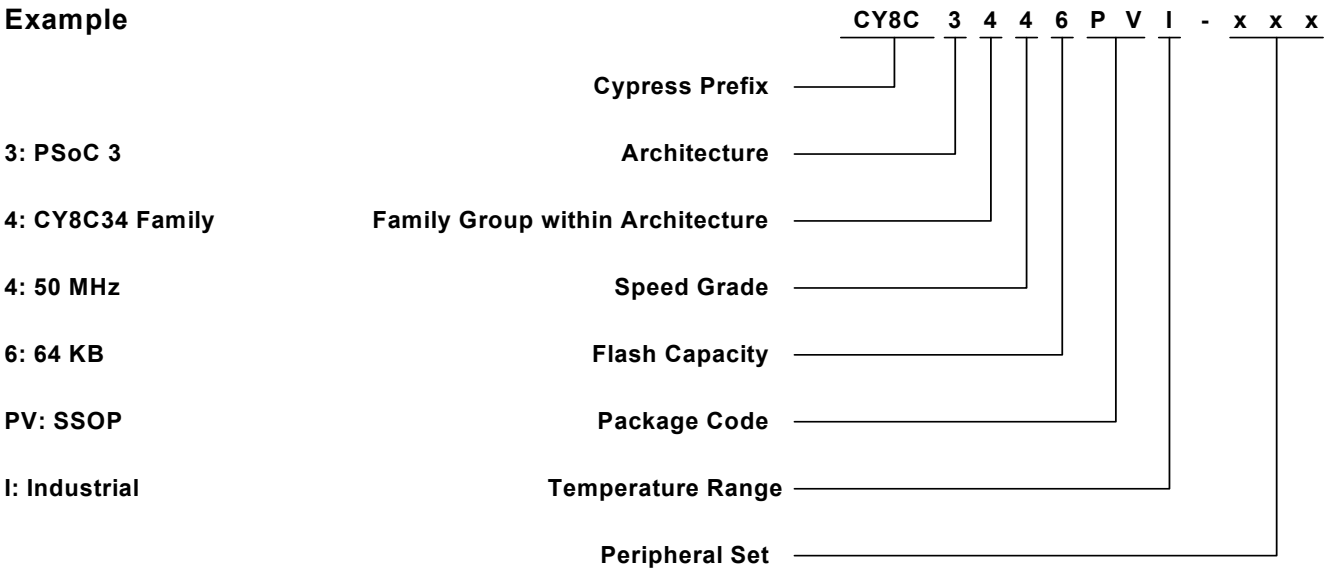
58. Analog blocks support a wide variety of functionality including TIA, PGA, and mixers. See the Example Peripherals on page 39 for more information on how analog blocks can be used.
59. UDBs support a wide variety of functionality including SPI, LIN, UART, timer, counter, PWM, PRS, and others. Individual functions may use a fraction of a UDB or multiple UDBs. Multiple functions can share a single UDB. See the Example Peripherals on page 39 for more information on how UDBs can be used.
60. The I/O Count includes all types of digital I/O: GPIO, SIO, and the two USB I/O. See the I/O System and Routing on page 32 for details on the functionality of each of these types of I/O.
61. The JTAG ID has three major fields. The most significant nibble (left digit) is the version, followed by a 2 byte part number and a 3 nibble manufacturer ID.

### 12.1 Part Numbering Conventions

PSoC 3 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
  - 4: CY8C34 family
  - 6: CY8C36 family
  - 8: CY8C38 family
- c: Speed grade
  - 4: 50 MHz
  - 6: 67 MHz
- d: Flash capacity
  - 4: 16 KB
  - 5: 32 KB
  - 6: 64 KB
- ef: Package code
  - Two character alphanumeric
  - AX: TQFP
  - LT: QFN
  - PV: SSOP
- g: Temperature range
  - C: commercial
  - I: industrial
  - A: automotive
- xxx: Peripheral set
  - Three character numeric
  - No meaning is associated with these three characters.




