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"

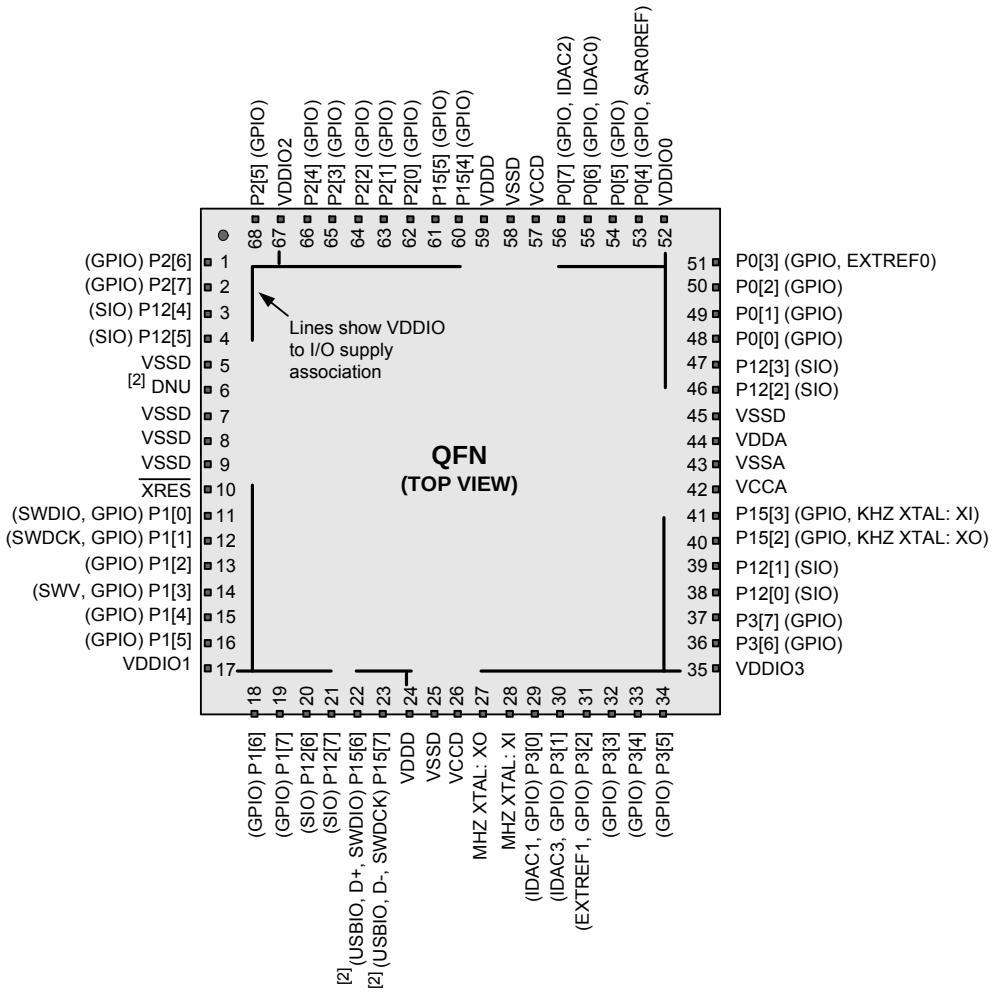
Details	
Product Status	Obsolete
Core Processor	ARM® Cortex®-M3
Core Size	32-Bit Single-Core
Speed	40MHz
Connectivity	I²C, LINbus, SPI, UART/USART, USB
Peripherals	CapSense, DMA, LCD, POR, PWM, WDT
Number of I/O	60
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	32K 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	100-LQFP
Supplier Device Package	100-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy8c5247axi-051

PSoC uses a SWD interface for programming, debug, and test. Using this standard interface enables the designer to debug or program the PSoC with a variety of hardware solutions from Cypress or third party vendors. The Cortex-M3 debug and trace modules include FPB, DWT, and ITM. These modules have many features to help solve difficult debug and trace problems. Details of the programming, test, and debugging interfaces are discussed in the "Programming, Debug Interfaces, Resources" section on page 49 of this data sheet.

2. Pinouts

The VDDIO pin that supplies a particular set of pins is indicated by the black lines drawn on the pinout diagrams in Figure 2-1 and Figure 2-2. Using the VDDIO pins, a single PSoC can support multiple interface voltage levels, eliminating the need for off-chip level shifters. Each VDDIO may sink up to 20 mA total to its associated I/O pins and opamps, and each set of VDDIO associated pins may sink up to 100 mA.

Figure 2-1. 68-pin QFN Part Pinout[1]



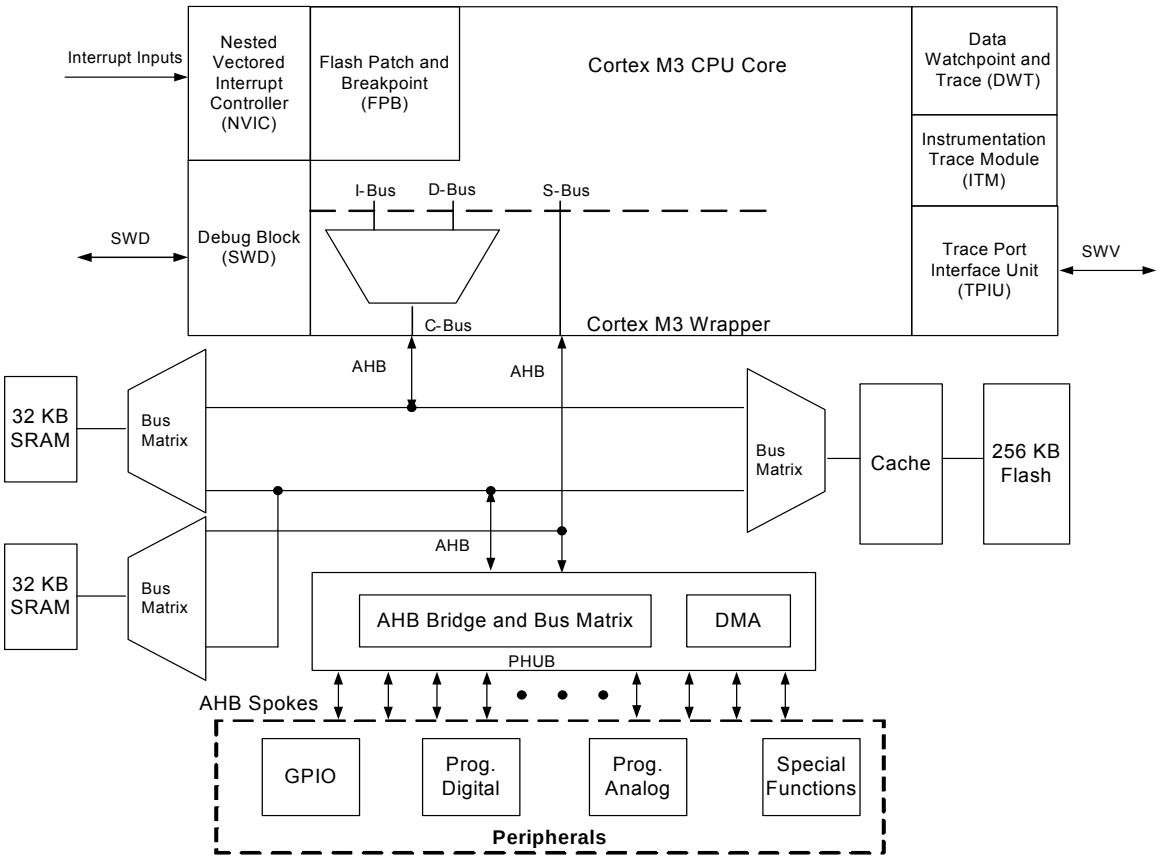
Notes
1. The center pad on the QFN package should be connected to digital ground (V_{SSD}) for best mechanical, thermal, and electrical performance. If not connected to ground, it should be electrically floated and not connected to any other signal.
2. Pins labeled Do Not Use (DNU) must be left floating.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

