



Welcome to [E-XFL.COM](#)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	50MHz
Connectivity	EBI/EMI, I ² C, LINbus, SPI, UART/USART
Peripherals	CapSense, DMA, LCD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	1.71V ~ 5.5V
Data Converters	A/D 16x12b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy8c3446lti-083

More Information

Cypress provides a wealth of data at www.cypress.com to help you to select the right PSoC device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the knowledge base article [KBA86521, How to Design with PSoC 3, PSoC 4, and PSoC 5LP](#). Following is an abbreviated list for PSoC 3:

- Overview: [PSoC Portfolio](#), [PSoC Roadmap](#)
- Product Selectors: [PSoC 1](#), [PSoC 3](#), [PSoC 4](#), [PSoC 5LP](#)
In addition, PSoC Creator includes a device selection tool.
- Application notes: Cypress offers a large number of PSoC application notes and [code examples](#) covering a broad range of topics, from basic to advanced level. Recommended application notes for getting started with PSoC 3 are:
 - [AN54181: Getting Started With PSoC 3](#)
 - [AN61290: Hardware Design Considerations](#)
 - [AN57821: Mixed Signal Circuit Board Layout](#)
 - [AN58304: Pin Selection for Analog Designs](#)
 - [AN81623: Digital Design Best Practices](#)
 - [AN73854: Introduction To Bootloaders](#)
- Development Kits:
 - [CY8CKIT-030](#) is designed for analog performance, for developing high-precision analog, low-power, and low-voltage applications.
 - [CY8CKIT-001](#) provides a common development platform for any one of the PSoC 1, PSoC 3, PSoC 4, or PSoC 5LP families of devices.
 - The [MiniProg3](#) device provides an interface for flash programming and debug.
- Technical Reference Manuals (TRM)
 - [Architecture TRM](#)
 - [Registers TRM](#)
 - [Programming Specification](#)