All devices in the PSoC 3 CY8C34 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 |
|---|---|---|---|---|---|---|
| $T_A$ | Operating ambient temperature | | –40 | 25.00 | 85 | °C |
| $T_J$ | Operating junction temperature | | –40 | – | 100 | °C |
| $T_{JA}$ | Package $\theta_{JA}$ (48-pin SSOP) | | – | 49 | – | °C/Watt |
| $T_{JA}$ | Package $\theta_{JA}$ (48-pin QFN) | | – | 14 | – | °C/Watt |
| $T_{JA}$ | Package $\theta_{JA}$ (68-pin QFN) | | – | 15 | – | °C/Watt |
| $T_{JA}$ | Package $\theta_{JA}$ (100-pin TQFP) | | – | 34 | – | °C/Watt |
| $T_{JC}$ | Package $\theta_{JC}$ (48-pin SSOP) | | – | 24 | – | °C/Watt |
| $T_{JC}$ | Package $\theta_{JC}$ (48-pin QFN) | | – | 15 | – | °C/Watt |
| $T_{JC}$ | Package $\theta_{JC}$ (68-pin QFN) | | – | 13 | – | °C/Watt |
| $T_{JC}$ | Package $\theta_{JC}$ (100-pin TQFP) | | – | 10 | – | °C/Watt |

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

| Package | Maximum Peak Temperature | Maximum Time at Peak Temperature |
|---|---|---|
| 48-pin SSOP | 260 °C | 30 seconds |
| 48-pin QFN | 260 °C | 30 seconds |
| 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 |
|---|---|
| 48-pin SSOP | MSL 3 |
| 48-pin QFN | MSL 3 |
| 68-pin QFN | MSL 3 |
| 100-pin TQFP | MSL 3 |

**Figure 13-1. 48-pin (300 mil) SSOP Package Outline**

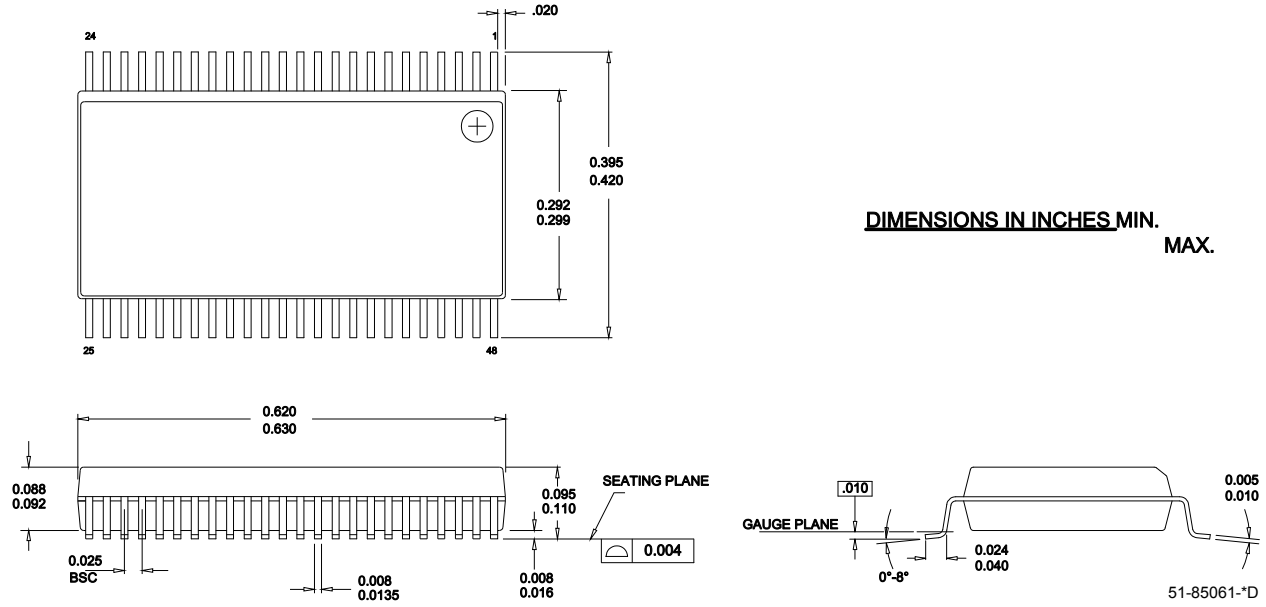
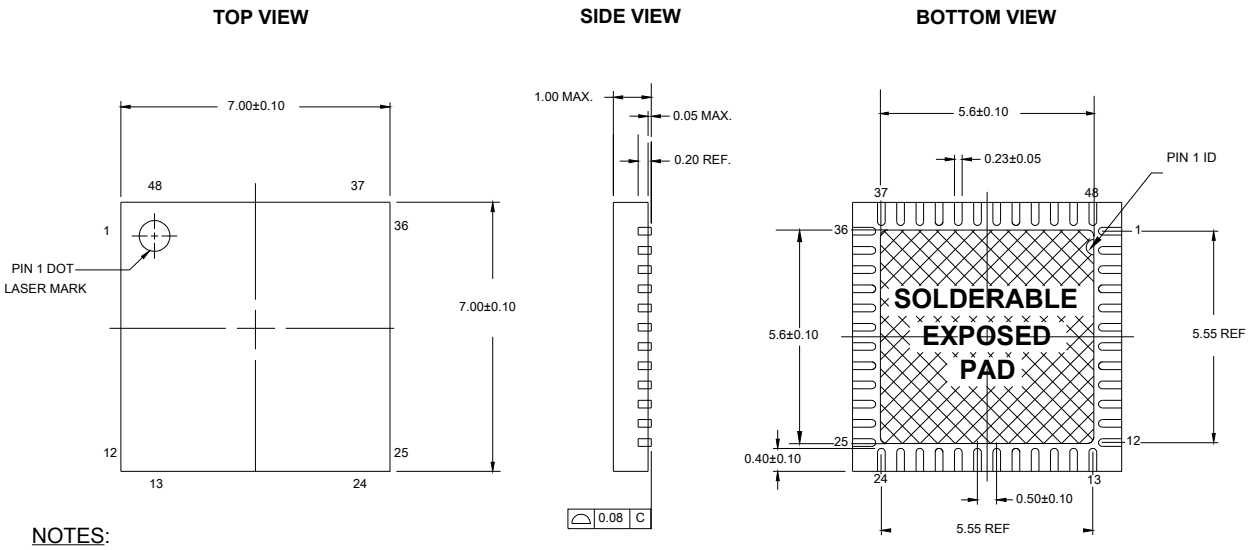



