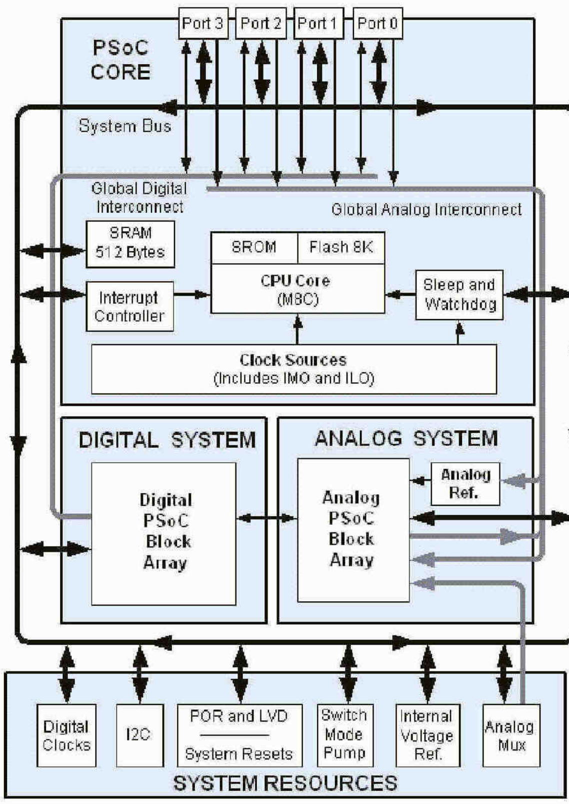
EXHIBIT 38

CY8C21234, CY8C21334, CY8C21434, CY8C21534, and CY8C21634

Features

- **Powerful Harvard Architecture Processor**
  - M8C Processor Speeds to 24 MHz
  - Low Power at High Speed
  - 2.4V to 5.25V Operating Voltage
  - Operating Voltages Down to 1.0V Using On-Chip Switch Mode Pump (SMP)
  - Industrial Temperature Range: -40°C to +85°C
- **Advanced Peripherals (PSoC Blocks)**
  - 4 Analog Type "E" PSoC Blocks Provide:
    - 2 Comparators with DAC Refs
    - Single or Dual 8-Bit 28 Channel ADC
  - 4 Digital PSoC Blocks Provide:
    - 8- to 32-Bit Timers, Counters, and PWMs
    - CRC and PRS Modules
    - Full-Duplex UART, SPI™ Master or Slave
    - Connectable to All GPIO Pins
  - Complex Peripherals by Combining Blocks
- **Flexible On-Chip Memory**
  - 8K Flash Program Storage 50,000 Erase/Write Cycles
  - 512 Bytes SRAM Data Storage
  - In-System Serial Programming (ISSP™)
  - Partial Flash Updates
  - Flexible Protection Modes
  - EEPROM Emulation in Flash
- **Complete Development Tools**
  - Free Development Software (PSoC™ Designer)
  - Full-Featured, In-Circuit Emulator and Programmer
  - Full Speed Emulation
  - Complex Breakpoint Structure
  - 128K Trace Memory
- **Precision, Programmable Clocking**
  - Internal ±2.5% 24/48 MHz Oscillator
  - Internal Oscillator for Watchdog and Sleep
- **Programmable Pin Configurations**
  - 25 mA Drive on All GPIO
  - Pull Up, Pull Down, High Z, Strong, or Open Drain Drive Modes on All GPIO
  - Up to 8 Analog Inputs on GPIO
  - Configurable Interrupt on All GPIO
- **Versatile Analog Mux**
  - Common Internal Analog Bus
  - Simultaneous Connection of IO Combinations
  - Capacitive Sensing Application Capability
- **Additional System Resources**
  - I²C™ Master, Slave and Multi-Master to 400 kHz
  - Watchdog and Sleep Timers
  - User-Configurable Low Voltage Detection
  - Integrated Supervisory Circuit
  - On-Chip Precision Voltage Reference



PSoC® Functional Overview

The PSoC® family consists of many *Mixed-Signal Array with On-Chip Controller* devices. These devices are designed to replace multiple traditional MCU-based system components with one, low cost single-chip programmable component. A PSoC device includes configurable blocks of analog and digital logic, as well as programmable interconnect. This architecture allows the user to create customized peripheral configurations, to match the requirements of each individual application. Additionally, a fast CPU, Flash program memory, SRAM data memory, and configurable IO are included in a range of convenient pinouts.