PSoC Creator

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

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

Figure 1. Multiple-Sensor Example Project in PSoC Creator

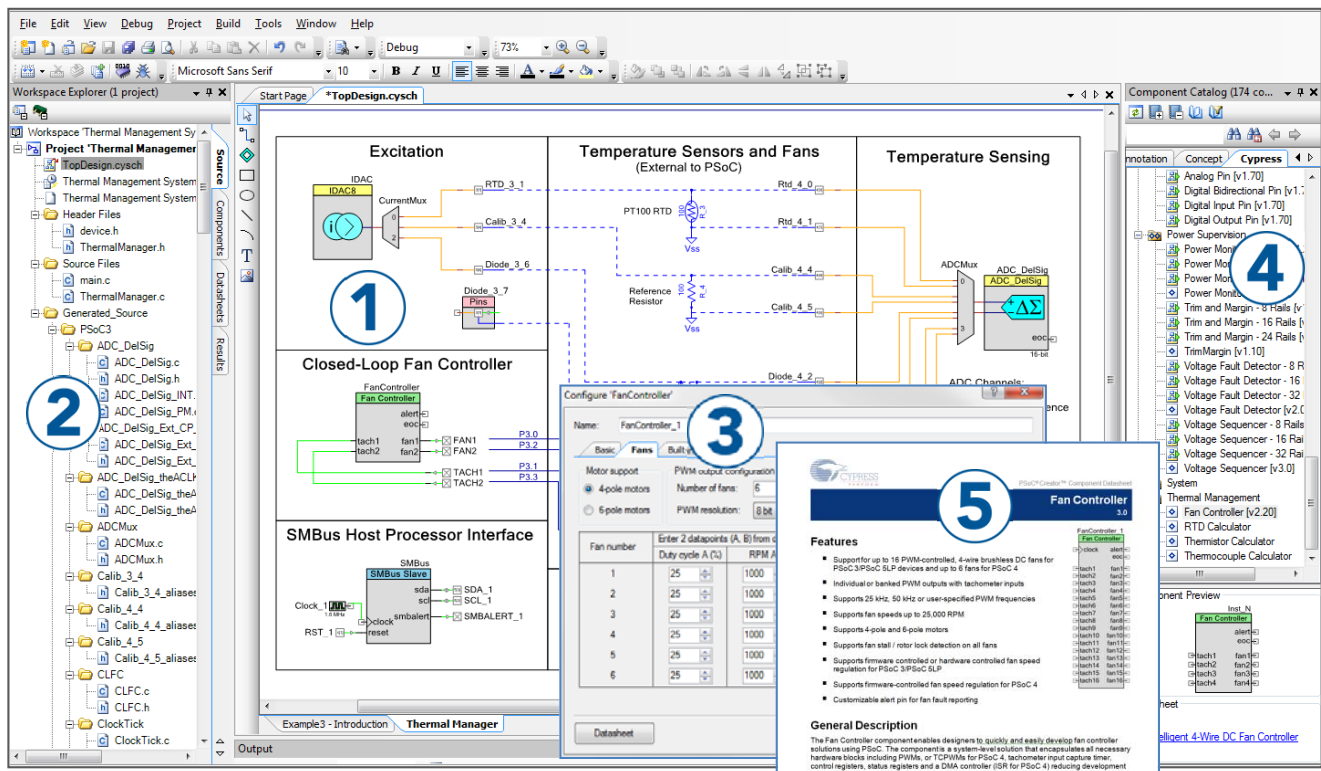


Figure 2-6. 100-pin TQFP Part Pinout

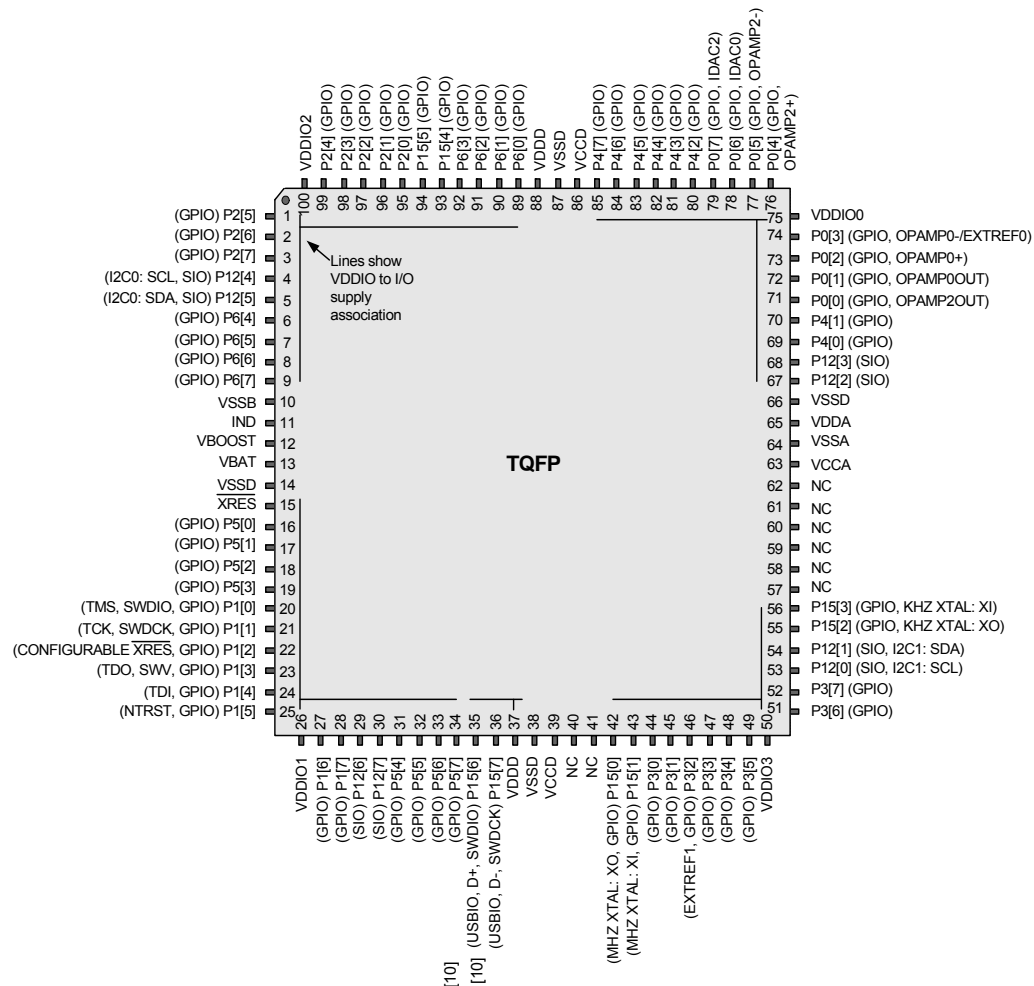


Table 2-1. VDDIO and Port Pin Associations

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

Note

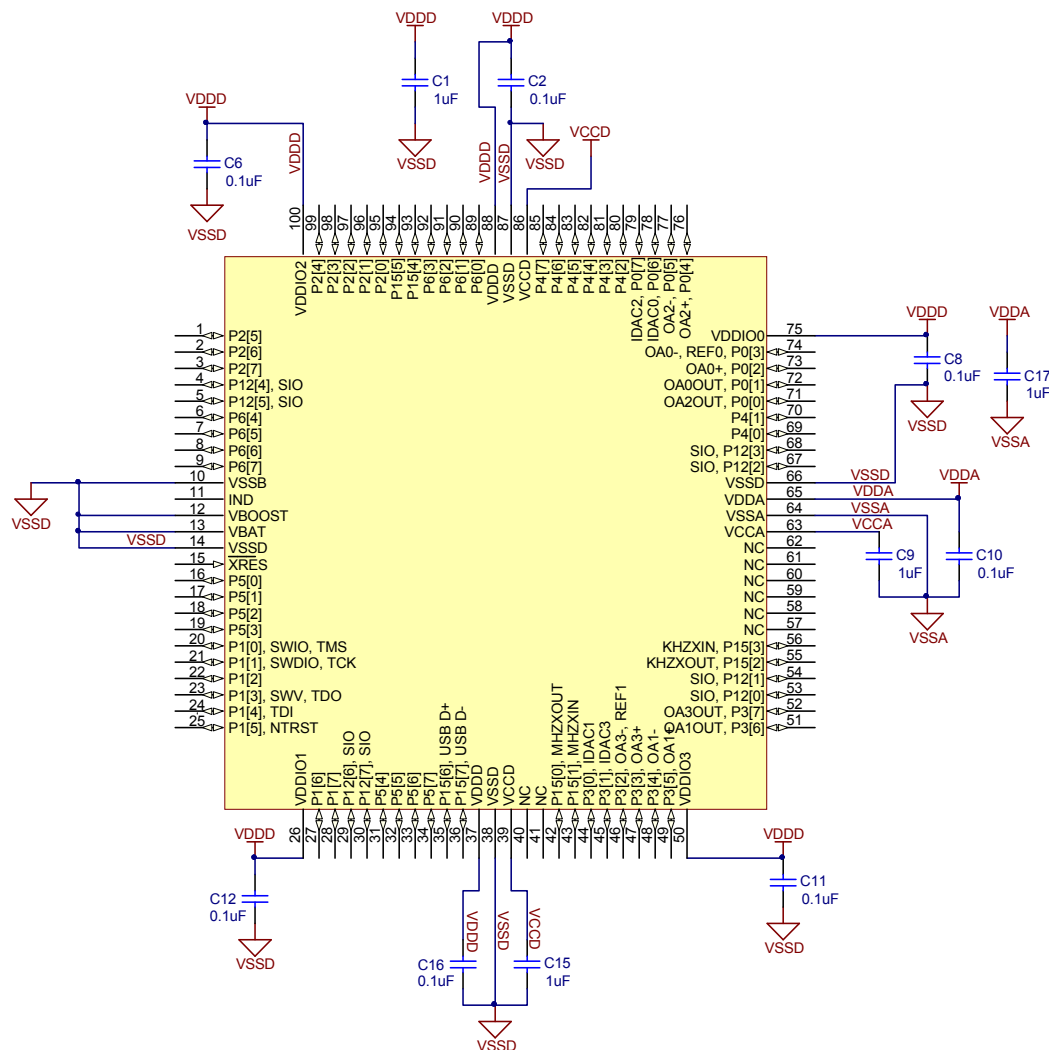
10. Pins are Do Not Use (DNU) on devices without USB. The pin must be left floating.

Figure 2-7 and Figure 2-8 on page 11 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 2-7 and Power System on page 30. 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 2-7. 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 2-8 on page 11.

For more information on pad layout, refer to <http://www.cypress.com/cad-resources/psoc-3-cad-libraries>.

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 the analog core regulator or the input to the analog core. Requires a 1uF capacitor to VSSA. The regulator output is not designed to drive external circuits. **Note that if you use the device with an external core regulator (externally regulated mode), the voltage applied to this pin must not exceed the allowable range of 1.71 V to 1.89 V.** When using the internal core regulator, (internally regulated mode, the default), do not tie any power to this pin. For details see [Power System](#) on page 30.

VCCD.

Output of the digital core regulator or the input to the digital core. The two VCCD pins must be shorted together, with the trace between them as short as possible, and a 1uF capacitor to VSSD. The regulator output is not designed to drive external circuits. **Note that if you use the device with an external core regulator (externally regulated mode), the voltage applied to this pin must not exceed the allowable range of 1.71 V to 1.89 V.** When using the internal core regulator (internally regulated mode, the default), do not tie any power to this pin. For details see [Power System](#) on page 30.

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.

$\overline{\text{XRES}}$ (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 23.

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
- 512-byte instruction cache between CPU and flash
- 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.

4.3.1.3 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 16 lists the available Boolean instructions.