4. CPU

4.1 ARM Cortex-M3 CPU

The CY8C52 family of devices has an ARM Cortex-M3 CPU core. The Cortex-M3 is a low power 32-bit three-stage pipelined Harvard architecture CPU that delivers 1.25 DMIPS/MHz. It is intended for deeply embedded applications that require fast interrupt handling features.

Figure 4-1. ARM Cortex-M3 Block Diagram



The Cortex-M3 CPU subsystem includes these features:

- ARM Cortex-M3 CPU
- Programmable nested vectored interrupt controller (NVIC), tightly integrated with the CPU core
- Full-featured debug and trace module, tightly integrated with the CPU core
- Up to 256 KB of flash memory, 2 KB of EEPROM, and 64 KB of SRAM
- Cache controller with 128 bytes of memory
- Peripheral HUB (PHUB)
- DMA controller

4.1.1 Cortex-M3 Features

The Cortex-M3 CPU features include:

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

- The Thumb®-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.

4.3.4.6 Scatter Gather DMA

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

4.3.4.7 Packet Queuing DMA

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

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

4.3.4.8 Nested DMA

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

Table 4-6. Interrupt Vector Table (continued)

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

6. System Integration

6.1 Clocking System

The clocking system generates, divides, and distributes clocks throughout the PSoC system. The IMO and PLL together can generate up to a 40 MHz clock, accurate to ±5% over voltage and temperature. Additional internal and external clock sources allow each design to optimize accuracy, power, and cost. All of the system clock sources can be used to generate other clock frequencies in the 16-bit clock dividers and UDBs for anything you want, for example a UART baud rate generator.

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

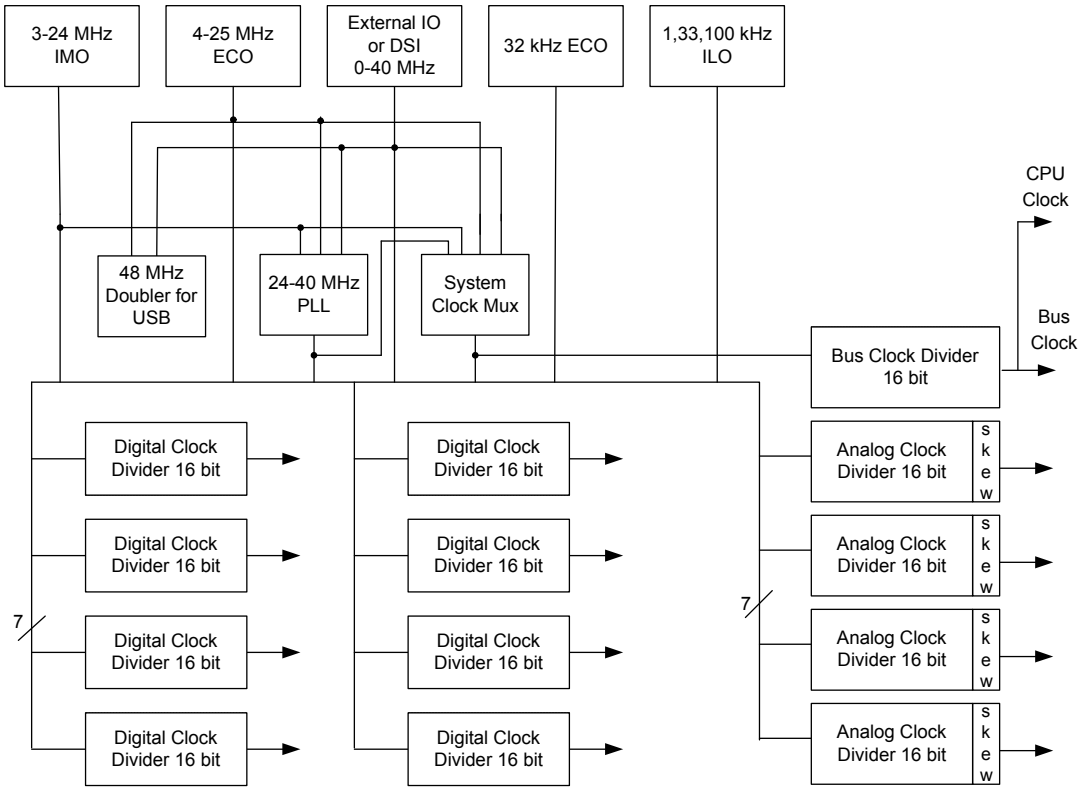
Key features of the clocking system include:

- Seven general purpose clock sources
  - 3 to 24 MHz IMO, ±5% at 3 MHz
  - 4 to 25 MHz external crystal oscillator (MHzECO)
  - Clock doubler provides a doubled clock frequency output for the USB block, see USB Clock Domain on page 21.
  - DSI signal from an external I/O pin or other logic
  - 24 to 40 MHz fractional phase-locked loop (PLL) sourced from IMO, MHzECO, or DSI
  - 1 kHz, 33 kHz, 100 kHz ILO for watchdog timer (WDT) and Sleep Timer
  - 32.768 kHz external crystal oscillator (ECO) for RTC
- Independently sourced clock dividers in all clocks
- Eight 16-bit clock dividers for the digital system
- Four 16-bit clock dividers for the analog system
- Dedicated 16-bit divider for the CPU bus and CPU clock
- Automatic clock configuration in PSoC Creator

Table 6-1. Oscillator Summary

Source	Fmin	Tolerance at Fmin	Fmax	Tolerance at Fmax	Startup Time
IMO	3 MHz	±5% over voltage and temperature	24 MHz	±8%	12 µs max
MHzECO	4 MHz	Crystal dependent	25 MHz	Crystal dependent	5 ms typ, max is crystal dependent
DSI	0 MHz	Input dependent	40 MHz	Input dependent	Input dependent
PLL	24 MHz	Input dependent	40 MHz	Input dependent	250 µs max
Doubler	48 MHz	Input dependent	48 MHz	Input dependent	1 µs max
ILO	1 kHz	–50%, +100%	100 kHz	–55%, +100%	15 ms max in lowest power mode
kHzECO	32 kHz	Crystal dependent	32 kHz	Crystal dependent	500 ms typ, max is crystal dependent

Figure 6-1. Clocking Subsystem



6.1.1 Internal Oscillators

6.1.1.1 Internal Main Oscillator

The IMO operates with no external components and outputs a stable clock. A factory trim for each frequency range is stored in the device. With the factory trim, tolerance varies from ±5% at 3 MHz, up to ±8% at 24 MHz. The IMO, in conjunction with the PLL, allows generation of CPU and system clocks up to the device's maximum frequency. The IMO provides clock outputs at 3, 6, 12, and 24 MHz.