**Figure 13-2. 48-pin QFN Package Outline**



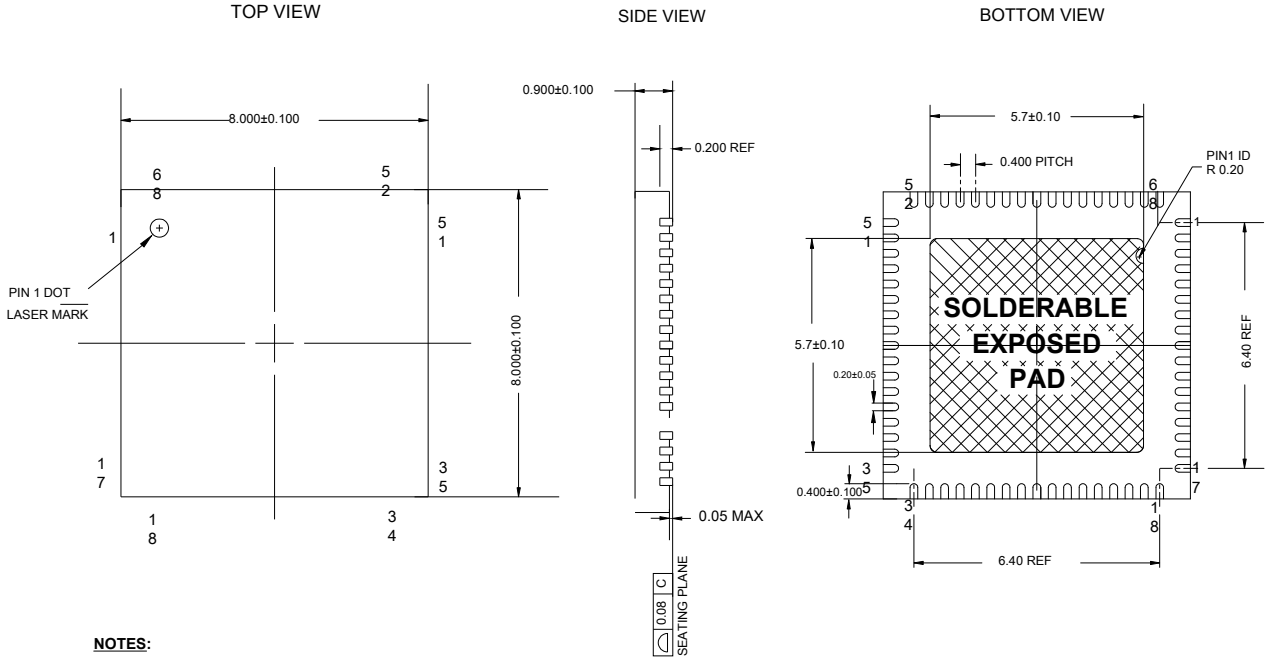

NOTES:

1. HATCH AREA IS SOLDERABLE EXPOSED METAL.
2. REFERENCE JEDEC#: MO-220
3. PACKAGE WEIGHT: 0.13g
4. ALL DIMENSIONS ARE IN MM [MIN/MAX]
5. PACKAGE CODE