The PSoC architecture, as illustrated on the left, is comprised of four main areas: the Core, the System Resources, the Digital System, and the Analog System. Configurable global bus resources allow all the device resources to be combined into a complete custom system. Each CY8C21x34 PSoC device includes four digital blocks and four analog blocks. Depending on the PSoC package, up to 28 general purpose IO (GPIO) are also included. The GPIO provide access to the global digital and analog interconnects.

The PSoC Core

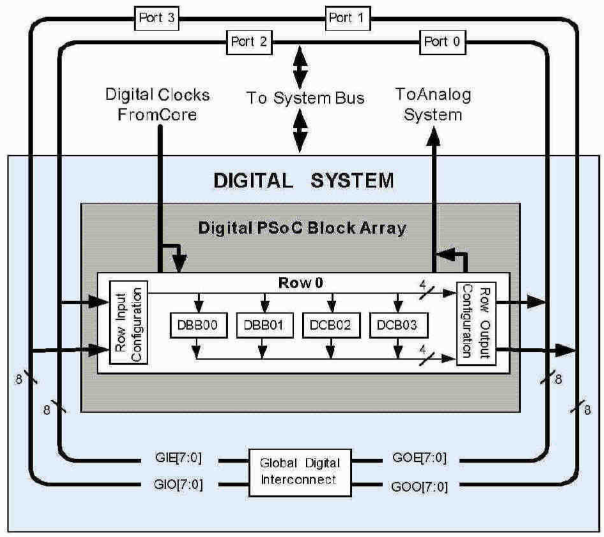
The PSoC Core is a powerful engine that supports a rich instruction set. It encompasses SRAM for data storage, an interrupt controller, sleep and watchdog timers, and IMO (internal main oscillator) and ILO (internal low speed oscillator). The

CPU core, called the M8C, is a powerful processor with speeds up to 24 MHz. The M8C is a four MIPS 8-bit Harvard architecture microprocessor.

System Resources provide additional capability, such as digital clocks to increase the flexibility of the PSoC mixed-signal arrays, I2C functionality for implementing an I2C master, slave, MultiMaster, an internal voltage reference that provides an absolute value of 1.3V to a number of PSoC subsystems, a switch mode pump (SMP) that generates normal operating voltages off a single battery cell, and various system resets supported by the M8C.

The Digital System is composed of an array of digital PSoC blocks, which can be configured into any number of digital peripherals. The digital blocks can be connected to the GPIO through a series of global buses that can route any signal to any pin. Freeing designs from the constraints of a fixed peripheral controller.

The Analog System is composed of four analog PSoC blocks, supporting comparators and analog-to-digital conversion up to 8 bits in precision.



Digital System Block Diagram

The Digital System

The Digital System is composed of 4 digital PSoC blocks. Each block is an 8-bit resource that can be used alone or combined with other blocks to form 8, 16, 24, and 32-bit peripherals, which are called user module references. Digital peripheral configurations include those listed below.

- PWMs (8 to 32 bit)
- PWMs with Dead band (8 to 32 bit)
- Counters (8 to 32 bit)
- Timers (8 to 32 bit)
- UART 8 bit with selectable parity
- SPI master and slave
- I2C slave and multi-master
- Cyclical Redundancy Checker/Generator (8 to 32 bit)
- IrDA
- Pseudo Random Sequence Generators (8 to 32 bit)

The digital blocks can be connected to any GPIO through a series of global buses that can route any signal to any pin. The buses also allow for signal multiplexing and for performing logic operations. This configurability frees your designs from the constraints of a fixed peripheral controller.

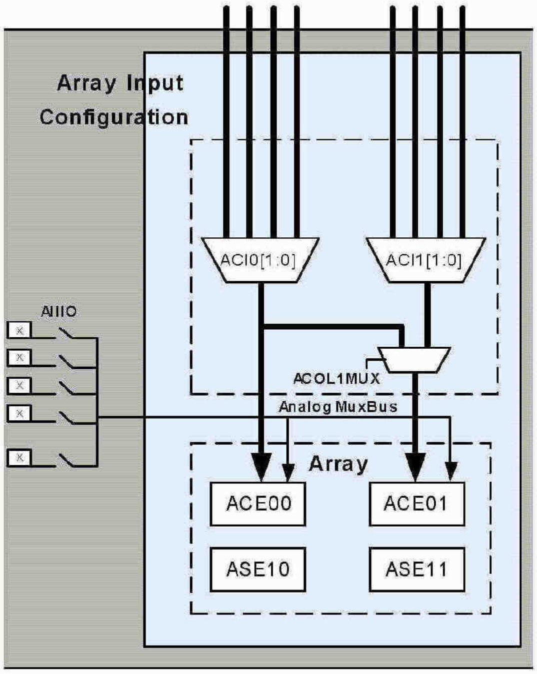
Digital blocks are provided in rows of four, where the number of blocks varies by PSoC device family. This allows you the optimum choice of system resources for your application. Family resources are shown in the table titled "PSoC Device Characteristics" on page 3.