6.1.1.2 Clock Doubler

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

6.1.1.3 Phase-Locked Loop

The PLL allows low frequency, high accuracy clocks to be multiplied to higher frequencies. This is a tradeoff between higher clock frequency and accuracy and, higher power consumption and increased startup time. The PLL block provides a mechanism for generating clock frequencies based upon a variety of input sources. The PLL outputs clock frequencies in the range of 24 to 40 MHz. Its input and feedback dividers supply 4032 discrete ratios to create almost any desired system clock frequency. The accuracy of the PLL output depends on the accuracy of the PLL input source. The most common PLL use is to multiply the IMO clock at 3 MHz, where it is most accurate, to generate the CPU and system clocks up to the device's maximum frequency.

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

6.1.1.4 Internal Low Speed Oscillator

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

The 1 kHz clock (CLK1K) is typically used for a background 'heartbeat' timer. This clock inherently lends itself to long sleep intervals using the central timewheel (CTW). The central timewheel is a free running counter clocked by the ILO 1 kHz output. The central timewheel is always enabled except in hibernate mode and when the CPU is stopped during debug on chip mode. It can be used to generate periodic interrupts for timing purposes or to wake the system from a low power mode. Firmware can reset the central timewheel.

The central timewheel can be programmed to wake the system periodically and optionally issue an interrupt. This enables flexible, periodic wakeups from low power modes or coarse timing applications. Systems that require accurate timing should use the RTC capability instead of the central timewheel. The 100 kHz clock (CLK100K) works as a low-power system clock to run the CPU. It can also generate fast time intervals using the fast timewheel.

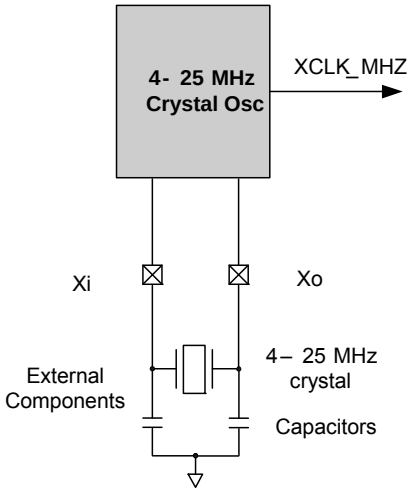
The fast timewheel is a 100 kHz, 5-bit counter clocked by the ILO that can also be used to generate periodic interrupts. The fast timewheel settings are programmable, and the counter automatically resets when the terminal count is reached. This enables flexible, periodic interrupts to the CPU at a higher rate than is allowed using the central timewheel. The fast timewheel can generate an optional interrupt each time the terminal count is reached. The 33 kHz clock (CLK33K) comes from a divide-by-3 operation on CLK100K. This output can be used as a reduced accuracy version of the 32.768 kHz ECO clock with no need for a crystal. The fast timewheel cannot be used as a wakeup source and must be turned off before entering sleep or hibernate mode.

6.1.2 External Oscillators

6.1.2.1 MHz External Crystal Oscillator

The MHzECO provides high frequency, high precision clocking using an external crystal (see Figure 6-2). It supports crystals in the range of 4 to 25 MHz. When used in conjunction with the PLL, it can generate CPU and system clocks up to the device's maximum frequency (see Phase-Locked Loop on page 19). The MHzECO with a 24 MHz crystal can be used with the clock doubler to generate a 48 MHz clock for the USB. If a crystal is not used then Xi must be shorted to ground and Xo must be left floating. MHzECO accuracy depends on the crystal chosen.