| PART # | DESCRIPTION |
|---|---|
| LT48D | LEAD FREE |

001- 45616 *B

**Figure 13-3. 68-pin QFN 8 × 8 with 0.4 mm Pitch Package Outline (Sawn Version)**

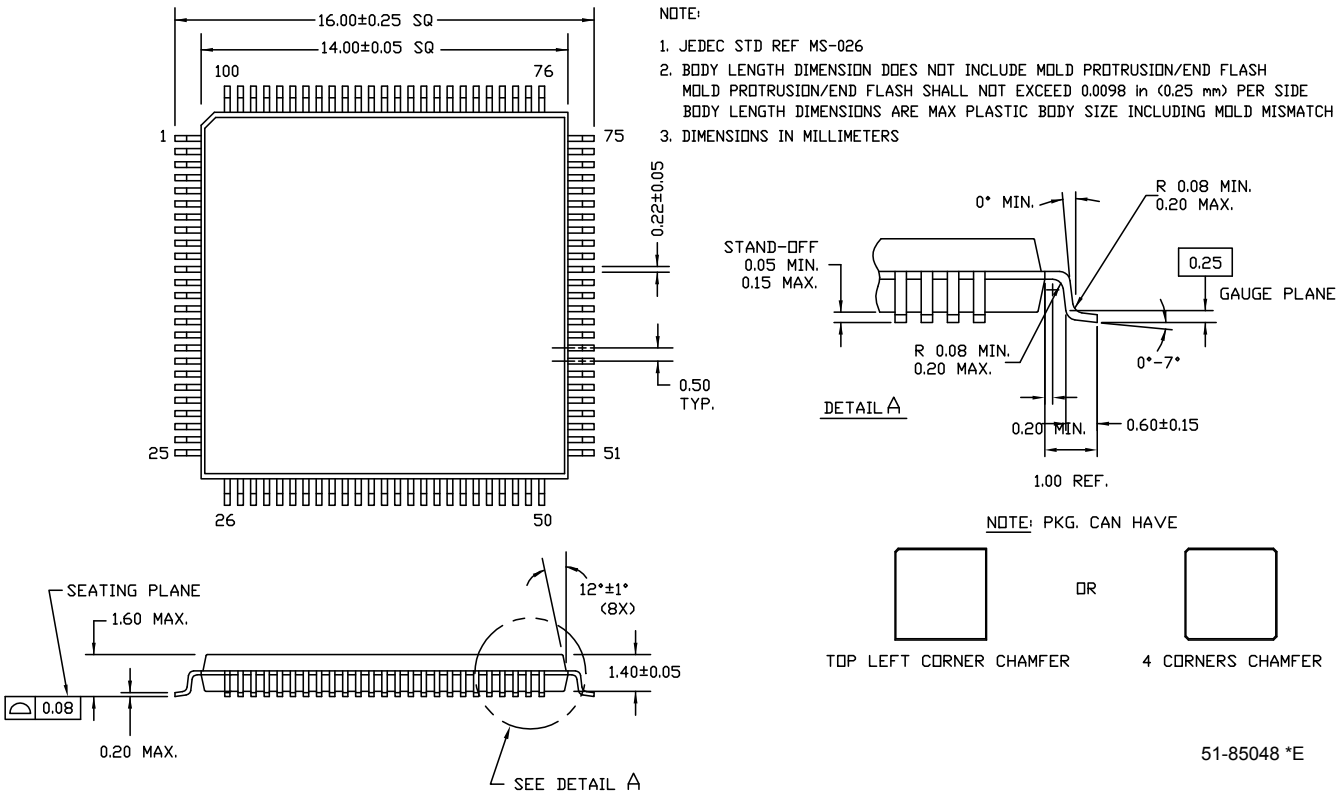



**NOTES:**

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

001-09618 *C

**Figure 13-4. 100-pin TQFP (14 × 14 × 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® | 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 |
| ECC | error correcting code |
| ECO | external crystal oscillator |
| EEPROM | electrically erasable programmable read-only memory |
| EMI | electromagnetic interference |
| EMIF | external memory interface |
| EOC | end of conversion |
| EOF | end of frame |
| EPSR | execution program status register |
| ESD | electrostatic discharge |
| ETM | embedded trace macrocell |

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

| Acronym | Description |
|---|---|
| FIR | finite impulse response, see also IIR |
| FPB | flash patch and breakpoint |
| FS | full-speed |
| GPIO | general-purpose input/output, applies to a PSoC pin |
| HVI | high-voltage interrupt, see also LVI, LVD |
| IC | integrated circuit |
| IDAC | current DAC, see also DAC, VDAC |
| IDE | integrated development environment |
| $I^2C$, 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 |

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

| Acronym | Description |
|---|---|
| PHUB | peripheral hub |
| PHY | physical layer |
| PICU | port interrupt control unit |
| 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 |
| PRES | precise 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 | $I^2C$ serial clock |
| SDA | $I^2C$ serial data |
| S/H | sample and hold |
| SINAD | signal to noise and distortion ratio |
| SIO | special input/output, GPIO with advanced features. See GPIO. |
| SOC | start of conversion |

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

| Acronym | Description |
|---|---|
| SOF | start of frame |
| SPI | Serial Peripheral Interface, a communications protocol |
| SR | slew rate |
| SRAM | static random access memory |
| SRES | software reset |
| SWD | serial wire debug, a test protocol |
| SWV | single-wire viewer |
| TD | transaction descriptor, see also DMA |
| THD | total harmonic distortion |
| TIA | transimpedance amplifier |
| TRM | technical reference manual |
| 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® 3 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® 3: CY8C34 Family Data Sheet Programmable System-on-Chip (PSoC®)<br>Document Number: 001-53304 | | | | |
|---|---|---|---|---|
| **Rev.** | **ECN No.** | **Submission Date** | **Orig. of Change** | **Description of Change** |