The Analog System

The Analog System is composed of 4 configurable blocks, allowing the creation of complex analog signal flows. Analog peripherals are very flexible and can be customized to support specific application requirements. Some of the common PSoC analog functions for this device (most available as user modules) are listed below.

- Analog-to-digital converters (single or dual, with 8-bit resolution)
- Pin-to-pin comparator
- Single-ended comparators (up to 2) with absolute (1.3V) reference or 8-bit DAC reference
- 1.3V reference (as a System Resource)

In most PSoC devices, analog blocks are provided in columns of three, which includes one CT (Continuous Time) and two SC (Switched Capacitor) blocks. The CY8C21x34 devices provide limited functionality Type "E" analog blocks. Each column contains one CT Type E block and one SC Type E block. Refer to the *PSoC Mixed-Signal Array Technical Reference Manual* for detailed information on the CY8C21x34's Type E analog blocks.



Analog System Block Diagram

The Analog Multiplexer System

The Analog Mux Bus can connect to every GPIO pin. Pins can be connected to the bus individually or in any combination. The bus also connects to the analog system for analysis with comparators and analog-to-digital converters. An additional 8:1 analog input multiplexer provides a second path to bring Port 0 pins to the analog array.

Switch control logic enables selected pins to precharge continuously under hardware control. This enables capacitive measurement for applications such as touch sensing. Other multiplexer applications include:

- Track pad, finger sensing.
- Chip-wide mux that allows analog input from any IO pin.
- Crosspoint connection between any IO pin combinations.

Additional System Resources

System Resources, some of which have been previously listed, provide additional capability useful to complete systems. Additional resources include a switch mode pump, low voltage detection, and power on reset. Brief statements describing the merits of each system resource are presented below.

- Digital clock dividers provide three customizable clock frequencies for use in applications. The clocks can be routed to both the digital and analog systems. Additional clocks can be generated using digital PSoC blocks as clock dividers.
- The I2C module provides 100 and 400 kHz communication over two wires. Slave, master, and multi-master modes are all supported.
- Low Voltage Detection (LVD) interrupts can signal the application of falling voltage levels, while the advanced POR (Power On Reset) circuit eliminates the need for a system supervisor.
- An internal 1.3 voltage reference provides an absolute reference for the analog system, including ADCs and DACs.
- An integrated switch mode pump (SMP) generates normal operating voltages from a single 1.2V battery cell, providing a low cost boost converter.
- Versatile analog multiplexer system.

PSoC Device Characteristics

Depending on your PSoC device characteristics, the digital and analog systems can have 16, 8, or 4 digital blocks and 12, 6, or 4 analog blocks. The following table lists the resources available for specific PSoC device groups. The PSoC device covered by this data sheet is highlighted below.

PSoC Device Characteristics

PSoC Part Number	Digital IO	Digital Rows	Digital Blocks	Analog Inputs	Analog Outputs	Analog Columns	Analog Blocks	SRAM Size	Flash Size
CY8C29x66	up to 64	4	16	12	4	4	12	2K	32K
CY8C27x43	up to 44	2	8	12	4	4	12	256 Bytes	16K
CY8C24x94	49	1	4	48	2	2	6	1K	16K
CY8C24x23	up to 24	1	4	12	2	2	6	256 Bytes	4K
CY8C24x23A	up to 24	1	4	12	2	2	6	256 Bytes	4K
CY8C21x34	**up to 28**	**1**	**4**	**28**	**0**	**2**	**4[a]**	**512 Bytes**	**8K**
CY8C21x23	16	1	4	8	0	2	4[a]	256 Bytes	4K

a. Limited analog functionality.

Getting Started

The quickest path to understanding the PSoC silicon is by reading this data sheet and using the PSoC Designer Integrated Development Environment (IDE). This data sheet is an overview of the PSoC integrated circuit and presents specific pin, register, and electrical specifications. For in-depth information, along with detailed programming information, reference the *PSoC Mixed-Signal Array Technical Reference Manual*, which can be found on http://www.cypress.com/psoc.