Figure 6-2. MHzECO Block Diagram

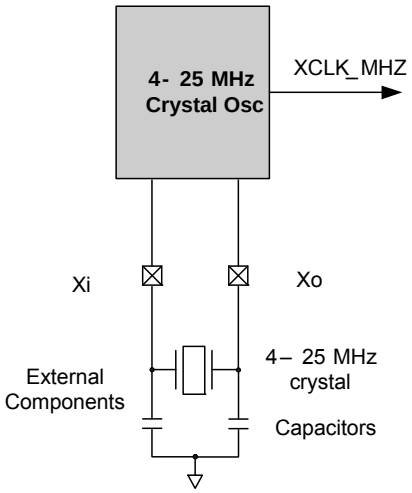


6.1.2.2 32.768 kHz ECO

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

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

Figure 6-3. 32kHzECO Block Diagram



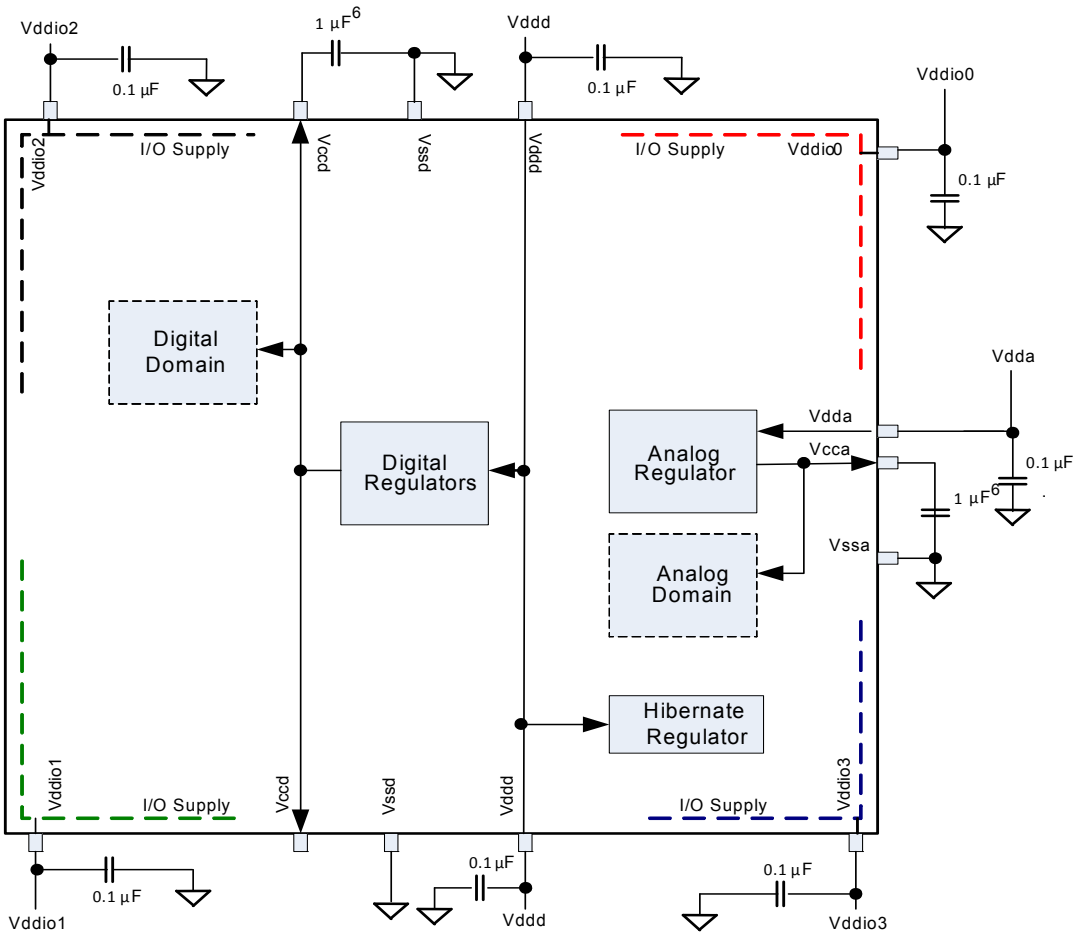
It is recommended that the external 32.768-kHz watch crystal have a load capacitance (CL) of 6 pF or 12.5 pF. Check the crystal manufacturer's datasheet. The two external capacitors, CL1 and CL2, are typically of the same value, and their total capacitance, CL1CL2 / (CL1 + CL2), including pin and trace capacitance, should equal the crystal CL value. For more information, refer to application note AN54439: PSoC 3 and PSoC 5 External Oscillators. See also pin capacitance specifications in the "GPIO" section on page 57.

6.1.2.3 Digital System Interconnect

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

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

Figure 6-4. PSoC Power System



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

Note
6. 10 µF is required for sleep mode. See Table 11-3.

6.4.18 Special Pin Functionality

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

- Digital
  - 32.768 kHz crystal oscillator
  - SWD and SWV interface pins
  - External reset
- Analog
  - High current IDAC output
  - External reference inputs

7. Digital Subsystem

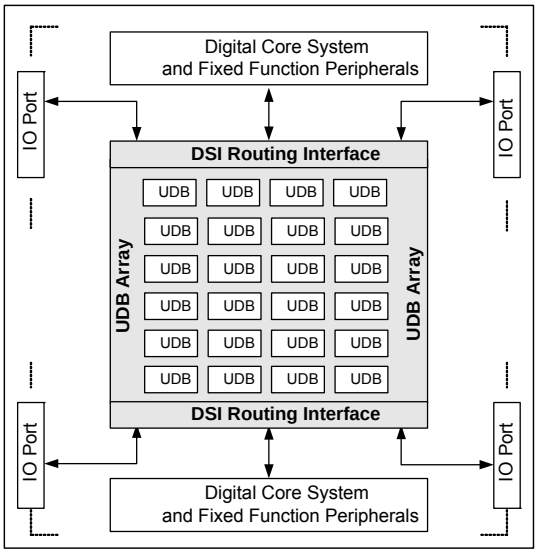
The digital programmable system creates application specific combinations of both standard and advanced digital peripherals and custom logic functions. These peripherals and logic are then interconnected to each other and to any pin on the device, providing a high level of design flexibility and IP security.

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

The main components of the digital programmable system are:

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

Figure 7-1. CY8C52 Digital Programmable Architecture



7.1 Example Peripherals

The flexibility of the CY8C52 family's UDBs and analog blocks allow you to create a wide range of components (peripherals). The most common peripherals were built and characterized by Cypress and are shown in the PSoC Creator component catalog. However, you may also create your own custom components using PSoC Creator. Using PSoC Creator, you may also create their own components for reuse within their organization, for example sensor interfaces, proprietary algorithms, and display interfaces.

The number of components available through PSoC Creator is too numerous to list in the data sheet, and the list is always growing. An example of a component available for use in CY8C52 family, but, not explicitly called out in this data sheet is the UART component.

7.1.1 Example Digital Components

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

- Communications
  - I^2C (1 to 3 UDBs)
  - UART (1 to 3 UDBs)
- Functions
  - PWM (1 to 2 UDBs)
- Logic (x CPLD product terms per logic function)
  - NOT
  - OR
  - XOR
  - AND

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

8.1 Analog Routing

The PSoC 5 family of devices has a flexible analog routing architecture that provides the capability to connect GPIOs and different analog blocks, and also route signals between different analog blocks. One of the strong points of this flexible routing architecture is that it allows dynamic routing of input and output connections to the different analog blocks. All analog routing switches are open when the device is in sleep or hibernate mode.

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

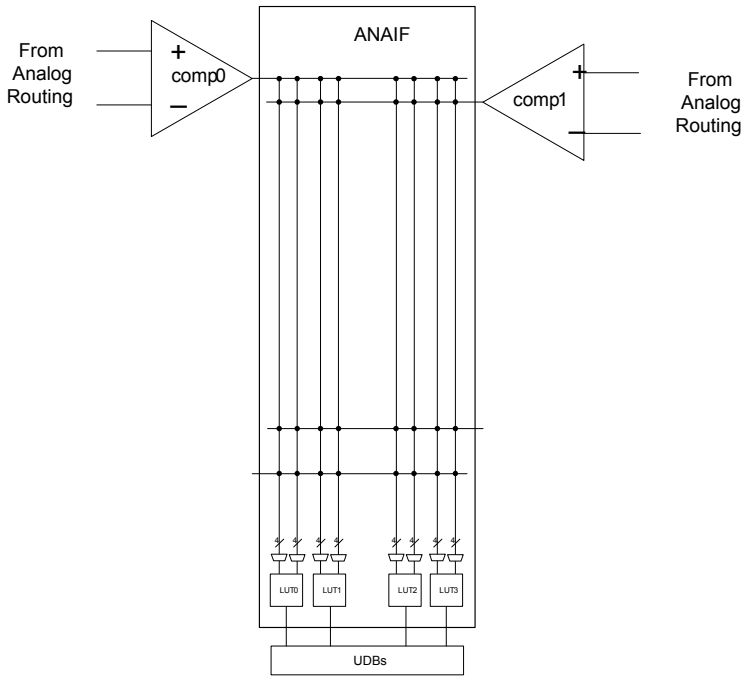
8.1.1 Features

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

8.1.2 Functional Description

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

Figure 8-4. Analog Comparator



8.3.2 LUT

The CY8C52 family of devices contains two LUTs. The LUT is a two input, one output lookup table that is driven by one or two of the comparators in the chip. The output of any LUT is routed to the digital system interface of the UDB array. From the digital system interface of the UDB array, these signals can be connected to UDBs, DMA controller, I/O, or the interrupt controller. The LUT control word written to a register sets the logic function on the output. The available LUT functions and the associated control word is shown in Table 8-1**.**