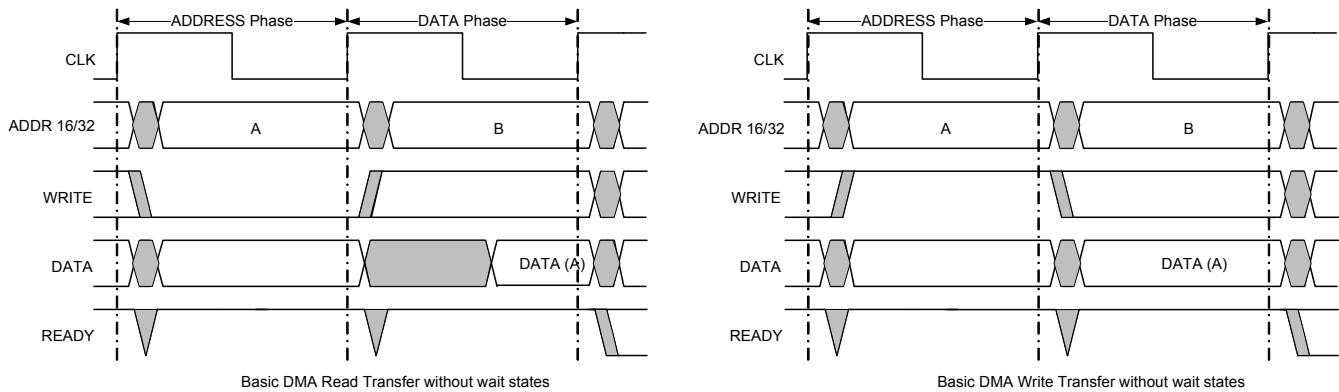
Table 4-3. Data Transfer Instructions

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

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



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

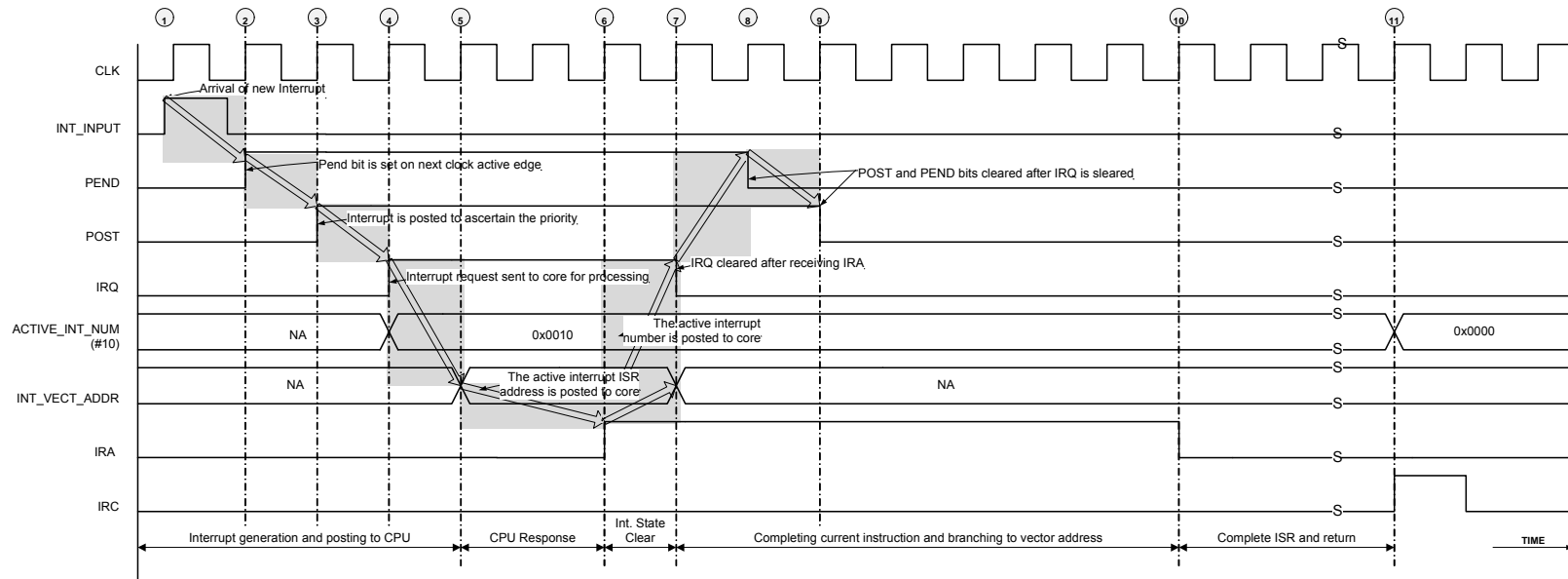
When an interrupt is pending, the current instruction is completed and the program counter is pushed onto the stack. Code execution then jumps to the program address provided by the vector. After the ISR is completed, a RETI instruction is executed and returns execution to the instruction following the 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 represents typical flow of events when an interrupt triggered. Figure 4-3 on page 20 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

Table 4-8. Interrupt Vector Table

#	Fixed Function	DMA	UDB
0	LVD	phub_termout0[0]	udb_intr[0]
1	Cache/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 ² C	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]

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.

5.7.4 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.

5.7.5 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 36.

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.

5.7.5.1 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 24.

Table 5-5. XDATA Data Address Map

Address Range	Purpose
0x00 0000 – 0x00 1FFF	SRAM
0x00 4000 – 0x00 42FF	Clocking, PLLs, and oscillators
0x00 4300 – 0x00 43FF	Power management
0x00 4400 – 0x00 44FF	Interrupt controller
0x00 4500 – 0x00 45FF	Ports interrupt control
0x00 4700 – 0x00 47FF	Flash programming interface
0x00 4800 – 0x00 48FF	Cache controller
0x00 4900 – 0x00 49FF	I ² C controller
0x00 4E00 – 0x00 4EFF	Decimator
0x00 4F00 – 0x00 4FFF	Fixed timer/counter/PWMs
0x00 5000 – 0x00 51FF	I/O ports control
0x00 5400 – 0x00 54FF	External Memory Interface (EMIF) control registers
0x00 5800 – 0x00 5FFF	Analog Subsystem interface
0x00 6000 – 0x00 60FF	USB controller
0x00 6400 – 0x00 6FFF	UDB Working Registers
0x00 7000 – 0x00 7FFF	PHUB configuration
0x00 8000 – 0x00 8FFF	EEPROM
0x00 A000 – 0x00 A400	CAN
0x01 0000 – 0x01 FFFF	Digital Interconnect configuration
0x05 0220 – 0x05 02F0	Debug controller
0x08 0000 – 0x08 1FFF	Flash ECC bytes
0x80 0000 – 0xFF FFFF	External Memory Interface

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) can be used as a low power master clock. It can also generate time intervals using the fast timewheel.