For up-to-date Ordering, Packaging, and Electrical Specification information, reference the latest PSoC device data sheets on the web at http://www.cypress.com.

Development Kits

Development Kits are available from the following distributors: Digi-Key, Avnet, Arrow, and Future. The Cypress Online Store contains development kits, **C** compilers, and all accessories for PSoC development. Go to the Cypress Online Store web site at http://www.cypress.com, click the Online Store shopping cart icon at the bottom of the web page, and click *PSoC (Programmable System-on-Chip)* to view a current list of available items.

Technical Training

Free PSoC technical training is available for beginners and is taught by a marketing or application engineer over the phone. PSoC training classes cover designing, debugging, advanced analog, as well as application-specific classes covering topics such as PSoC and the LIN bus. Go to http://www.cypress.com, click on Design Support located on the left side of the web page, and select Technical Training for more details.

Consultants

Certified PSoC Consultants offer everything from technical assistance to completed PSoC designs. To contact or become a PSoC Consultant go to http://www.cypress.com, click on Design Support located on the left side of the web page, and select CYPros Consultants.

Technical Support

PSoC application engineers take pride in fast and accurate response. They can be reached with a 4-hour guaranteed response at http://www.cypress.com/support/login.cfm.

Application Notes

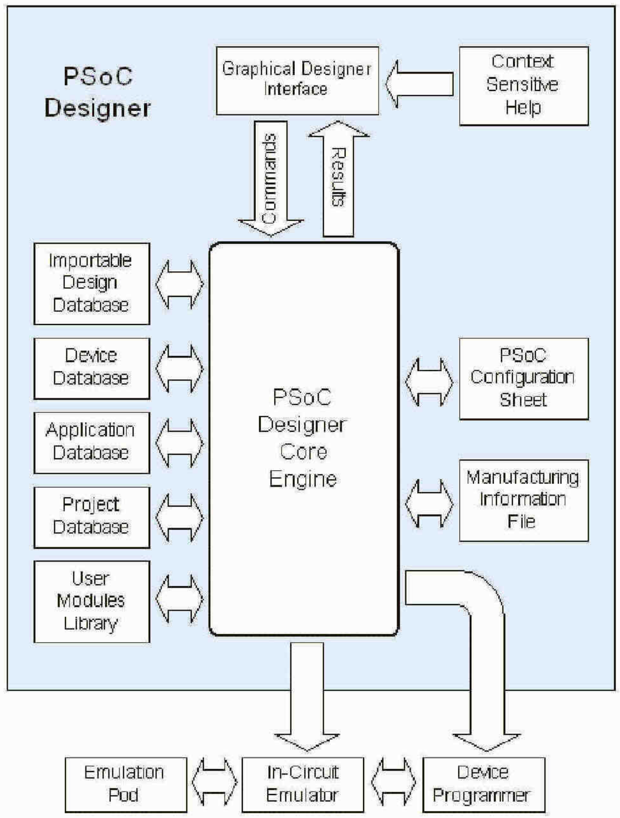
A long list of application notes will assist you in every aspect of your design effort. To view the PSoC application notes, go to the http://www.cypress.com web site and select Application Notes under the Design Resources list located in the center of the web page. Application notes are sorted by date by default.

Development Tools

PSoC Designer is a Microsoft® Windows-based, integrated development environment for the Programmable System-on-Chip (PSoC) devices. The PSoC Designer IDE and application runs on Windows NT 4.0, Windows 2000, Windows Millennium (Me), or Windows XP. (Reference the PSoC Designer Functional Flow diagram below.)

PSoC Designer helps the customer to select an operating configuration for the PSoC, write application code that uses the PSoC, and debug the application. This system provides design database management by project, an integrated debugger with In-Circuit Emulator, in-system programming support, and the CYASM macro assembler for the CPUs.

PSoC Designer also supports a high-level C language compiler developed specifically for the devices in the family.



PSoC Designer Subsystems

PSoC Designer Software Subsystems

Device Editor

The device editor subsystem allows the user to select different onboard analog and digital components called user modules using the PSoC blocks. Examples of user modules are ADCs, DACs, Amplifiers, and Filters.

The device editor also supports easy development of multiple configurations and dynamic reconfiguration. Dynamic reconfiguration allows for changing configurations at run time.