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

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

8.4 LCD Direct Drive

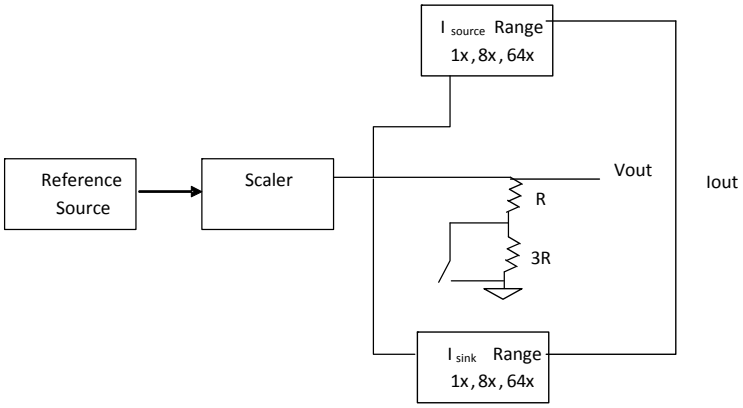
The PSoC LCD driver system is a highly configurable peripheral designed to allow PSoC to directly drive a broad range of LCD glass. All voltages are generated on chip, eliminating the need for external components. With a high multiplex ratio of up to 1/16, the CY8C52 family LCD driver system can drive a maximum of 736 segments. The PSoC LCD driver module was also designed with the conservative power budget of portable devices in mind, enabling different LCD drive modes and power down modes to conserve power.

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

Key features of the PSoC LCD segment system are:

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

Figure 8-6. DAC Block Diagram



8.7.1 Current DAC

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

8.7.2 Voltage DAC

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

9. Programming, Debug Interfaces, Resources

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

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

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

For more information on PSoC 5 programming, refer to the application note AN64359 - In-System Programming for PSoC® 5.

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

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

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

9.1 Debug Port Acquisition

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

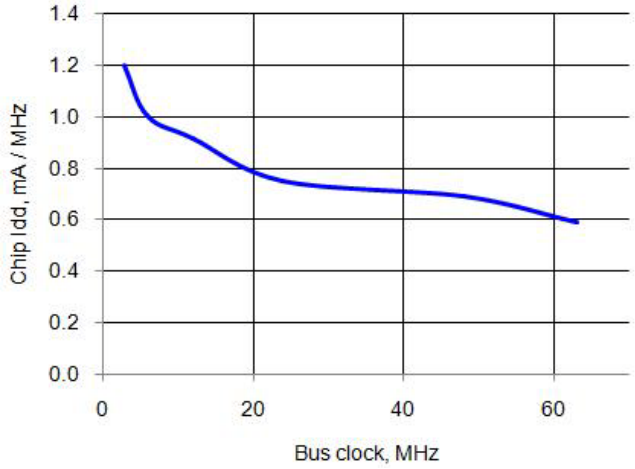
9.2 SWD Interface

SWD uses two pins, either two port 1 pins or the USBIO D+ and D- pins. The USBIO pins are useful for in system programming of USB solutions that would otherwise require a separate programming connector. One pin is used for the data clock (SWDCK) and the other is used for data input and output (SWDIO). SWD can be enabled on only one of the pin pairs at a time. When USB pins D+ and D- are used for SWD function, the SWDCK pin of port P1[1] is not available for use as a general purpose I/O and it should be externally pulled down using a resistor of less than 100 KΩ. SWD is used for debugging or for programming the flash memory. In addition, the SWD interface supports the SWV trace output. The SWD interface also includes the SWV interface, see "SWV Interface" on page 51. When using the SWD/SWV pins as standard GPIO, make sure that the GPIO functionality and PCB circuits do not interfere with SWD/SWV use. The SWV trace output is automatically activated whenever the SWD is activated.

Table 11-2. DC Specifications (continued)

Parameter	Description	Conditions			Min	Typ	Max	Units
		Device Configuration	$V_{DD} = V_{DDIO}$	Temp				
	Sleep Mode[17]	CPU = OFF SleepTimer=ON POR = ON	4.5 V to 5.5 V	–40 °C	–	1.4	–	µA
				25 °C	–	1.2	–	
				85 °C	–	11	–	
			2.7 V to 3.6 V	–40 °C	–	1.2	–	
				25 °C	–	2	–	
				85 °C	–	10	–	
	Hibernate Mode	All oscillators and regulators off, except hibernate regulator. SRAM retention	4.5 V to 5.5 V	–40 °C	–	0.3	–	µA
				25 °C	–	0.6	–	
				85 °C	–	10	–	
			2.7 V to 3.6 V	–40 °C	–	0.2	–	
				25 °C	–	0.3	–	
				85 °C	–	8	–	
I_{DDAR}	Analog current consumption while device is reset[16]	$V_{DDA} \leq 3.6$ V			–	0.3	–	mA
		$V_{DDA} > 3.6$ V			–	1.4	–	mA
I_{DDDR}	Digital current consumption while device is reset[16]	$V_{DDD} \leq 3.6$ V			–	1.1	–	mA
		$V_{DDD} > 3.6$ V			–	0.7	–	mA

Figure 11-1. Active Mode Device I_{DD}, mA/MHz



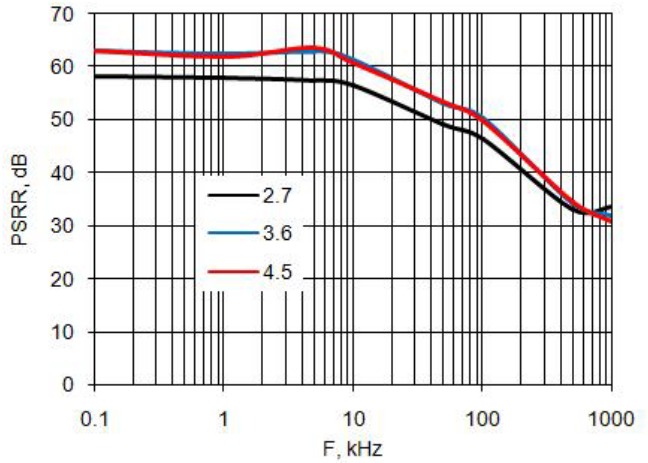
Note
16. Based on device characterization (not production tested). USBIO pins tied to ground (VSSD).

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.

