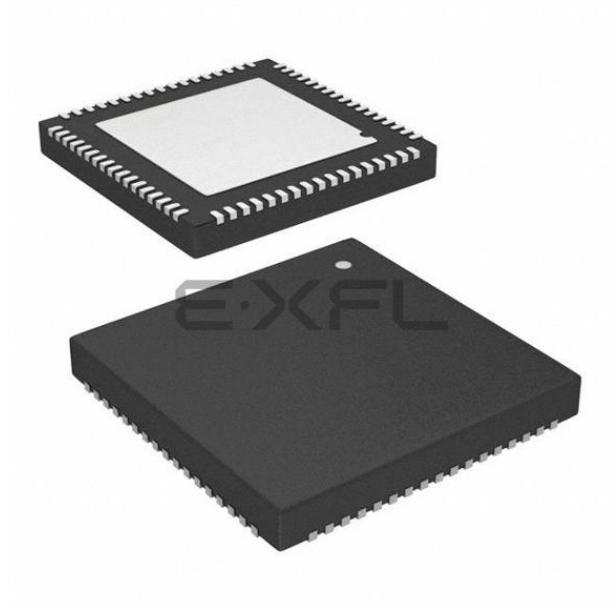
Infineon Technologies - CY8C5248LTI-030 Datasheet



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	ARM® Cortex®-M3
Core Size	32-Bit Single-Core
Speed	40MHz
Connectivity	I²C, LINbus, SPI, UART/USART, USB
Peripherals	CapSense, DMA, LCD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 1x12b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	68-VFQFN Exposed Pad
Supplier Device Package	68-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy8c5248lti-030

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

  The Cortex-M3 does not support ARM instructions.
- Bit-band support for the SRAM region. Atomic bit-level write and read operations for SRAM addresses.
- Unaligned data storage and access. Contiguous storage of data of different byte lengths.
- Operation at two privilege levels (privileged and user) and in two modes (thread and handler). Some instructions can only be executed at the privileged level. There are also two stack pointers: Main (MSP) and Process (PSP). These features support a multitasking operating system running one or more user-level processes.
- Extensive interrupt and system exception support.

4.1.2 Cortex-M3 Operating Modes

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

Table 4-1. Operational Level

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

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

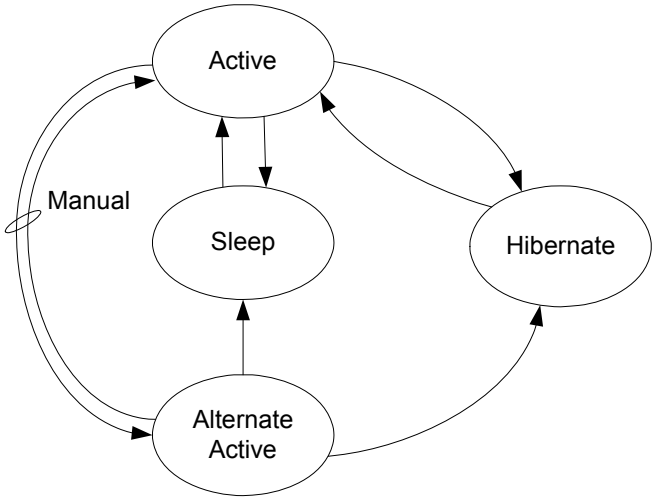
4.1.3 CPU Registers

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

Table 4-2. Cortex M3 CPU Registers

Register	Description
R0-R12	General purpose registers R0-R12 have no special architecturally defined uses. Most instructions that specify a general purpose register specify R0-R12. ■ Low Registers: Registers R0-R7 are accessible by all instructions that specify a general purpose register. ■ High Registers: Registers R8-R12 are accessible by all 32-bit instructions that specify a general purpose register; they are not accessible by all 16-bit instructions.
R13	R13 is the stack pointer register. It is a banked register that switches between two 32-bit stack pointers: the main stack pointer (MSP) and the process stack pointer (PSP). The PSP is used only when the CPU operates at the user level in thread mode. The MSP is used in all other privilege levels and modes. Bits[0:1] of the SP are ignored and considered to be 0, so the SP is always aligned to a word (4 byte) boundary.
R14	R14 is the link register (LR). The LR stores the return address when a subroutine is called.
R15	R15 is the program counter (PC). Bit 0 of the PC is ignored and considered to be 0, so instructions are always aligned to a half word (2 byte) boundary.
xPSR	The program status registers are divided into three status registers, which are accessed either together or separately: ■ Application program status register (APSR) holds program execution status bits such as zero, carry, negative, in bits[27:31]. ■ Interrupt program status register (IPSR) holds the current exception number in bits[0:8]. ■ Execution program status register (EPSR) holds control bits for interrupt continuable and IF-THEN instructions in bits[10:15] and [25:26]. Bit 24 is always set to 1 to indicate Thumb mode. Trying to clear it causes a fault exception.
PRIMASK	A 1-bit interrupt mask register. When set, it allows only the nonmaskable interrupt (NMI) and hard fault exception. All other exceptions and interrupts are masked.
FAULTMASK	A 1-bit interrupt mask register. When set, it allows only the NMI. All other exceptions and interrupts are masked.
BASEPRI	A register of up to nine bits that define the masking priority level. When set, it disables all interrupts of the same or higher priority value. If set to 0 then the masking function is disabled.

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 powers down the CPU and other internal circuitry to reduce power consumption. However, supervisory services such as the central timewheel (CTW) remain available in this mode. The device can wake up using CTW or system reset. The wake up time from sleep mode is 125 µs (typical).

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 reset (XRES).

6.2.1.5 Wakeup Events

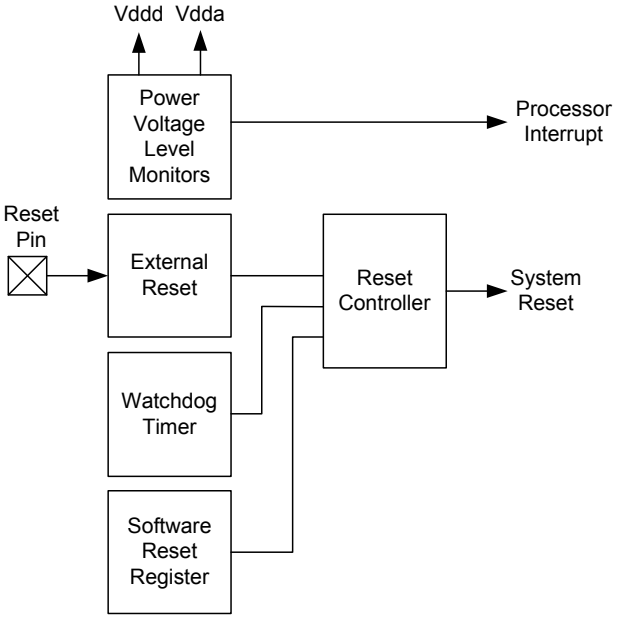
Wakeup events can come from the central timewheel or device reset. A wakeup event restores the system to active mode. The central timewheel allows the system to periodically wake up, poll peripherals, do voltage monitoring, or perform real-time functions. Reset event sources include the external reset pin (XRES).