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

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

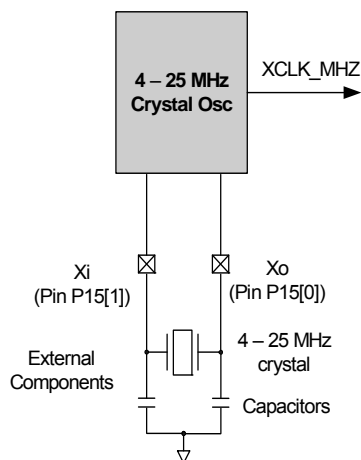
6.1.2 External Oscillators

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

6.1.2.1 MHz External Crystal Oscillator

The MHzECO provides high frequency, high precision clocking using an external crystal (see Figure 6-2). It supports a wide variety of crystal types, in the range of 4 to 25 MHz. When used in conjunction with the PLL, it can generate other clocks up to the device's maximum frequency (see "Phase-Locked Loop" section on page 28). 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

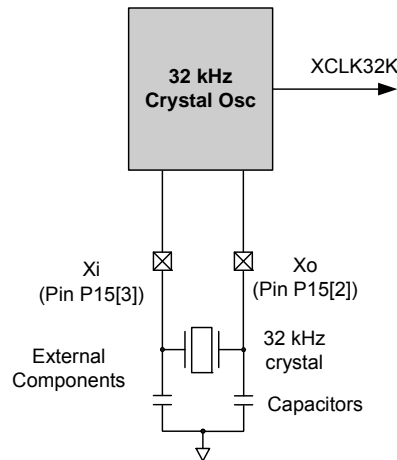


6.1.2.2 32.768-kHz ECO

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



It is recommended that the external 32.768-kHz watch crystal have a load capacitance (CL) of 6 pF or 12.5 pF. Check the crystal manufacturer's datasheet. The two external capacitors, CL1 and CL2, are typically of the same value, and their total capacitance, $CL1 \cdot CL2 / (CL1 + CL2)$, including pin and trace capacitance, should equal the crystal CL value. For more information, refer to application note [AN54439: PSoC 3 and PSoC 5 External Oscillators](#). See also pin capacitance specifications in the "GPIO" section on page 79.

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 master clock is used to select and supply the fastest clock in the system for general clock requirements and clock synchronization of the PSoC device.
- Bus Clock 16-bit divider uses the master clock to generate the 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

boost typically draws 250 μ A in active mode and 25 μ A in standby mode. The boost operating modes must be used in conjunction with chip power modes to minimize total power consumption. Table 6-4 lists the boost power modes available in different chip power modes.

Table 6-4. Chip and Boost Power Modes Compatibility

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

6.2.2.1 Boost Firmware Requirements

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

6.2.2.2 Boost Design Process

Correct operation of the boost converter requires specific component values determined for each design's unique operating conditions. The C_{BAT} capacitor, Inductor, Schottky diode, and C_{BOOST} capacitor components are required with the values specified in the electrical specifications, Table 11-7 on page 77. The only variable component value is the inductor L_{BOOST} which is primarily sized for correct operation of the boost across operating conditions and secondarily for efficiency. Additional operating region constraints exist for V_{OUT} , V_{BAT} , I_{OUT} , and T_A .

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

1. Choose desired V_{BAT} , V_{OUT} , T_A , and I_{OUT} operating condition ranges for the application.
2. Determine if V_{BAT} and V_{OUT} ranges fit the boost operating range based on the **T_A range over V_{BAT} and V_{OUT}** chart, Figure 11-8 on page 77. If the operating ranges are not met,

modify the operating conditions or use an external boost regulator.

3. Determine if the desired ambient temperature (T_A) range fits the ambient temperature operating range based on the **T_A range over V_{BAT} and V_{OUT}** chart, Figure 11-8 on page 77. If the temperature range is not met, modify the operating conditions and return to step 2, or use an external boost regulator.
4. Determine if the desired output current (I_{OUT}) range fits the output current operating range based on the **I_{OUT} range over V_{BAT} and V_{OUT}** chart, Figure 11-9 on page 77. If the output current range is not met, modify the operating conditions and return to step 2, or use an external boost regulator.
5. Find the allowed inductor values based on the **L_{BOOST} values over V_{BAT} and V_{OUT}** chart, Figure 11-10 on page 77.
6. Based on the allowed inductor values, inductor dimensions, inductor cost, boost efficiency, and V_{RIPPLE} choose the optimum inductor value for the system. Boost efficiency and V_{RIPPLE} typical values are provided in the **Efficiency vs V_{BAT}** and **V_{RIPPLE} vs V_{BAT}** charts, Figure 11-11 on page 78 through Figure 11-14 on page 78. In general, if high efficiency and low V_{RIPPLE} are most important, then the highest allowed inductor value should be used. If low inductor cost or small inductor size are most important, then one of the smaller allowed inductor values should be used. If the allowed inductor(s) efficiency, V_{RIPPLE} , cost or dimensions are not acceptable for the application than an external boost regulator should be used.

6.3 Reset

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

- **Power source monitoring** – The analog and digital power voltages, V_{DDA} , V_{DDD} , V_{CCA} , and V_{CCD} are monitored in several different modes during power up, active mode, and sleep mode (buzzing). If any of the voltages goes outside predetermined ranges then a reset is generated. The monitors are programmable to generate an interrupt to the processor under certain conditions before reaching the reset thresholds.
- **External** – The device can be reset from an external source by pulling the reset pin (XRES) low. The XRES pin includes an internal pull-up to V_{DDIO1} . V_{DDD} , V_{DDA} , and V_{DDIO1} 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.

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.

7.1.4.2 Component Catalog

The component catalog is a repository of reusable design elements that select device functionality and customize your PSoC device. It is populated with an impressive selection of content; from simple primitives such as logic gates and device registers, through the digital timers, counters and PWMs, plus analog components such as ADC, DACs, and filters, and communication protocols, such as I²C, USB, and CAN. See [Example Peripherals](#) on page 43 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

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

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

7.1.4.5 Nonintrusive Debugging

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

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

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](#).