11.5.3 Analog Globals

Table 11-18. Analog Globals DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
Rppag	Resistance pin-to-pin through analog global	V_{DDA} = 3.0 V	–	1200	1500	Ω
Rppmuxbus	Resistance pin-to-pin through analog mux bus	V_{DDA} = 3.0 V	–	700	1000	Ω

11.5.4 Comparator

Table 11-19. Comparator DC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
V_{OS}	Input offset voltage in fast mode	Factory trim, $V_{IN} \geq 0.5$ V	–		15	mV
	Input offset voltage in slow mode	Factory trim, $V_{IN} \geq 0.5$ V	–		15	mV
V_{OS}	Input offset voltage in ultra low power mode		–	±12	–	mV
V_{HYST}	Hysteresis	Hysteresis enable mode	–	10	32	mV
V_{ICM}	Input common mode voltage	High current / fast mode	V_{SSA}	–	V_{DDA} – 0.1	V
		Low current / slow mode	V_{SSA}	–	V_{DDA}	V
		Ultra low power mode	V_{SSA}	–	V_{DDA} – 0.9	
CMRR	Common mode rejection ratio		–	50	–	dB
I_{CMP}	High current mode/fast mode[31]		–	–	400	µA
	Low current mode/slow mode[31]		–	–	100	µA
	Ultra low power mode[31]		–	6	–	µA

Table 11-20. Comparator AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
T_{RESP}	Response time, high current mode[31]	50 mV overdrive, measured pin-to-pin	–	75	110	ns
	Response time, low current mode[31]	50 mV overdrive, measured pin-to-pin	–	155	200	ns
	Response time, ultra low power mode[31]	50 mV overdrive, measured pin-to-pin	–	55	–	µs