| ** | 2714270 | 06/03/09 | VVSK | New data sheet |
| *A | 2758970 | 09/02/09 | MKEA | Updated Part Numbering Conventions<br>Added Section 11.7.5 (EMIF Figures and Tables)<br>Updated GPIO and SIO AC specifications<br>Updated XRES Pin Description and Xdata Address Map specifications<br>Updated DFB and Comparator specifications<br>Updated PHUB features section and RTC in sleep mode<br>Updated IDAC and VDAC DC and Analog Global specifications<br>Updated USBIO AC and Delta Sigma ADC specifications<br>Updated PPOR and Voltage Monitors DC specifications<br>Updated Drive Mode diagram<br>Added 48-QFN Information<br>Updated other electrical specifications |
| *B | 2824546 | 12/09/09 | MKEA | Updated I2C section to reflect 1 Mbps. Updated Table 11-6 and 11- 7 (Boost AC and DC specs); also added Shottky Diode specs. Changed current for sleep/hibernate mode to include SIO; Added footnote to analog global specs. Updated Figures 1-1, 6-2, 7-14, and 8-1. Updated Table 6-2 and Table 6-3 (Hibernate and Sleep rows) and Power Modes section. Updated GPIO and SIO AC specifications. Updated Gain error in IDAC and VDAC specifications. Updated description of $V_{DDA}$ spec in Table 11-1 and removed GPIO Clamp Current parameter. Updated number of UDBs on page 1.<br>Moved FILO from ILO DC to AC table.<br>Added PCB Layout and PCB Schematic diagrams.<br>Updated Fgpioout spec (Table 11-9). Added duty cycle frequency in PLL AC spec table. Added note for Sleep and Hibernate modes and Active Mode specs in Table 11-2. Linked URL in Section 10.3 to PSoC Creator site.<br>Updated Ja and Jc values in Table 13-1. Updated Single Sample Mode and Fast FIR Mode sections. Updated Input Resistance specification in Del-Sig ADC table. Added Tio_init parameter. Updated PGA and UGB AC Specs. Removed SPC ADC. Updated Boost Converter section.<br>Added section 'SIO as Comparator'; updated Hysteresis spec (differential mode) in Table 11-10.<br>Updated $V_{BAT}$ condition and deleted Vstart parameter in Table 11-6.<br>Added 'Bytes' column for Tables 4-1 to 4-5. |
| *C | 2873322 | 02/04/10 | MKEA | Changed maximum value of PPOR_TR to '1'. Updated Vbias specification. Updated PCB Schematic. Updated Figure 8-1 and Figure 6-3. Updated Interrupt Vector table, Updated Sales links. Updated JTAG and SWD specifications. Removed Jp-p and Jperiod from ECO AC Spec table. Added note on sleep timer in Table 11-2. Updated ILO AC and DC specifications. Added Resolution parameter in VDAC and IDAC tables. Updated $I_{OUT}$ typical and maximum values. Changed Temperature Sensor range to –40 °C to +85 °C. Removed Latchup specification from Table 11-1. |