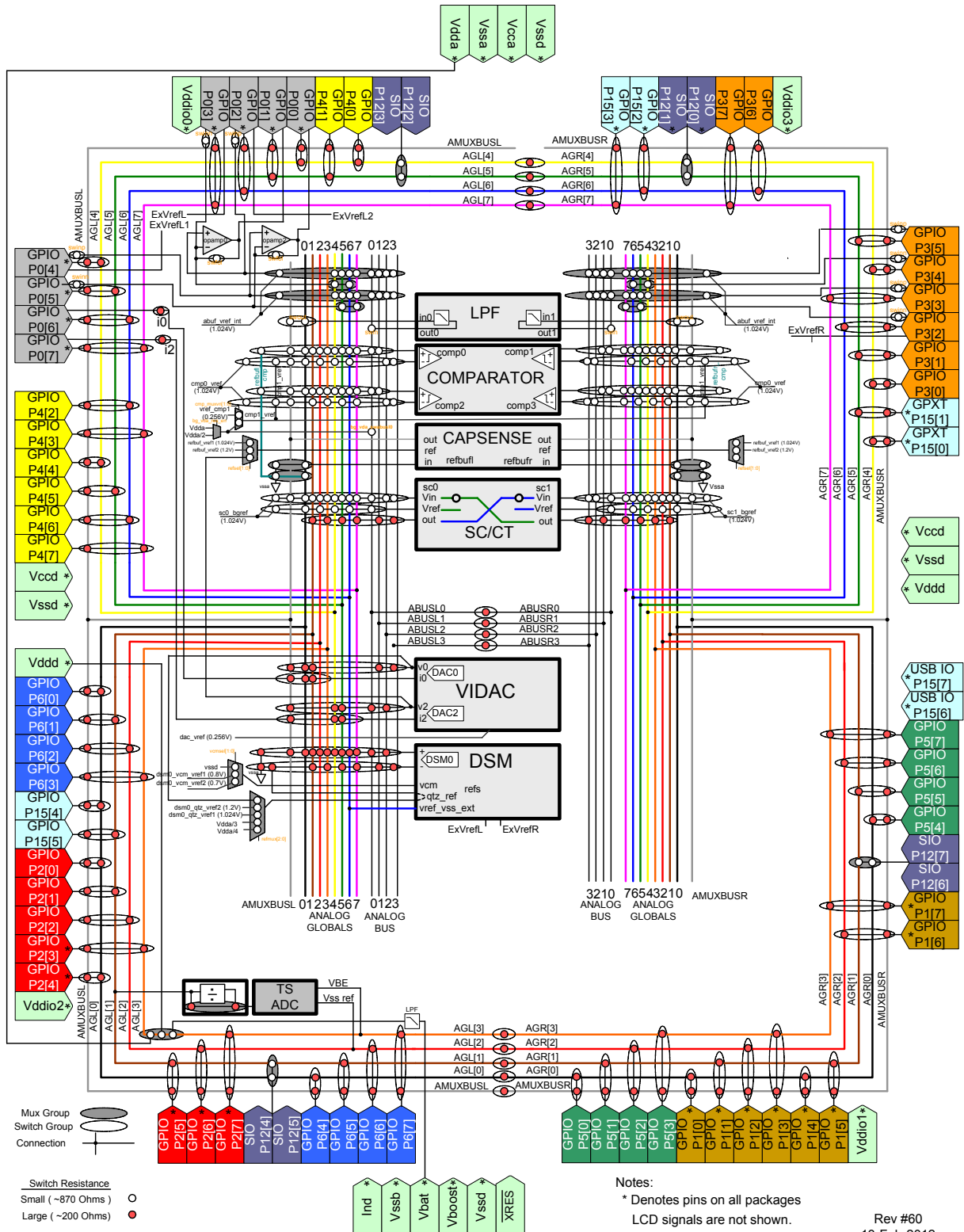
8.1.1 Features

- 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



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

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 μ A. 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.

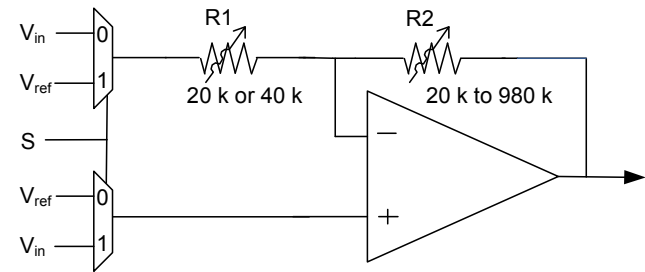
8.5.3 PGA

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 8-8. The schematic in Figure 8-8 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 8-8. 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 I_{in} , the output voltage is $V_{REF} - I_{in} \times R_{fb}$, where V_{REF} is the value placed on the non inverting input. The feedback resistor R_{fb} is programmable between 20 K Ω and 1 M Ω through a configuration register. Table 8-4 shows the possible values of R_{fb} and associated configuration settings.