6.3 Reset

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

- Power source monitoring: The analog and digital power voltages, V_{DDA}, V_{DDD}, V_{CCA}, and V_{CCD} are monitored in several different modes during power up and active mode. The monitors are programmable to generate an interrupt to the processor under certain conditions.
- 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. The watchdog timer can be used only when the part remains in active mode.
- Software: The device can be reset under program control.

Figure 6-6. Resets



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

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

6.3.1 Power Voltage Level Monitors

- IPOR - Initial Power on Reset

  At initial power on, IPOR monitors the power voltages V_{DDD} and V_{DDA}, both directly at the pins and at the outputs of the corresponding internal regulators. The trip level is not precise. It is set to approximately 1 volt, which is below the lowest specified operating voltage but high enough for the internal circuits to be reset and to hold their reset state. The monitor generates a reset pulse that is at least 100 ns wide. It may be much wider if one or more of the voltages ramps up slowly.

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

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

  Interrupt circuits are available to detect when V_{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 6-4.

Table 6-4. Analog/Digital Low Voltage Interrupt, Analog High Voltage Interrupt

Interrupt	Supply	Normal Voltage Range	Available Trip Settings
DLVI	V_{DDD}	2.7 V-5.5 V	2.45 V-5.45 V in 250 mV increments. The 2.45 V setting is used for LVD.
ALVI	V_{DDA}	2.7 V-5.5 V	2.45 V-5.45 V in 250 mV increments. The 2.45 V setting is used for LVD.
AHVI	V_{DDA}	2.7 V-5.5 V	5.75 V

  The monitors are disabled until after IPOR. The monitors are not available in low-power modes. To monitor voltages in sleep mode, wake up periodically using the CTW. After wakeup, the 2.45 V LVI interrupt may trigger. Voltage monitoring is not available in hibernate mode.

6.3.2 Other Reset Sources

- XRES - External Reset

  CY8C52 has a dedicated XRES pin which holds the part in reset while held active (low). The response to an XRES is the same as to an IPOR reset.

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

- SRES - Software Reset

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

  Another register bit exists to disable this function.

- WRES - Watchdog Timer Reset

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

  Note IPOR disables the watchdog function. The program must enable the watchdog function at an appropriate point in the code by setting a register bit. When this bit is set, it cannot be cleared again except by an IPOR power on reset event. The watchdog timer can be used only when the part remains in active mode.

6.4 I/O System and Routing

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

7.1.2 Example Analog Components

The following is a sample of the analog components available in PSoC Creator for the CY8C52 family. The exact amount of hardware resources (SC/CT blocks, routing, RAM, flash) used by a component varies with the features selected in PSoC Creator for the component.

- ADC
  - Successive Approximation (SAR ADC)
- DACs
  - Current
  - Voltage
  - PWM
- Comparators

7.1.3 Example System Function Components

The following is a sample of the system function components available in PSoC Creator for the CY8C52 family. The exact amount of hardware resources (UDBs, routing, RAM, flash) used by a component varies with the features selected in PSoC Creator for the component.

- CapSense
- LCD drive
- LCD control
- Filters

7.1.4 Designing with PSoC Creator

7.1.4.1 More Than a Typical IDE

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

PSoC Creator is that design tool.

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

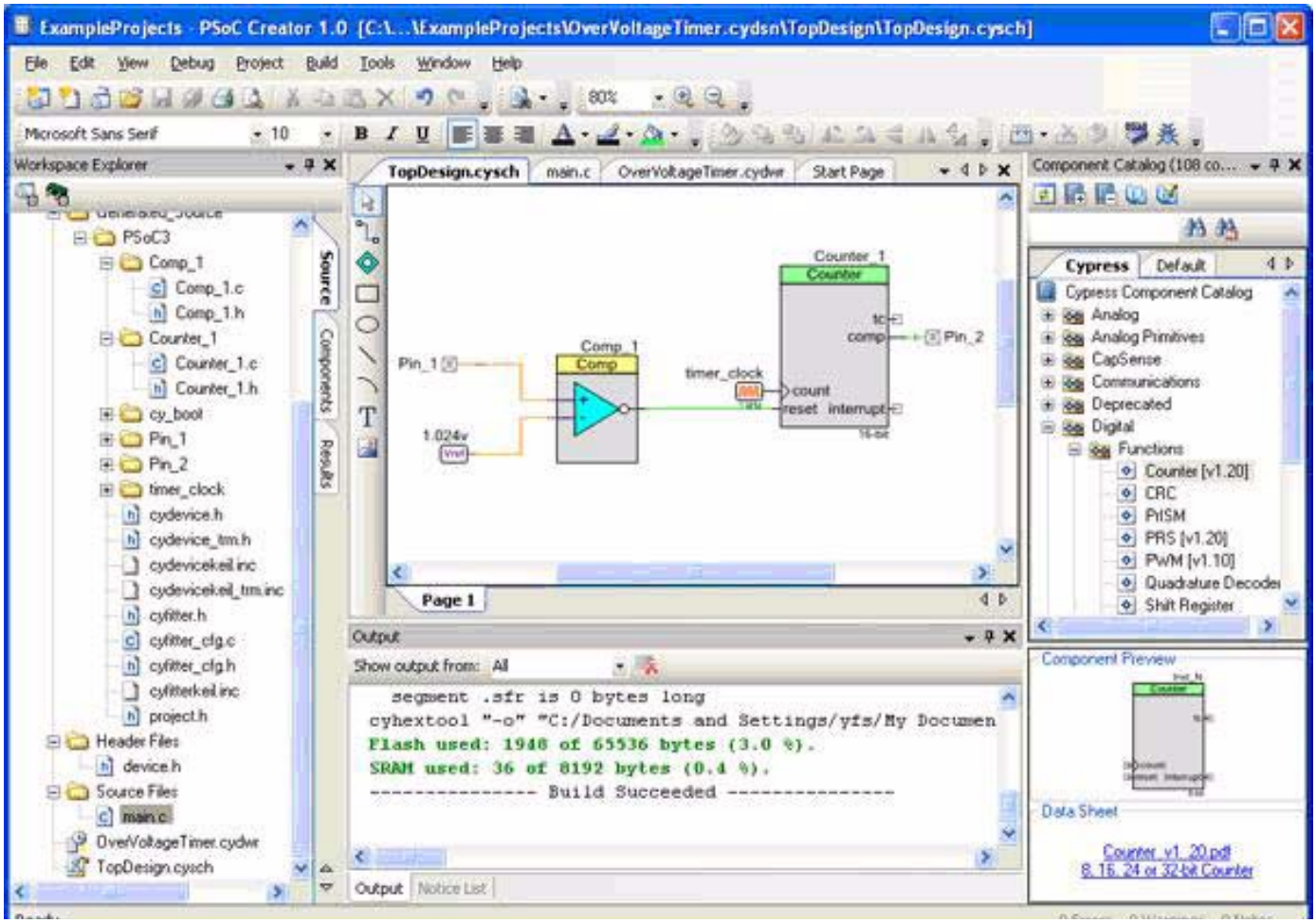
Graphical design entry simplifies the task of configuring a particular part. You can select the required functionality from an extensive catalog of components and place it in your design. All components are parameterized and have an editor dialog that allows you to tailor functionality to your needs.