| Description Title: PSoC® 3: CY8C34 Family Data Sheet Programmable System-on-Chip (PSoC®)<br>Document Number: 001-53304 | | | | |
|---|---|---|---|---|
| *D | 2903576 | 04/01/10 | MKEA | Updated Vb pin in PCB Schematic<br>Updated Tstartup parameter in AC Specifications table<br>Added Load regulation and Line regulation parameters to Inductive Boost Regulator DC Specifications table<br>Updated $I_{CC}$ parameter in LCD Direct Drive DC Specs table<br>In page 1, updated internal oscillator range under Precision programmable clocking to start from 3 MHz<br>Updated $I_{OUT}$ parameter in LCD Direct Drive DC Specs table<br>Updated Table 6-2 and Table 6-3<br>Removed DFB block in Figure 1-1.<br>Added bullets on CapSense in page 1; added CapSense column in Section 12<br>Removed some references to footnote [1]<br>Changed INC_Rn cycles from 3 to 2 (Table 4-1)<br>Added footnote in PLL AC Specification table<br>Added PLL intermediate frequency row with footnote in PLL AC Specs table<br>Added UDBs subsection under 11.6 Digital Peripherals<br>Updated Figure 2-6 (PCB Layout)<br>Updated Pin Descriptions section and modified Figures 6-6, 6-8, 6-9<br>Updated LVD in Tables 6-2 and 6-3; modified Low-power modes bullet in page 1<br>Added note to Figures 2-5 and 6-2; Updated Figure 6-2 to add capacitors for $V_{DDA}$ and $V_{DDD}$ pins.<br>Updated boost converter section (6.2.2)<br>Updated Tstartup values in Table 11-3.<br>Removed IPOR rows from Table 11-67.<br>Updated 6.3.1.1, Power Voltage Level Monitors.<br>Updated section 5.2 and Table 11-2 to correct suggestion of execution from flash.<br>Updated IMO max frequency in Figure 6-1, Table 11-77, and Table 11-78.<br>Updated $V_{REF}$ specs in Table 11-21.<br>Updated IDAC uncompensated gain error in Table 11-25.<br>Updated Delay from Interrupt signal input to ISR code execution from ISR code in Table 11-57. Removed other line in table.<br>Added sentence to last paragraph of section 6.1.1.3.<br>Updated $T_{RESP}$, high and low-power modes, in Table 11-24.<br>Updated f_TCK values in Table 11-72 and f_SWDCK values in Table 11-73.<br>Updated SNR condition in Table 11-20.<br>Updated sleep wakeup time in Table 6-3 and Tsleep in Table 11-3.<br>Added 1.71 V <= $V_{DDD}$ < 3.3 V, SWD over USBIO pins value to Table 11-73.<br>Removed mention of hibernate reset (HRES) from page 1 features, Table 6-3, Section 6.2.1.4, Section 6.3, and Section 6.3.1.1. Change PPOR/PRES to TBDs in Section 6.3.1.1, Section 6.4.1.6 (changed PPOR to reset), Table 11-3 (changed PPOR to PRES), Table 11-67 (changed title, values TBD), and Table 11-68 (changed PPOR_TR to PRES_TR).<br>Added sentence saying that LVD circuits can generate a reset to Section 6.3.1.1.<br>Changed $I_{DD}$ values on page 1, page 5, and Table 11-2.<br>Changed resume time value in Section 6.2.1.3.<br>Changed ESD HBM value in Table 11-1.<br>Changed sample rate row in Table 11-20. Removed $V_{DDA}$ = 1.65 V rows and changed BWag value in Table 11-22.<br>Changed $V_{IOFF}$ values and changed CMRR value in Table 11-23.<br>Changed INL max value in Table 11-27.<br>Added max value to the Quiescent current specs in Tables 11-29 and 11-31.<br>Changed occurrences of “Block” to “Row” and deleted the “ECC not included” footnote in Table 11-55.<br>Changed max response time value in Tables 11-68 and 11-70.<br>Changed the Startup time in Table 11-78.<br>Added condition to intermediate frequency row in Table 11-84.<br>Added row to Table 11-68.<br>Added brown out note to Section 11.8.1. |