Table 8-4. Feedback Resistor Settings

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

Figure 8-9. Continuous Time TIA Schematic

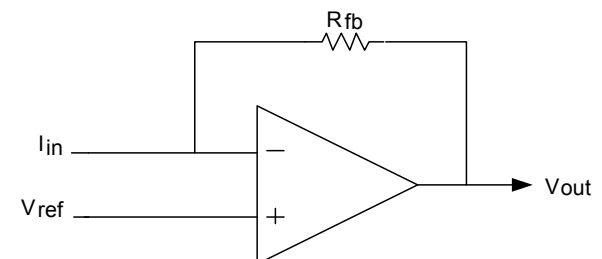


Table 11-12. SIO AC Specifications (continued)

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}	–	–	33	MHz

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

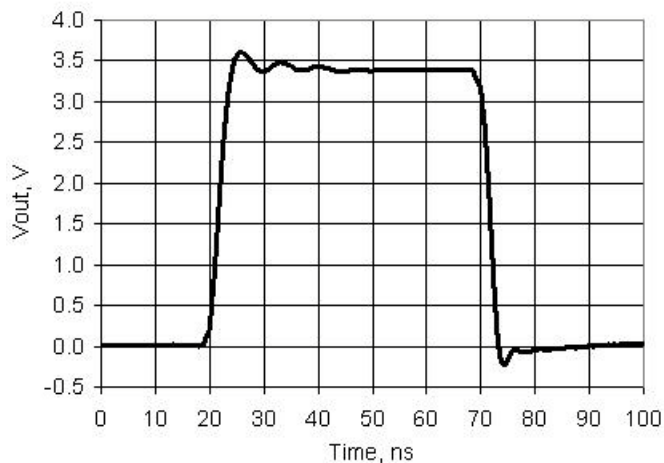
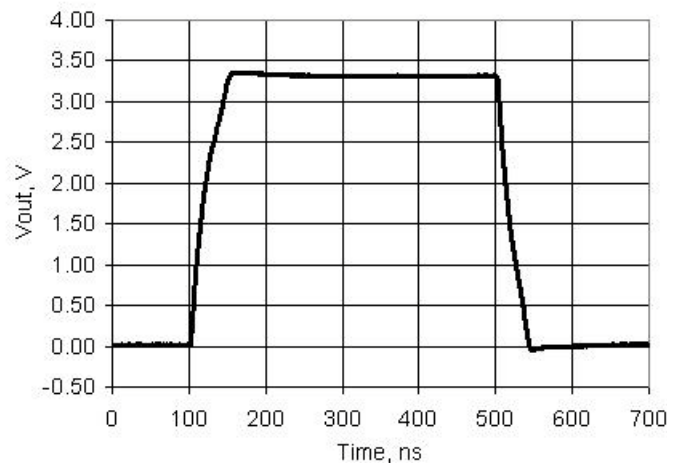


Figure 11-21. SIO Output Rise and Fall Times, Slow Strong Mode, V_{DDIO} = 3.3 V, 25 pF Load



11.4.4 XRES

Table 11-17. 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
R _{pullup}	Pull-up resistor		3.5	5.6	8.5	kΩ
C _{IN}	Input capacitance ^[46]		–	3	–	pF
V _H	Input voltage hysteresis (Schmitt-Trigger) ^[46]		–	100	–	mV
I _{diode}	Current through protection diode to V _{DDIO} and V _{SSIO}		–	–	100	μA

Table 11-18. XRES AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
T _{RESET}	Reset pulse width		1	–	–	μs