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

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

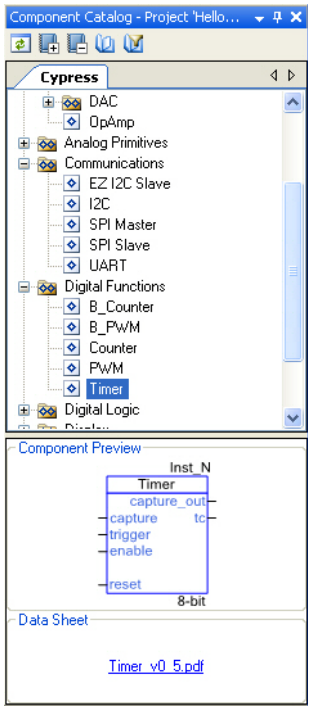
You also have the ability to change the C compiler and evaluate an alternative. Components are designed for portability and are validated against all devices, from all families, and against all supported tool chains. Switching compilers is as easy as editing from the project options and rebuilding the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

Figure 7-2. PSoC Creator Framework



7.1.4.2 Component Catalog

Figure 7-3. Component Catalog



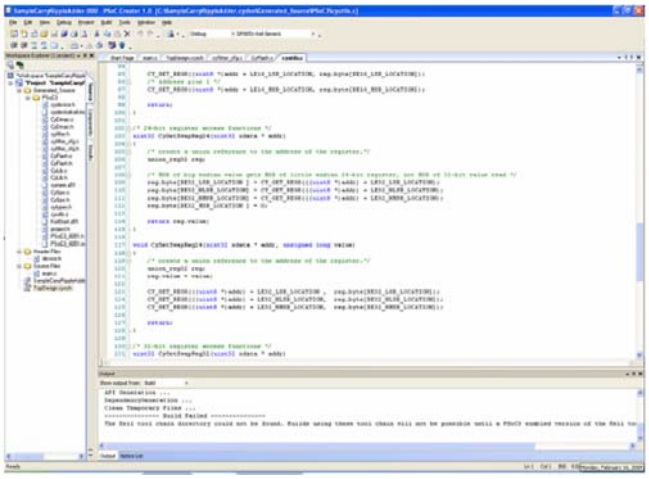
The component catalog is a repository of reusable design elements that select device functionality and customize your PSoC device. It is populated with an impressive selection of content; from simple primitives such as logic gates and device registers, through the digital timers, counters and PWMs, plus analog components such as ADC and DAC, and communication protocols such as I^2C and USB. See "Example Peripherals" section on page 32 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

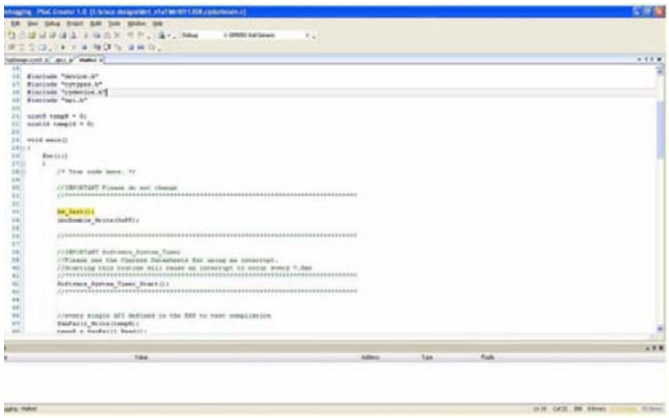


Anchoring the tool is a modern, highly customizable user interface. It includes project management and integrated editors for C and assembler source code, as well the design entry tools.

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

7.2.2.8 Conditionals

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

7.2.2.9 Variable MSB

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

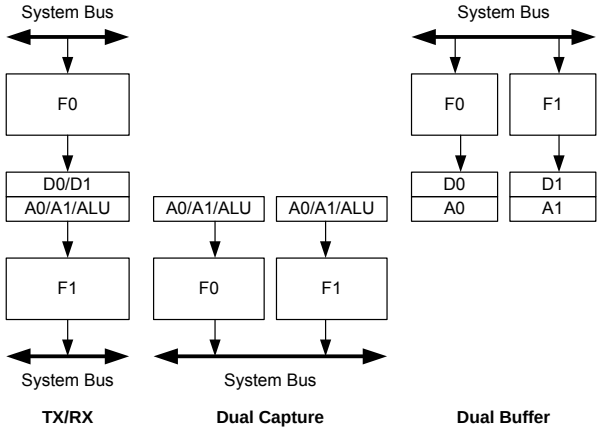
7.2.2.10 Built in CRC/PRS

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

7.2.2.11 Input/Output FIFOs

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

Figure 7-9. Example FIFO Configurations



7.2.2.12 Chaining

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

7.2.2.13 Time Multiplexing

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

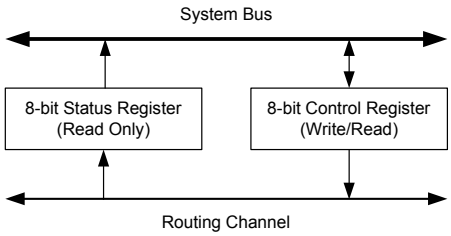
7.2.2.14 Datapath I/O

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

7.2.3 Status and Control Module

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

Figure 7-10. Status and Control Registers



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

7.2.3.15 Usage Examples

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

7.2.3.16 Clock Generation

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

8. Analog Subsystem

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

- Flexible, configurable analog routing architecture provided by analog globals, analog mux bus, and analog local buses
- Successive approximation (SAR) ADC
- One 8-bit DAC that provides either voltage or current output
- Two comparators with optional connection to configurable LUT outputs
- CapSense subsystem to enable capacitive touch sensing
- Precision reference for generating an accurate analog voltage for internal analog blocks