PSoC Designer sets up power-on initialization tables for selected PSoC block configurations and creates source code for an application framework. The framework contains software to operate the selected components and, if the project uses more than one operating configuration, contains routines to switch between different sets of PSoC block configurations at run time. PSoC Designer can print out a configuration sheet for a given project configuration for use during application programming in conjunction with the Device Data Sheet. Once the framework is generated, the user can add application-specific code to flesh out the framework. It's also possible to change the selected components and regenerate the framework.

Design Browser

The Design Browser allows users to select and import preconfigured designs into the user's project. Users can easily browse a catalog of preconfigured designs to facilitate time-to-design. Examples provided in the tools include a 300-baud modem, LIN Bus master and slave, fan controller, and magnetic card reader.

Application Editor

In the Application Editor you can edit your C language and Assembly language source code. You can also assemble, compile, link, and build.

Assembler. The macro assembler allows the assembly code to be merged seamlessly with C code. The link libraries automatically use absolute addressing or can be compiled in relative mode, and linked with other software modules to get absolute addressing.

C Language Compiler. A C language compiler is available that supports the PSoC family of devices. Even if you have never worked in the C language before, the product quickly allows you to create complete C programs for the PSoC family devices.

The embedded, optimizing C compiler provides all the features of C tailored to the PSoC architecture. It comes complete with embedded libraries providing port and bus operations, standard keypad and display support, and extended math functionality.

Debugger

The PSoC Designer Debugger subsystem provides hardware in-circuit emulation, allowing the designer to test the program in a physical system while providing an internal view of the PSoC device. Debugger commands allow the designer to read the program and read and write data memory, read and write IO registers, read and write CPU registers, set and clear breakpoints, and provide program run, halt, and step control. The debugger also allows the designer to create a trace buffer of registers and memory locations of interest.

Online Help System

The online help system displays online, context-sensitive help for the user. Designed for procedural and quick reference, each functional subsystem has its own context-sensitive help. This system also provides tutorials and links to FAQs and an Online Support Forum to aid the designer in getting started.

Hardware Tools

In-Circuit Emulator

A low cost, high functionality ICE (In-Circuit Emulator) is available for development support. This hardware has the capability to program single devices.

The emulator consists of a base unit that connects to the PC by way of a USB port. The base unit is universal and will operate with all PSoC devices. Emulation pods for each device family are available separately. The emulation pod takes the place of the PSoC device in the target board and performs full speed (24 MHz) operation.

Designing with User Modules

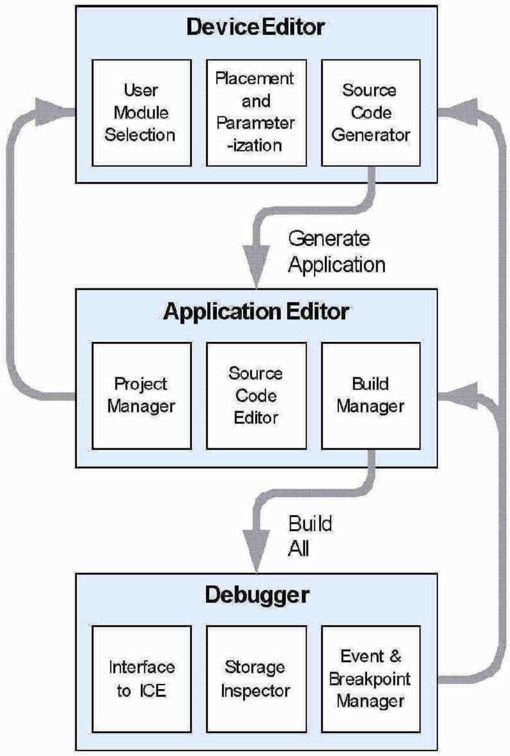
The development process for the PSoC device differs from that of a traditional fixed function microprocessor. The configurable analog and digital hardware blocks give the PSoC architecture a unique flexibility that pays dividends in managing specification change during development and by lowering inventory costs. These configurable resources, called PSoC Blocks, have the ability to implement a wide variety of user-selectable functions. Each block has several registers that determine its function and connectivity to other blocks, multiplexers, buses and to the IO pins. Iterative development cycles permit you to adapt the hardware as well as the software. This substantially lowers the risk of having to select a different part to meet the final design requirements.