Note

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

Figure 11-30. Opamp Noise vs Frequency, Power Mode = High, $V_{DDA} = 5V$

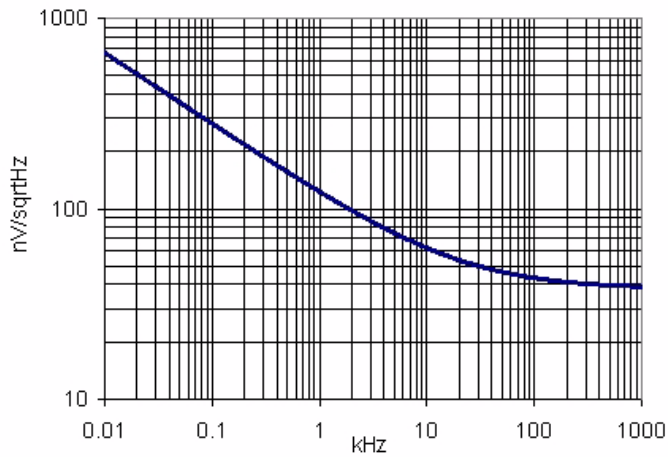


Figure 11-31. Opamp Step Response, Rising

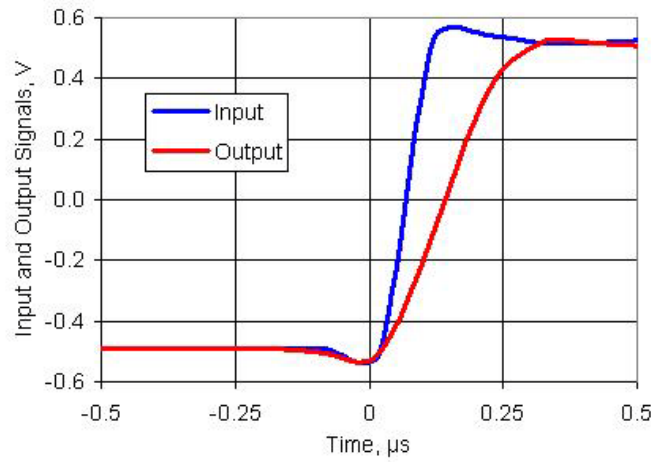
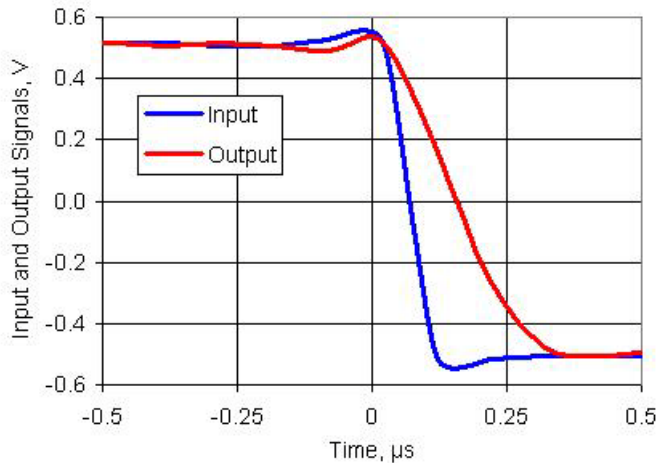


Figure 11-32. Opamp Step Response, Falling



11.6 Digital Peripherals

Specifications are valid for $-40\text{ }^{\circ}\text{C} \leq T_A \leq 85\text{ }^{\circ}\text{C}$ and $T_J \leq 100\text{ }^{\circ}\text{C}$, except where noted. Specifications are valid for 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 11-41. 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 11-42. 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 11-43. 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 11-44. 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

Description Title: PSoC® 3: CY8C34 Family Datasheet Programmable System-on-Chip (PSoC®) (continued)
Document Number: 001-53304

Revision	ECN	Submission Date	Orig. of Change	Description of Change
*P	3732521	09/03/2012	MKEA	<p>Replaced I_{DDDR} and I_{DDAR} specs in Table 11-2, “DC Specifications,” on page 71 that were dropped out in *N revision.</p> <p>Updated Table 11-32, “Mixer DC Specifications,” on page 102, V_{OS} Max value from 10 to 15.</p> <p>Updated Table 11-21, “12-bit Delta-sigma ADC DC Specifications,” on page 91, I_{DD 12} Max value from 1.4 to 1.95 mA</p> <p>Replaced PSoC® 3 Programming AN62391 with TRM in footnote #59 and Section Table 9., “Programming, Debug Interfaces, Resources,” on page 65</p> <p>Removed Figure 11-8 (Efficiency vs Vout)</p> <p>Removed 62-MHz sub-row in Table 11-2, “DC Specifications,” on page 71</p> <p>Updated Table 11-19, “Opamp DC Specifications,” on page 88, I_{DD} Quiescent current row values from 200 and 270 to 250 and 400 respectively.</p> <p>Updated conditions for Storage Temperature in Table 11-1, “Absolute Maximum Ratings DC Specifications[18],” on page 70</p> <p>Updated conditions and min values for NVL data retention time in Table 11-58, “NVL AC Specifications,” on page 109.</p> <p>Updated Table 11-75, “ILO DC Specifications,” on page 118.</p> <p>Removed following pruned parts from “Ordering Information” section on page 121.</p> <p>CY8C3444AXI-116</p> <p>CY8C3445LTI-089</p> <p>CY8C3446AXI-105</p> <p>CY8C3446LTI-083</p> <p>CY8C3446PVI-091</p> <p>CY8C3446PVI-102</p> <p>Updated PSoC 3 boost circuit value throughout the document.</p> <p>Updated package diagram 51-85061 to *F revision.</p>
*Q	3922905	03/06/2013	MKEA	<p>Updated I_{DD XX} parameters under Table 11-21, “12-bit Delta-sigma ADC DC Specifications,” on page 91.</p> <p>Updated I²C section and updated GPIO and SIO DC specification tables.</p>
*R	4064707	07/18/2013	MKEA	<p>Added USB test ID in Features.</p> <p>Updated schematic in Section 2.</p> <p>Added paragraph for device reset warning in Section 5.4.</p> <p>Added NVL bit for DEBUG_EN in Section 5.5.</p> <p>Updated UDB PLD array diagram in Section 7.2.1.</p> <p>Changed Tstartup specs in Section 11.2.1.</p> <p>Changed GPIO rise and fall time specs in Section 11.4.</p> <p>Added Opamp IIB spec in Section 11.5.1.</p> <p>Added IMO spec condition: pre-assembly in Section 11.9.1.</p> <p>Added Appendix for CSP package (preliminary)</p>
*S	4118845	09/10/2013	MKEA	<p>Removed T_{STG} spec and added note clarifying the maximum storage temperature range in Table 11-1.</p> <p>Updated Vos spec conditions and TCVo_s in Table 11-21.</p> <p>Updated 100-TQFP package diagram.</p>
*T	4188568	11/14/2013	MKEA	<p>Updated delta-sigma Vos spec conditions.</p> <p>Added SIO Comparator specifications.</p>
*U	4385782	05/21/2014	MKEA	<p>Updated General Description and Features.</p> <p>Added More Information and PSoC Creator sections.</p> <p>Updated 100-pin TQFP package diagram.</p>