Figure 8-1. Analog Subsystem Block Diagram

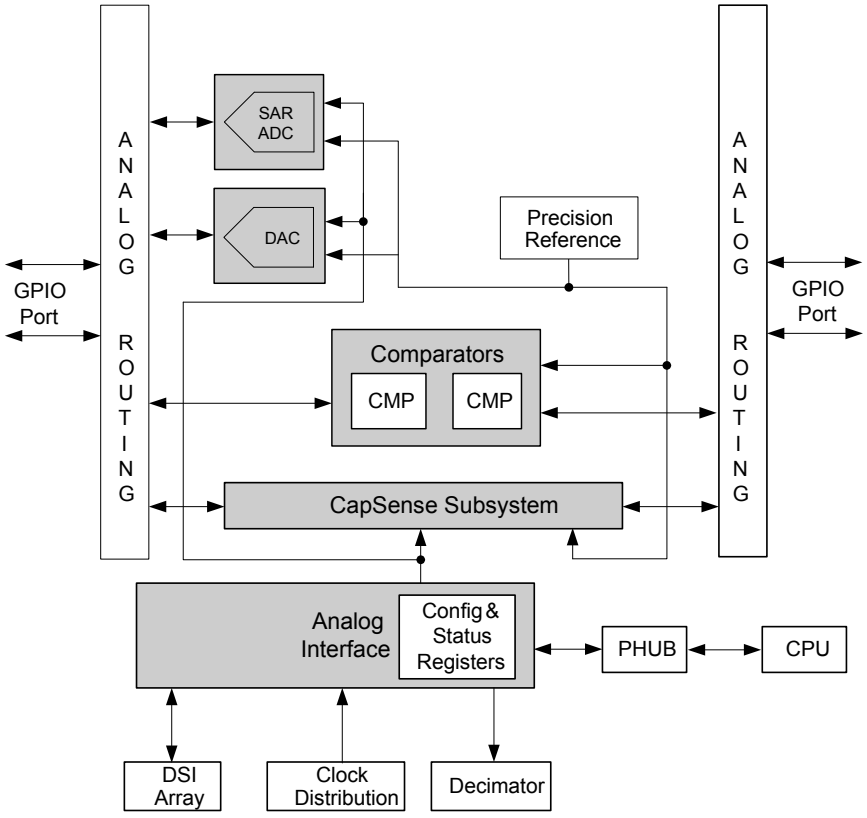
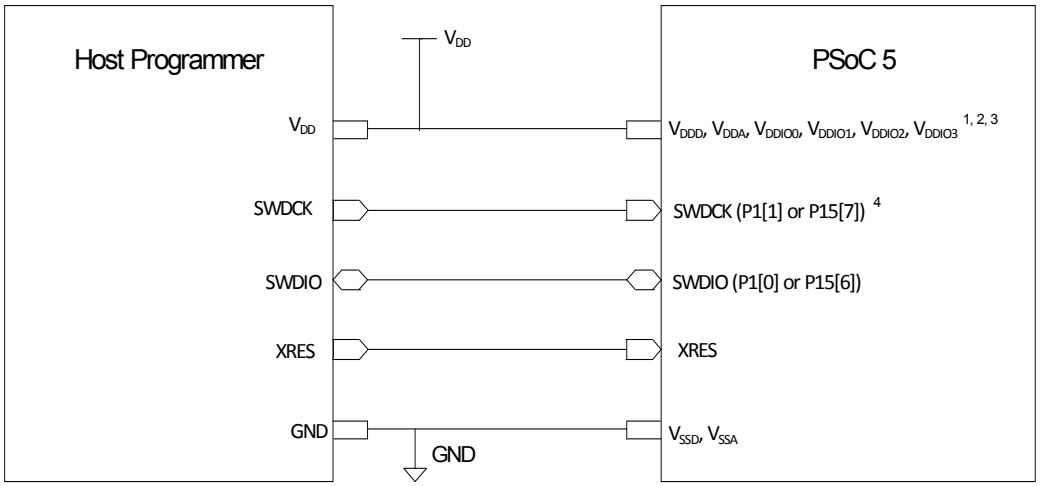


Figure 9-1. SWD Interface Connections between PSoC 5 and Programmer



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

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

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

[4] When USB SWD pins are used for Programming, the P1[1] SWDCK pin must be externally connected to Ground using external pull-down resistor (around 100 K resistor). This is required for P15[7] SWDCK signal to be seen by PSoC 5's internal logic.

9.3 Debug Features

The CY8C52 supports the following debug features:

- Halt and single-step the CPU
- View and change CPU and peripheral registers, and RAM addresses
- Six program address breakpoints and two literal access breakpoints
- Data watchpoint events to CPU
- Patch and remap instruction from flash to SRAM
- Debugging at the full speed of the CPU
- Compatible with PSoC Creator and MiniProg3 programmer and debugger

9.4 Trace Features

The following trace features are supported:

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

9.5 SWV Interface

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

9.6 Programming Features

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

9.7 Device Security

PSoC 5 offers an advanced security feature called device security, which permanently disables all test, programming, and debug ports, protecting your application from external access. The device security is activated by programming a 32-bit key (0x50536F43) to a Write Once Latch (WOL). The WOL must be programmed at $V_{DDD} \leq 3.3$ V and $T_J = 25$ °C ±15 °C.

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

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

You can write the key into the WOL to lock out external access only if no flash protection is set (see “Flash Security” section on page 16). However, after setting the values in the WOL, you still have access to the part until it is reset. Therefore, you can write the key into the WOL, program the flash protection data, and then reset the part to lock it.

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

Disclaimer

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

Cypress products meet the specifications contained in their particular Cypress data sheets. Cypress believes that its family of products is one of the most secure families of its kind on the market today, regardless of how they are used. There may be methods, unknown to Cypress, that can breach the code protection features. Any of these methods, to our knowledge, would be dishonest and possibly illegal. Neither Cypress nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Cypress is willing to work with the customer who is concerned about the integrity of their code. Code protection is constantly evolving. We at Cypress are committed to continuously improving the code protection features of our products.

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 2.7 V to 5.5 V, except where noted.

11.2.1 Device Level Specifications

Table 11-2. DC Specifications

Parameter	Description	Conditions				Min	Typ	Max	Units
V_{DDA}	Analog supply voltage and input to analog core regulator	Analog core regulator enabled				2.7	–	5.5	V
V_{DDD}	Digital supply voltage relative to V_{SSD}	Digital core regulator enabled				2.7	–	V_{DDA}[13]	V
V_{DDIO}[14]	I/O supply voltage relative to V_{SSIO}					1.71	–	V_{DDA}[13]	V
		Device Configuration	**V_{DDX}**	**F_{CPU}**	**Temp**				
I_{DD}[15]	Active Mode	Only IMO and CPU clock enabled. CPU executing simple loop from cache	2.7 V to 5.5 V	6 MHz	–40 °C	–	2.2	3	mA
					25 °C	–	2.4	3.5	
					85 °C	–	2.8	3.5	
		IMO enabled, bus clock and CPU clock enabled. CPU executing complex program from flash	2.7 V to 5.5 V	3 MHz	–40 °C	–	3.4	4	
					25 °C	–	3.6	4.5	
					85 °C	–	4.2	5	
				6 MHz	–40 °C	–	5.6	6	
					25 °C	–	6	7	
					85 °C	–	6.6	7.5	
				12 MHz	–40 °C	–	10	11	
					25 °C	–	11	12	
					85 °C	–	12	13	
				24 MHz	–40 °C	–	17	19	
					25 °C	–	18	20	
					85 °C	–	20	22	
				48 MHz	–40 °C	–	31	34	
					25 °C	–	33	35	
					85 °C	–	36	39	
				63 MHz	–40 °C	–	36	39	
					25 °C	–	37	40	
					85 °C	–	40	44	

Notes

13. V_{DDD} and V_{DDA} must be brought up in synchronization with each other, that is, at the same rates and levels. V_{DDA} must be greater than or equal to all other supplies.
14. 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}.
15. The current consumption of additional peripherals that are implemented only in programmed logic blocks can be found in their respective data sheets, available in PSoC Creator, the integrated design environment. To estimate total current, find CPU current at frequency of interest and add peripheral currents for your particular system from the device data sheet and component data sheets.