To speed the development process, the PSoC Designer Integrated Development Environment (IDE) provides a library of pre-built, pre-tested hardware peripheral functions, called "User Modules." User modules make selecting and implementing peripheral devices simple, and come in analog, digital, and mixed signal varieties. The standard User Module library contains over 50 common peripherals such as ADCs, DACs Timers, Counters, UARTs, and other not-so common peripherals such as DTMF Generators and Bi-Quad analog filter sections.

Each user module establishes the basic register settings that implement the selected function. It also provides parameters that allow you to tailor its precise configuration to your particular application. For example, a Pulse Width Modulator User Module configures one or more digital PSoC blocks, one for each 8 bits of resolution. The user module parameters permit you to establish the pulse width and duty cycle. User modules also provide tested software to cut your development time. The user module application programming interface (API) provides high-level functions to control and respond to hardware events at run time. The API also provides optional interrupt service routines that you can adapt as needed.

The API functions are documented in user module data sheets that are viewed directly in the PSoC Designer IDE. These data sheets explain the internal operation of the user module and provide performance specifications. Each data sheet describes the use of each user module parameter and documents the setting of each register controlled by the user module.

The development process starts when you open a new project and bring up the Device Editor, a graphical user interface (GUI) for configuring the hardware. You pick the user modules you need for your project and map them onto the PSoC blocks with point-and-click simplicity. Next, you build signal chains by interconnecting user modules to each other and the IO pins. At this stage, you also configure the clock source connections and enter parameter values directly or by selecting values from drop-down menus. When you are ready to test the hardware configuration or move on to developing code for the project, you perform the "Generate Application" step. This causes PSoC Designer to generate source code that automatically configures the device to your specification and provides the high-level user module API functions.



User Module and Source Code Development Flows

The next step is to write your main program, and any sub-routines using PSoC Designer's Application Editor subsystem. The Application Editor includes a Project Manager that allows you to open the project source code files (including all generated code files) from a hierarchal view. The source code editor provides syntax coloring and advanced edit features for both C and assembly language. File search capabilities include simple string searches and recursive "grep-style" patterns. A single mouse click invokes the Build Manager. It employs a professional-strength "makefile" system to automatically analyze all file dependencies and run the compiler and assembler as necessary. Project-level options control optimization strategies used by the compiler and linker. Syntax errors are displayed in a console window. Double clicking the error message takes you directly to the offending line of source code. When all is correct, the linker builds a HEX file image suitable for programming.

The last step in the development process takes place inside the PSoC Designer's Debugger subsystem. The Debugger downloads the HEX image to the In-Circuit Emulator (ICE) where it runs at full speed. Debugger capabilities rival those of systems costing many times more. In addition to traditional single-step, run-to-breakpoint and watch-variable features, the Debugger provides a large trace buffer and allows you define complex breakpoint events that include monitoring address and data bus values, memory locations and external signals.

Document Conventions

Acronyms Used

The following table lists the acronyms that are used in this document.

Acronym	Description
AC	alternating current
ADC	analog-to-digital converter
API	application programming interface
CPU	central processing unit
CT	continuous time
DAC	digital-to-analog converter
DC	direct current
ECO	external crystal oscillator
EEPROM	electrically erasable programmable read-only memory
FSR	full scale range
GPIO	general purpose IO
GUI	graphical user interface
HBM	human body model
ICE	in-circuit emulator
ILO	internal low speed oscillator
IMO	internal main oscillator
IO	input/output
IPOR	imprecise power on reset
LSb	least-significant bit
LVD	low voltage detect
MSb	most-significant bit
PC	program counter
PLL	phase-locked loop
POR	power on reset
PPOR	precision power on reset
PSoC®	Programmable System-on-Chip™
PWM	pulse width modulator
SC	switched capacitor
SLIMO	slow IMO
SMP	switch mode pump
SRAM	static random access memory

Units of Measure

A units of measure table is located in the Electrical Specifications section. Table 3-1 on page 17 lists all the abbreviations used to measure the PSoC devices.

Numeric Naming

Hexidecimal numbers are represented with all letters in uppercase with an appended lowercase 'h' (for example, '14h' or '3Ah'). Hexidecimal numbers may also be represented by a '0x' prefix, the C coding convention. Binary numbers have an appended lowercase 'b' (e.g., 01010100b' or '01000011b'). Numbers not indicated by an 'h', 'b', or 0x are decimal.

Table of Contents

For an in depth discussion and more information on your PSoC device, obtain the *PSoC Mixed-Signal Array Technical Reference Manual* on http://www.cypress.com. This document encompasses and is organized into the following chapters and sections.