| Description Title: PSoC® 3: CY8C34 Family Data Sheet Programmable System-on-Chip (PSoC®)<br>Document Number: 001-53304 | | | | |
|---|---|---|---|---|
| *E | 2938381 | 05/27/10 | MKEA | Replaced $V_{DDIO}$ with $V_{DDD}$ in USBIO diagram and specification tables, added text in USBIO section of Electrical Specifications.<br>Added Table 13-2 (Package MSL)<br>Modified Tstorag condition and changed max spec to 100<br>Added bullet (Pass) under ALU (section 7.2.2.2)<br>Added figures for kHzECO and MHzECO in the External Oscillator section<br>Updated Figure 6-1(Clocking Subsystem diagram)<br>Removed CPUCLK_DIV in table 5-2, Deleted Clock Divider SFR subsection<br>Updated PSoC Creator Framework image<br>Updated SIO DC Specifications ($V_{IH}$ and $V_{IL}$ parameters)<br>Updated bullets in Clocking System and Clocking Distribution sections<br>Updated Figure 8-2<br>Updated PCB Layout and Schematic, updated as per MTRB review comments<br>Updated Table 6-3 (power changed to current)<br>In 32kHZ EC DC Specifications table, changed $I_{CC}$ Max to 0.25<br>In IMO DC Specifications table, updated Supply Current values<br>Updated GPIO DC Specs table<br>Modified to support a maximum 50MHz CPU speed |
| *F | 2958674 | 06/22/10 | SHEA | Minor ECN to post data sheet to external website |
| *G | 2989685 | 08/04/10 | MKEA | Added USBIO 22 ohm DP and DM resistors to Simplified Block Diagram<br>Added to Table 6-6 a footnote and references to same.<br>Added sentences to the resistive pull-up and pull-down description bullets.<br>Added sentence to Section 6.4.11, Adjustable Output Level.<br>Updated section 5.5 External Memory Interface<br>Updated Table 11-73 JTAG Interface AC Specifications<br>Updated Table 11-74 SWD Interface AC Specifications<br>Updated style changes as per new template. |
| *H | 3078568 | 11/04/10 | MKEA | Updated Table 11-2, “DC Specifications,” on page 66<br>Updated “Current Digital-to-analog Converter (IDAC)” on page 88<br>Updated “Voltage Digital to Analog Converter (VDAC)” on page 93 |
| *I | 3107314 | 12/10/2010 | MKEA | Updated delta-sigma tables and graphs.<br>Updated Flash AC specs<br>Formatted table 11.2.<br>Updated interrupt controller table<br>Updated transimpedance amplifier section<br>Updated SIO DC specs table<br>Updated Voltage Monitors DC Specifications table<br>Updated LCD Direct Drive DC specs table<br>Replaced the Discrete Time Mixer and Continuous Time Mixer tables with Mixer DC and AC specs tables<br>Updated $ESD_{HBM}$ value.<br>Updated IDAC and VDAC sections<br>Removed ESO parts from ordering information<br>Changed USBIO pins from NC to DNU and removed redundant USBIO pin description notes<br>Updated POR with brown out DC and AC specs<br>Updated PGA AC specs<br>Updated 32 kHz External Crystal DC Specifications<br>Updated opamp AC specs<br>Updated XRES IO specs<br>Updated Inductive boost regulator section<br>Delta sigma ADC spec updates<br>Updated comparator section<br>Removed buzz mode from Power Mode Transition diagram<br>Updated opamp DC and AC spec tables<br>Updated PGA DC table |
| *J | 3179219 | 02/22/2011 | MKEA | Updated conditions for flash data retention time.<br>Updated 100-pin TQFP package spec.<br>Updated EEPROM AC specifications. |