Table 11-3. AC Specifications[18]

Parameter	Description	Conditions	Min	Typ	Max	Units
F_{CPU}	CPU frequency		DC	–	40.01	MHz
F_{BUSCLK}	Bus frequency		DC	–	40.01	MHz
Svdd	V_{DD} ramp rate		–	–	0.066	V/µs
$T_{STARTUP}$	Time from $V_{DDD}/V_{DDA}/V_{CCD}/V_{CCA}$ ≥ min operating voltage to CPU executing code at reset vector	No PLL used, IMO boot mode 12 MHz typ	–	45	80	µs
T_{SLEEP}	Wakeup from sleep – CTW timeout to beginning of execution of next CPU instruction		–	125	–	µs
T_{SLEEP_INT}	Sleep timer periodic wakeup interval		–	–	128	ms

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 2.7 V to 5.5 V, except where noted.

11.3.1 Digital Core Regulator

Table 11-4. Digital Core Regulator DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
V_{DDD}	Input voltage		2.7	–	5.5	V
V_{CCD}	Output voltage		–	1.80	–	V
	Regulator output capacitor[19]	±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 21	–	1	10	µF

Figure 11-2. Regulators V_{CC} vs V_{DD}

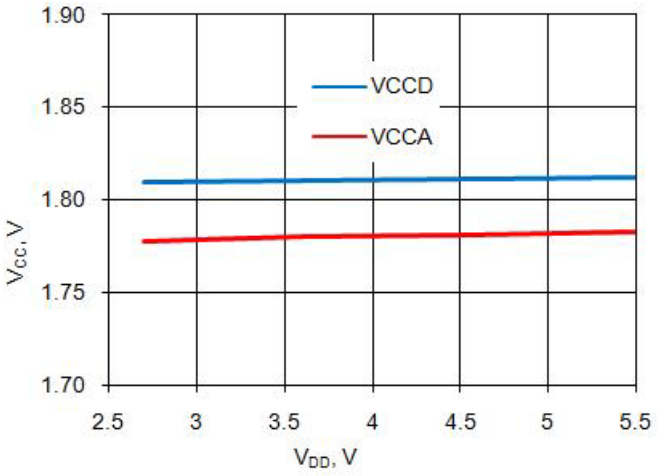
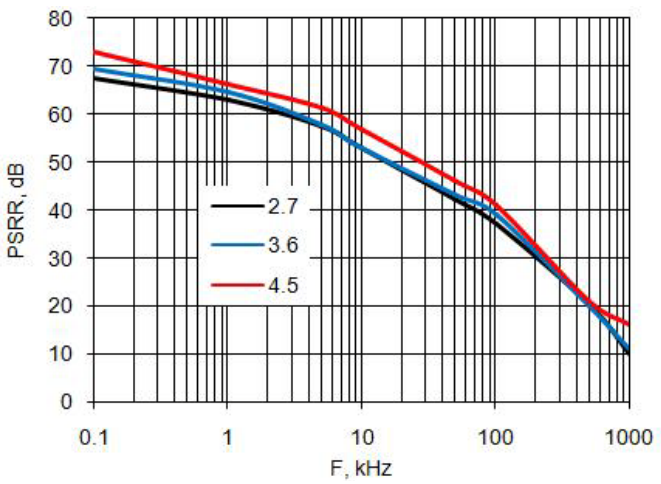


Figure 11-3. Digital Regulator PSRR vs Frequency and V_{DD}



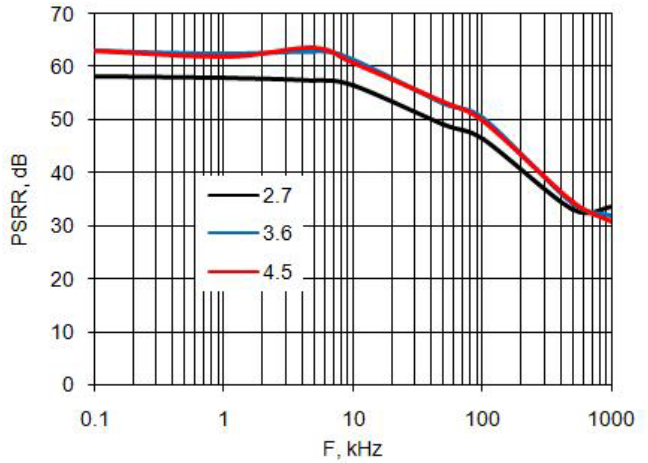
Notes
17. Sleep timer generates periodic interrupts to wake up the CPU. This specification applies only to those times that the CPU is off.
18. Based on device characterization (Not production tested).
19. 10 µF is required for sleep mode. See Table 11-3.

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		2.7	–	5.5	V
V_{CCA}	Output voltage		–	1.80	–	V
	Regulator output capacitor[22]	±10%, X5R ceramic or better	–	1	10	µF

Figure 11-4. Analog Regulator PSRR vs Frequency and V_{DD}



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 2.7 V to 5.5 V, except where noted. Unless otherwise specified, all charts and graphs show typical values.

11.4.1 GPIO

Table 11-6. GPIO DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
V_{IH}	Input voltage high threshold	CMOS Input, PRT[x]CTL = 0	0.7 × V_{DDIO}	–	–	V
V_{IL}	Input voltage low threshold	CMOS Input, PRT[x]CTL = 0	–	–	0.3 × V_{DDIO}	V
V_{IH}	Input voltage high threshold	LVTTL Input, PRT[x]CTL = 1	2.0	–	–	V
V_{IL}	Input voltage low threshold	LVTTL Input, PRT[x]CTL = 1	–	–	0.8	V
V_{OH}	Output voltage high	I_{OH} = 4 mA at 3.3 V_{DDIO}	V_{DDIO} – 0.6	–	–	V
V_{OL}	Output voltage low	I_{OL} = 8 mA at 3.3 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)[20]	25 °C, V_{DDIO} = 3.0 V	–	–	2	nA
C_{IN}	Input capacitance[20]	GPIOs not shared with kHzECO or SAR ADC external reference input	–	4	7	pF
		GPIOs shared with kHzECO[21]	–	5	7	pF
		GPIOs shared with SAR ADC external reference input	–	–	30	pF
V_H	Input voltage hysteresis (Schmitt-Trigger)[20]		–	150	–	mV

Notes
20. Based on device characterization (Not production tested).
21. For information on designing with PSoC 3 oscillators, refer to the application note, AN54439 - PSoC® 3 and PSoC 5 External Oscillator.
22. 10 µF is required for sleep mode. See Table 11-3.