Figure 11-39. VDAC INL vs Temperature, 1 V Mode

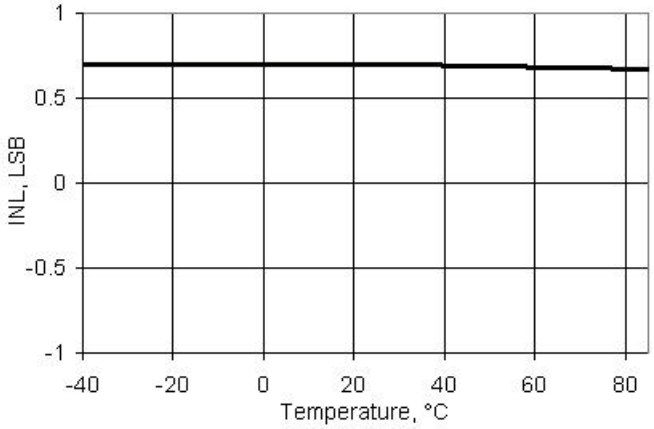


Figure 11-40. VDAC Full Scale Error vs Temperature, 1 V Mode

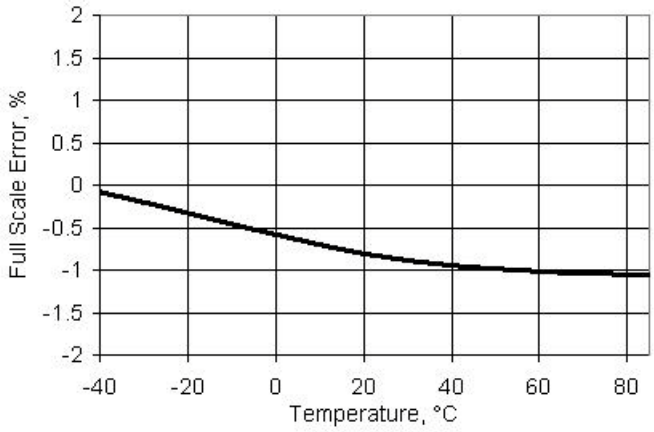


Figure 11-41. VDAC Operating Current vs Temperature, 1 V Mode, Slow Mode

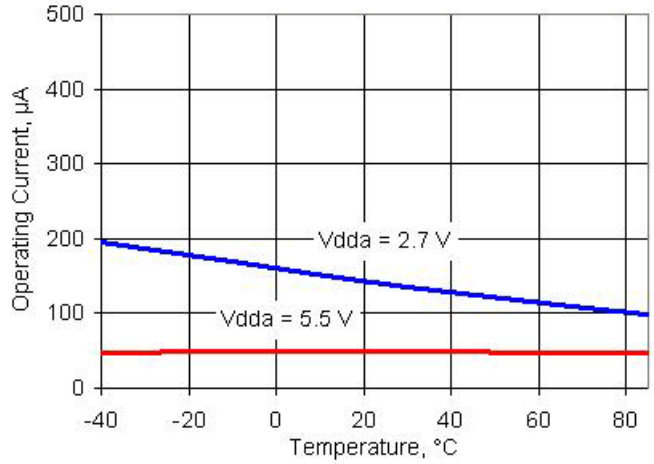


Figure 11-42. VDAC DNL vs Temperature, 1 V Mode

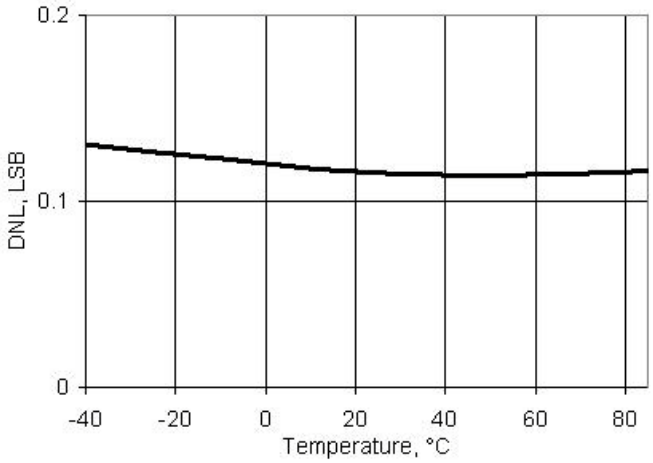


Figure 11-43. VDAC Full Scale Error vs Temperature, 4 V Mode

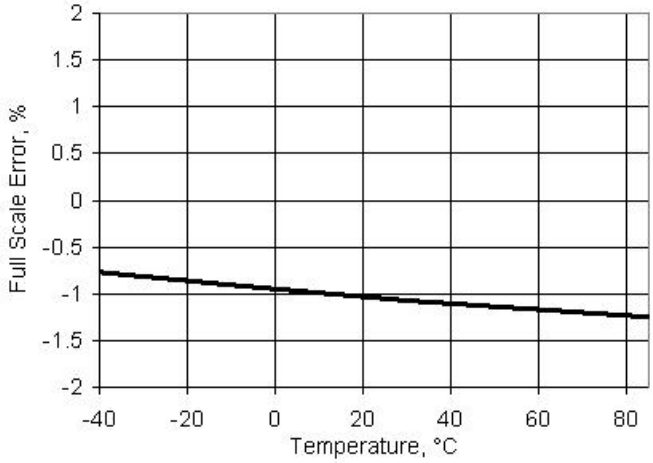


Figure 11-44. VDAC Operating Current vs Temperature, 1 V Mode, Fast Mode

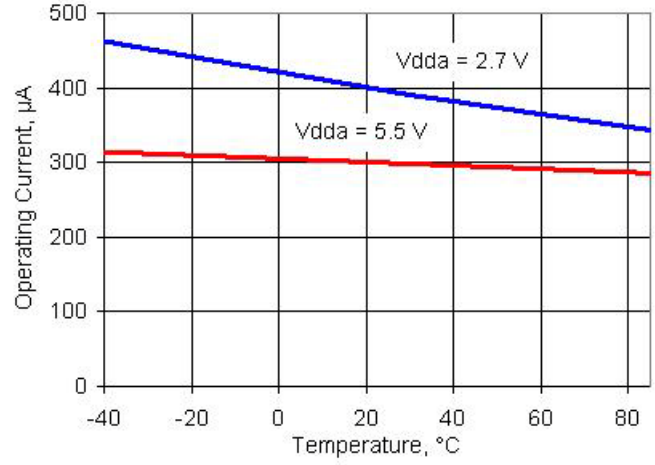


Figure 11-51. IMO Current vs. Frequency

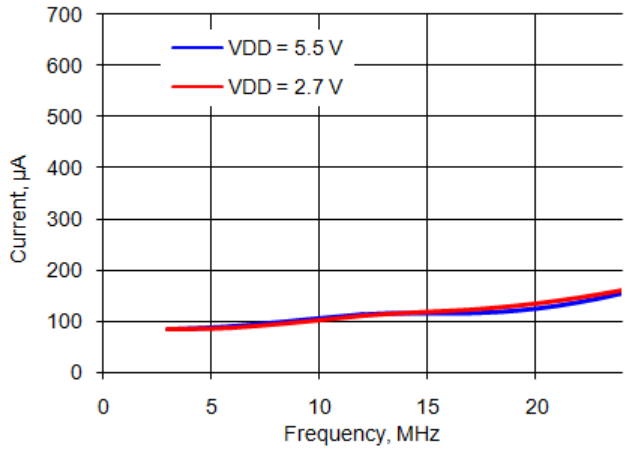


Table 11-52. IMO AC Specifications

Parameter	Description	Conditions	Min	Typ	Max	Units
F_{IMO}	IMO frequency stability (with factory trim)					
	24 MHz		–8	–	8	%
	12 MHz		–6.25	–	6.25	%
	6 MHz		–5.8	–	5.8	%
	3 MHz		–5	–	5	%
	Startup time[40]	From enable (during normal system operation) or wakeup from low power state	–	–	12	µs
Jp-p	Jitter (peak to peak)[40]					
	F = 24 MHz		–	0.5	–	ns
	F = 3 MHz		–	2.3	–	ns

Figure 11-52. IMO Frequency Variation vs. Temperature

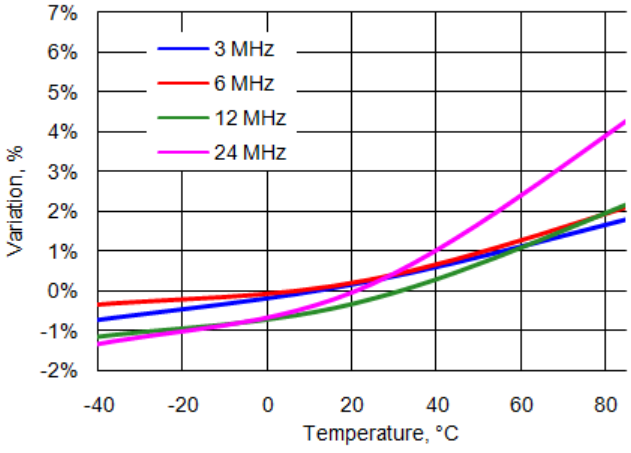
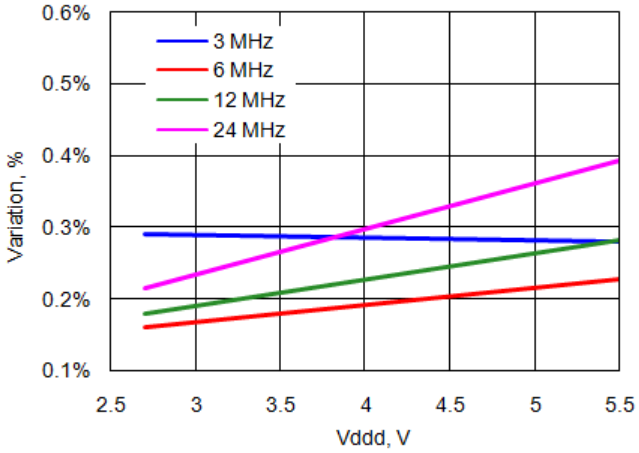
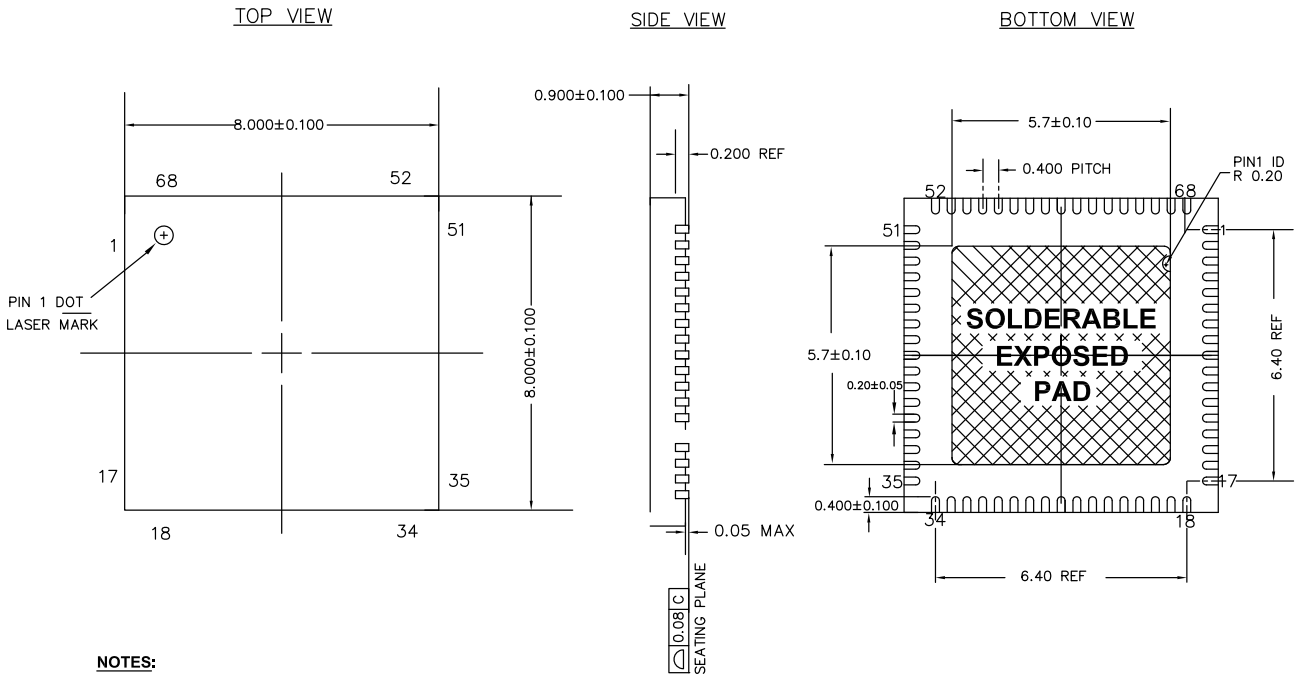