| Description Title: PSoC® 3: CY8C34 Family Data Sheet Programmable System-on-Chip (PSoC®)<br>Document Number: 001-53304 | | | | |
|---|---|---|---|---|
| *K | 3200146 | 03/28/2011 | MKEA | Removed Preliminary status from the data sheet.<br>Updated JTAG ID<br>Deleted Cin_G1, ADC input capacitance from Delta-Sigma ADC DC spec table<br>Updated JTAG Interface AC Specifications and SWD Interface Specifications tables<br>Updated USBIO DC specs<br>Added 0.01 to max speed<br>Updated Features on page 1<br>Added Section 5.5, Nonvolatile Latches<br>Updated Flash AC specs<br>Added CAN DC specs<br>Updated delta-sigma graphs, noise histogram figures and RMS Noise spec tables<br>Add reference to application note AN58304 in section 8.1<br>Updated 100-pin TQFP package spec<br>Added oscillator, I/O, VDAC, regulator graphs<br>Updated JTAG/SWD timing diagrams<br>Updated GPIO and SIO AC specs<br>Updated POR with Brown Out AC spec table<br>Updated IDAC graphs<br>Added DMA timing diagram, interrupt timing and interrupt vector, I2C timing diagrams<br>Updated opamp graphs and PGA graphs<br>Added full chip performance graphs<br>Changed MHzECO range.<br>Added “Solder Reflow Peak Temperature” table. |
| *L | 3259185 | 05/17/2011 | MKEA | Added JTAG and SWD interface connection diagrams<br>Updated $T_{JA}$and $T_{JC}$ values in Table 13-1<br>Changed typ and max values for the TCVos parameter in Opamp DC specifications table.<br>Updated Clocking subsystem diagram.<br>Changed Vssd to Vssb in the PSoC Power System diagram<br>Updated Ordering information. |

# 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 | cypress.com/go/automotive |
| Clocks & Buffers | cypress.com/go/clocks |
| Interface | cypress.com/go/interface |
| Lighting & Power Control | cypress.com/go/powerpsoc<br>cypress.com/go/plc |
| Memory | cypress.com/go/memory |
| Optical & Image Sensing | cypress.com/go/image |
| PSoC | cypress.com/go/psoc |
| Touch Sensing | cypress.com/go/touch |
| USB Controllers | cypress.com/go/USB |
| Wireless/RF | cypress.com/go/wireless |

### PSoC Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 5

© Cypress Semiconductor Corporation, 2009-2011. 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.



CapSense®, PSoC® 3, PSoC® 5, and PSoC® Creator™ are trademarks and PSoC® is a registered trademark of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Purchase of I2C components from Cypress or one of its sublicensed Associated Companies conveys a license under the Philips I2C Patent Rights to use these components in an I2C system, provided that the system conforms to the I2C Standard Specification as defined by Philips.

ARM is a registered trademark, and Keil, and RealView are trademarks, of ARM Limited. All products and company names mentioned in this document may be the trademarks of their respective holders..