Table 11-6. GPIO DC Specifications (continued)

Parameter	Description	Conditions	Min	Typ	Max	Units
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 11-5. GPIO Output High Voltage and Current

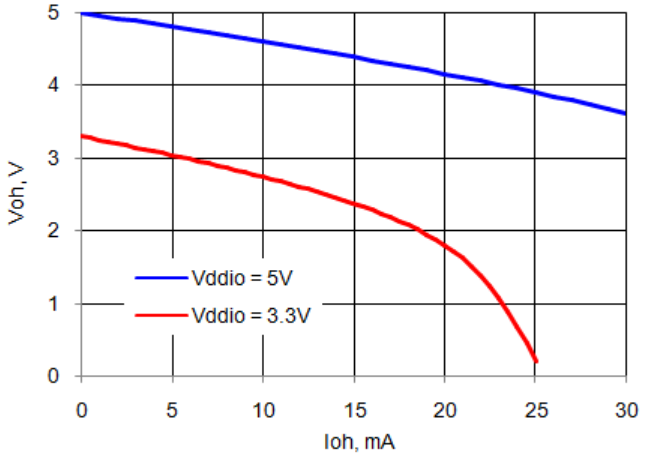


Figure 11-6. GPIO Output Low Voltage and Current

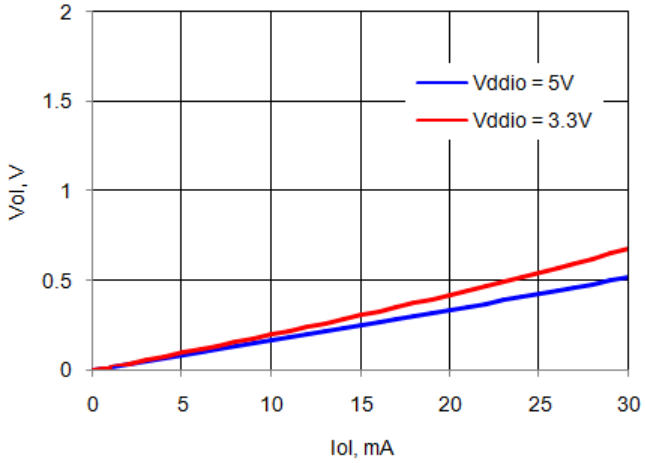


Table 11-11. USBIO AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
Tdrate	Full-speed data rate average bit rate	Using external 24 MHz crystal	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
		$V_{DDD} = 2.7\ V$	–	–	6	MHz
Tr_gpio	Rise time, GPIO mode, 10%/90% V_{DDD}	$V_{DDD} > 3\ V$, 25 pF load	–	–	12	ns
		$V_{DDD} = 2.7\ V$, 25 pF load	–	–	40	ns
Tf_gpio	Fall time, GPIO mode, 90%/10% V_{DDD}	$V_{DDD} > 3\ V$, 25 pF load	–	–	12	ns
		$V_{DDD} = 2.7\ V$, 25 pF load	–	–	40	ns

Figure 11-16. USBIO Output Rise and Fall Times, GPIO Mode, V_{DDD} = 3.3 V, 25 pF Load

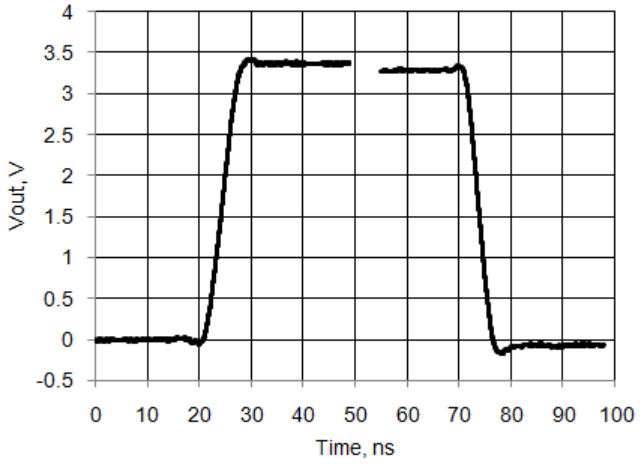


Figure 11-27. IDAC INL vs Temperature, Range = 255 µA, Fast Mode

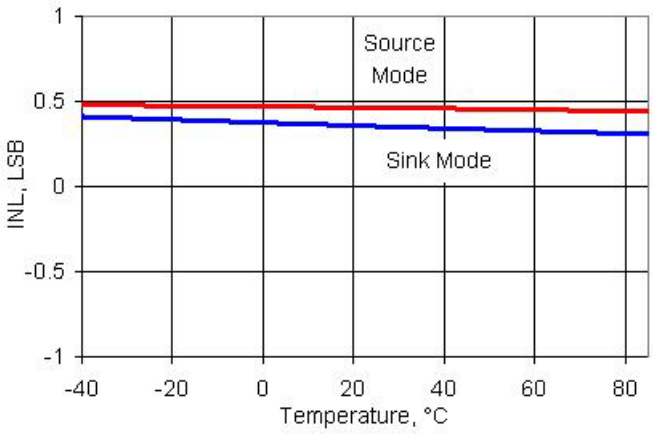


Figure 11-28. IDAC Full Scale Error vs Temperature, Range = 255 µA, Source Mode

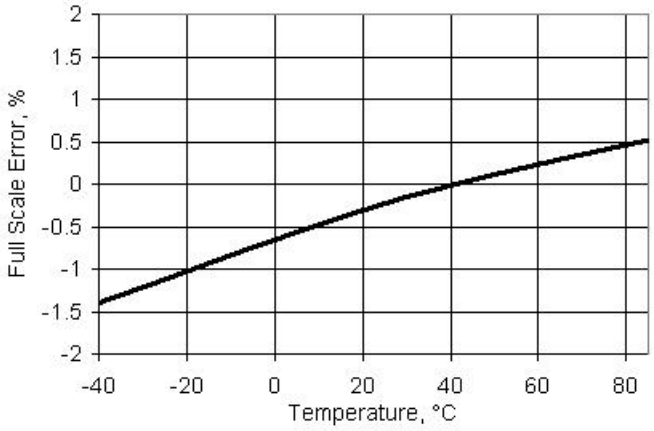


Figure 11-29. IDAC Operating Current vs Temperature, Range = 255 µA, Code = 0, Source Mode