Figure 11-53. IMO Frequency Variation vs. V_{DDD}



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

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

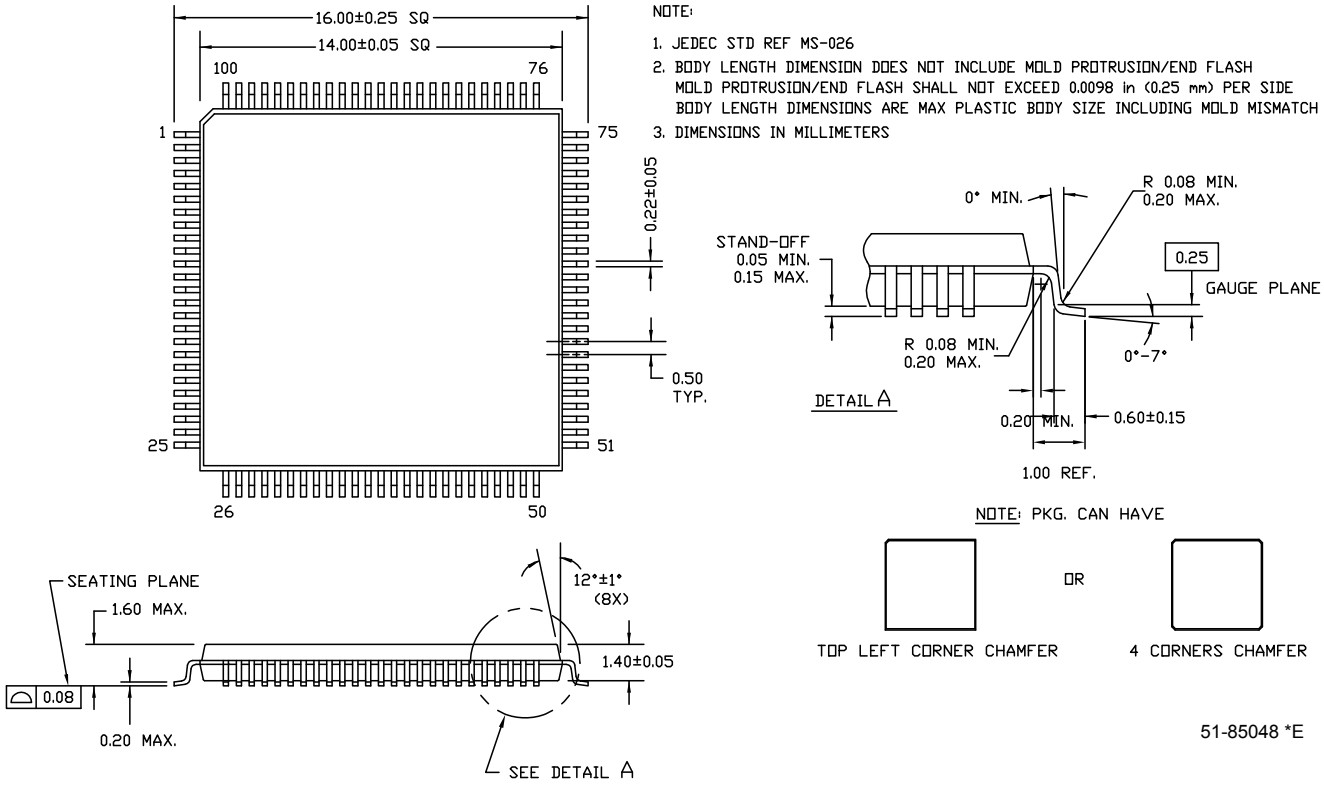


NOTES:

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

14. Acronyms

Table 14-1. Acronyms Used in this Document

Acronym	Description
abus	analog local bus
ADC	analog-to-digital converter
AG	analog global
AHB	AMBA (advanced microcontroller bus architecture) high-performance bus, an ARM data transfer bus
ALU	arithmetic logic unit
AMUXBUS	analog multiplexer bus
API	application programming interface
APSR	application program status register
ARM®	advanced RISC machine, a CPU architecture
ATM	automatic thump mode
BW	bandwidth
CMRR	common-mode rejection ratio
CPU	central processing unit
CRC	cyclic redundancy check, an error-checking protocol
DAC	digital-to-analog converter, see also IDAC, VDAC
DFB	digital filter block
DIO	digital input/output, GPIO with only digital capabilities, no analog. See GPIO.
DMA	direct memory access, see also TD
DNL	differential nonlinearity, see also INL
DNU	do not use
DR	port write data registers
DSI	digital system interconnect
DWT	data watchpoint and trace
ECO	external crystal oscillator
EEPROM	electrically erasable programmable read-only memory
EMI	electromagnetic interference
EOC	end of conversion
EOF	end of frame
EPSR	execution program status register
ESD	electrostatic discharge
FIR	finite impulse response, see also IIR
FPB	flash patch and breakpoint
FS	full-speed
GPIO	general-purpose input/output, applies to a PSoC pin

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

Acronym	Description
HVI	high-voltage interrupt, see also LVI, LVD
IC	integrated circuit
IDAC	current DAC, see also DAC, VDAC
IDE	integrated development environment
I^2C, or IIC	Inter-Integrated Circuit, a communications protocol
IIR	infinite impulse response, see also FIR
ILO	internal low-speed oscillator, see also IMO
IMO	internal main oscillator, see also ILO
INL	integral nonlinearity, see also DNL
I/O	input/output, see also GPIO, DIO, SIO, USBIO
IPOR	initial power-on reset
IPSR	interrupt program status register
IRQ	interrupt request
ITM	instrumentation trace macrocell
LCD	liquid crystal display
LIN	Local Interconnect Network, a communications protocol.
LR	link register
LUT	lookup table
LVD	low-voltage detect, see also LVI
LVI	low-voltage interrupt, see also HVI
LVTTL	low-voltage transistor-transistor logic
MAC	multiply-accumulate
MCU	microcontroller unit
MISO	master-in slave-out
NC	no connect
NMI	nonmaskable interrupt
NRZ	non-return-to-zero
NVIC	nested vectored interrupt controller
NVL	nonvolatile latch, see also WOL
opamp	operational amplifier
PAL	programmable array logic, see also PLD
PC	program counter
PCB	printed circuit board
PGA	programmable gain amplifier
PHUB	peripheral hub
PHY	physical layer
PICU	port interrupt control unit
PLA	programmable logic array