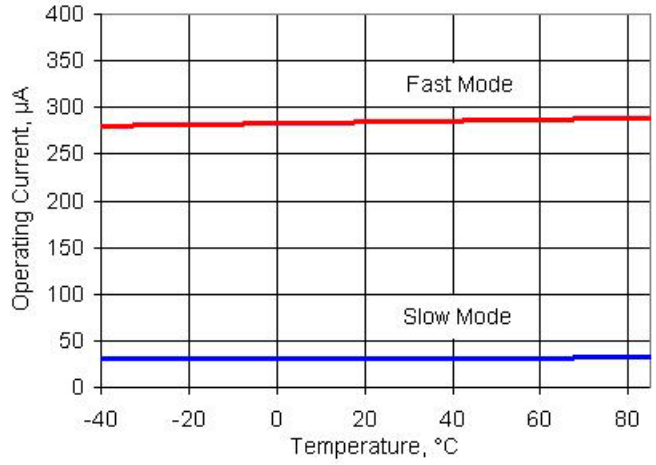


Figure 11-30. IDAC DNL vs Temperature, Range = 255 µA, Fast Mode

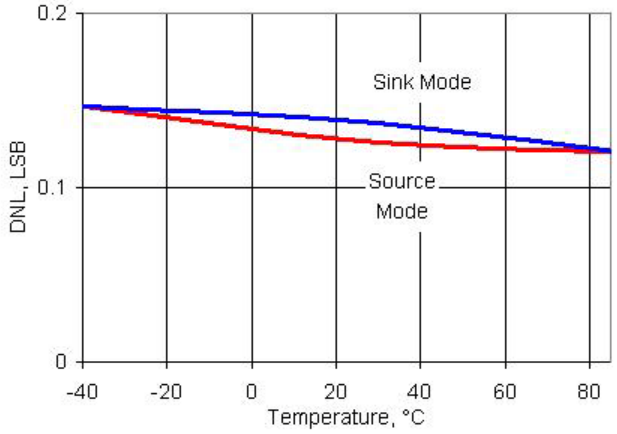


Figure 11-31. IDAC Full Scale Error vs Temperature, Range = 255 µA, Sink Mode

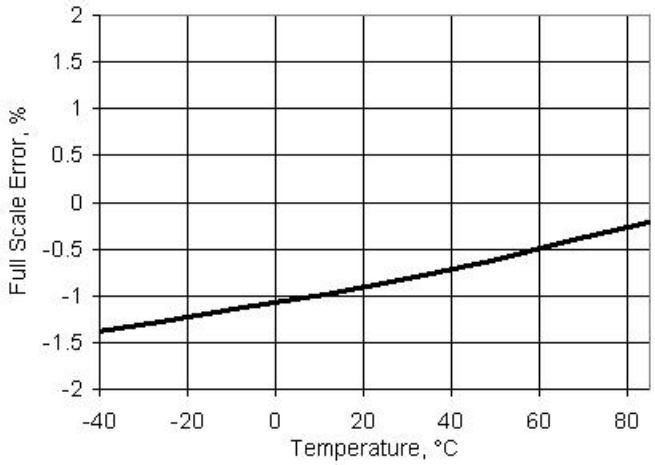
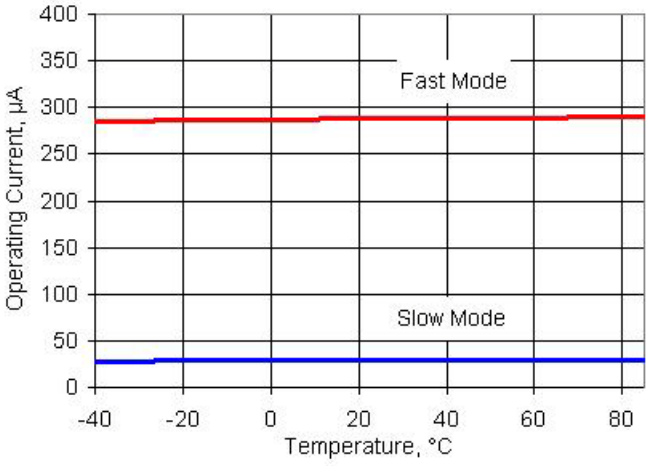


Figure 11-32. IDAC Operating Current vs Temperature, Range = 255 µA, Code = 0, Sink Mode



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 2.7 V to 5.5 V, except where noted.

11.6.1 Timer

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

Table 11-27. Timer DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
	16-bit timer block current consumption	Input clock frequency – 3 MHz	–	65	–	µA
		Input clock frequency –12 MHz	–	170	–	µA
		Input clock frequency – 40 MHz	–	650	–	µA

Table 11-28. Timer AC Specifications

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

11.6.2 Counter

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

Table 11-29. 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
	40 MHz		–	260	–	µA

Table 11-30. Counter AC Specifications

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

11.6.3 Pulse Width Modulation

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

Table 11-31. PWM DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
	16-bit PWM block current consumption	Input clock frequency – 3 MHz	–	65	–	µA
		Input clock frequency –12 MHz	–	170	–	µA
		Input clock frequency – 40 MHz	–	650	–	µA

Table 11-32. PWM AC Specifications

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

11.6.4 I^2C

Table 11-33. Fixed I^2C DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
	Block current consumption	Enabled, configured for 100 kbps	–	90	250	µA
		Enabled, configured for 400 kbps	–	100	250	µA

Table 11-34. Fixed I^2C AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
	Bit rate		–	–	400	Kbps

11.8.3 SWD Interface

Figure 11-50. SWD Interface Timing

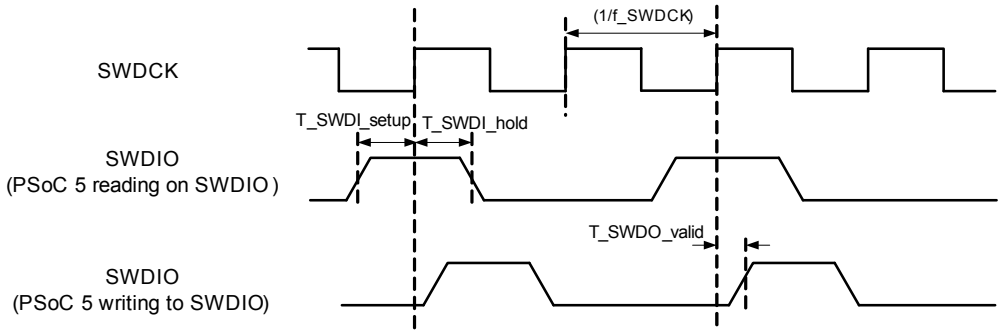


Table 11-47. SWD Interface AC Specifications[35]

Parameter	Description	Conditions	Min	Typ	Max	Units
f_SWDCK	SWDCLK frequency	3.3 V ≤ V_{DDD} ≤ 5 V	–	–	12[37]	MHz
		2.7 V ≤ V_{DDD} < 3.3 V	–	–	6.5[37]	MHz
		2.7 V ≤ V_{DDD} < 3.3 V, SWD over USBIO pins	–	–	5[37]	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	

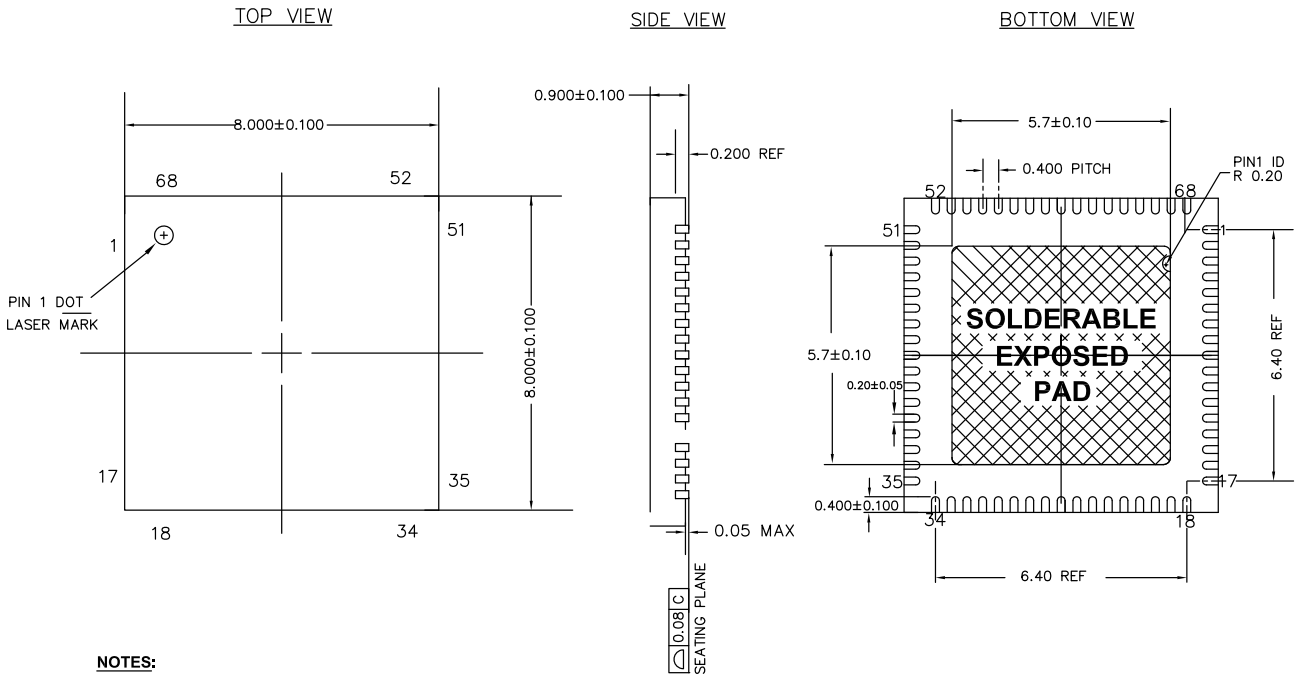
11.8.4 TPIU Interface

Table 11-48. TPIU Interface AC Specifications[35]

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

Notes
35. Based on device characterization (Not production tested).
36. f_TCK must also be no more than 1/3 CPU clock frequency.
37. f_SWDCK must also be no more than 1/3 CPU clock frequency.
38. SWV signal frequency and bit rate are limited by GPIO output frequency, see "GPIO AC Specifications" on page 59.

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

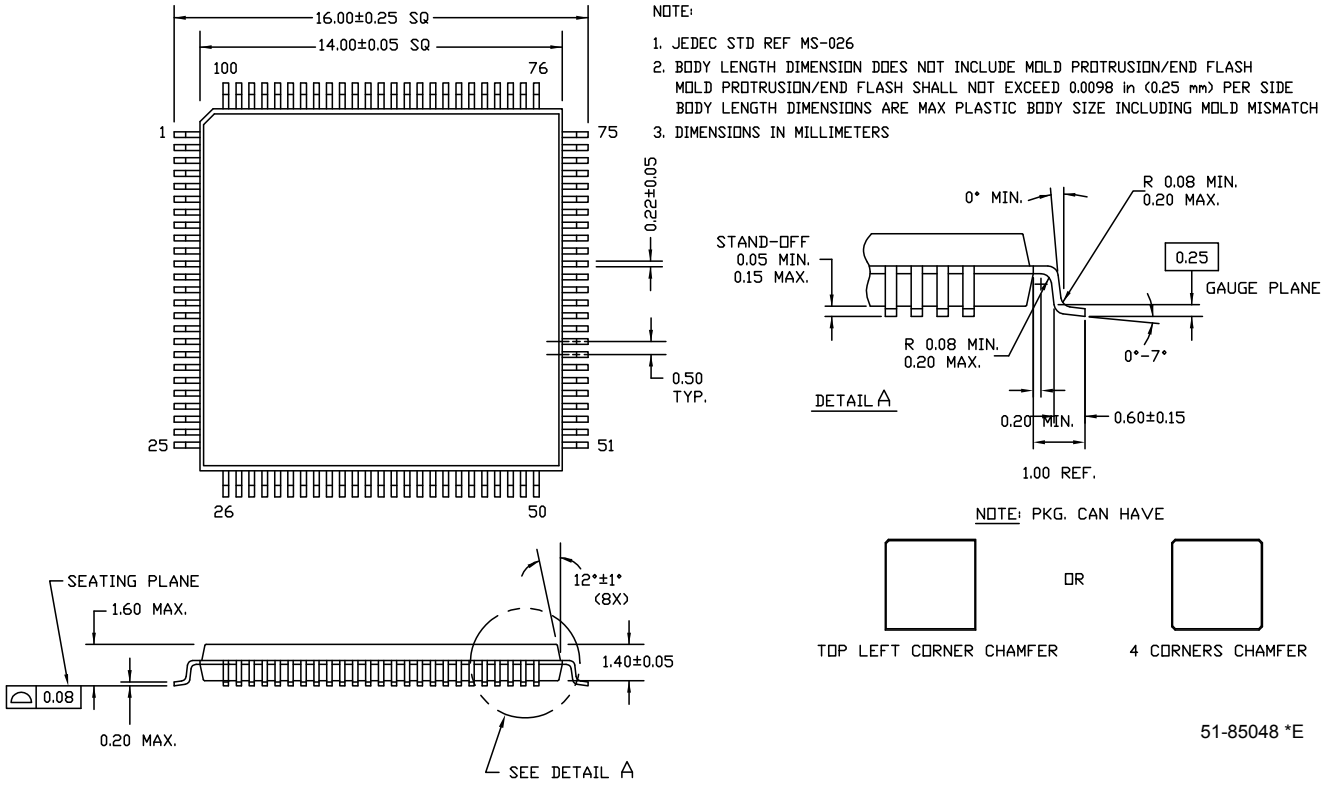


NOTES:

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

001-09618 *D

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



51-85048 *E