Why Use the Cypress PSoC?

Electronics have dramatically altered the world as we know it. One has simply to compare the conveniences and capabilities of today's world with those of the late 1900s. Computers have gone from the size of a room to the size of a pocket planner. Gone are the days of seeing a telephone only in the cars of the rich. Today, you see people in all walks of life with multiple cell phones. The advent of the Internet has brought the entire world a few steps closer and moved the target for tomorrow's electronic miracles out to seemingly sci-fi levels.

Much of what is possible with electronics today is due in part to integration of large circuits into smaller and smaller packages. Millions of transistors are etched into a fraction of a square inch of area and a paper thin thickness. Two advantages of this integration have been to allow electronics to be shrunk such that portable complex devices such as PDAs and cell phones can exist without having to lug around a 6-pound box full of bulky components. The other major advantage is that it dramatically cuts the expense of electronic production. The silicon wafer and packaging expense comprise much of the significant costs of building microcontrollers today.

As the cost of microcontrollers continues to drop, you will see that more powerful brains in existing designs will result in increased functionality and more flexible operation. There will also be a huge increase of new designs that have never before existed, since there are dozens of micros I’ve been shown that can be bought for well under a dollar in quantity. These micros boast such features as analog-to-digital conversion, Flash memory that can be self programmed, multiple timers, built-in hardware-based communications, and LCD drivers. Such capabilities have been
driving research into exploring such products as tattoos that can be altered to display what you want, clothes with brains built into them to change their look or insulation characteristics, and smart implants to dispense medicine, restore lost body functions to paralyzed victims, or build the basis to give sight to those blind from birth.

Cypress Semiconductor Corporation was formed to introduce the Cypress PSoC™ family. The product has been out for a few years now and is rapidly gaining popularity. The Cypress PSoC family offers a unique blend of adaptability in design for a relatively low price. In today’s world of ever increasing automation and digitization of the world around us, the PSoC family allows you to add brains and capable signal conditioning to a design in one complete chip. This level of integration previously existed only in much larger parts, making the advantages of integration pale in comparison to the added price and complexity. Past designs with analog needs would amplify, filter and condition the analog signals using dedicated analog circuitry and then use the micro’s analog-to-digital converter to read in the signal and do any final digital manipulations in the micro itself.

Digital manipulations of an analog signal can be quite processor-intensive and limiting to smaller micros. The Cypress PSoC adds analog manipulation capabilities into the micro for a complete one-chip analog/digital design. The PSoC chipset is designed with programmable digital blocks and programmable analog blocks that contain continuous analog and switched cap analog blocks. Since the analog blocks and all interconnects are all contained inside the chip, this provides a small design form factor. The PSoC allows real-time reconfiguration of the analog and digital blocks that allow the design to perform more than one function or to adapt to perform better.

When I was first introduced to the PSoC family, they weren’t yet in full production. I was intrigued by the advertised capabilities of the part and thought of several immediate projects where such a micro would be useful. Since then, I have worked on more than a dozen designs using the PSoC family. The adaptability and flexibility of the PSoC design assures it a place in current and future designs for quite some time to come.
Why Use the Cypress PSoC?

Notable Qualities of the PSoC Family
The PSoC family offers full 48-MHz CPU operation. The internal oscillator is rated at ±2.5% accuracy. If you aren’t concerned about creating a real-time clock or performing asynchronous communications, then the internal oscillator might very well suffice and you can save the price of a crystal. You can implement an external crystal if you do need the added accuracy in your design. The processor can also be run at lower speeds to allow a power savings if desired.

The 8-bit CPU is built around a Harvard architecture and has a built-in multiplier/accumulator (MAC) allowing for speedy instruction execution and single instruction multiplication.

The PSoC can be programmed in-circuit via a serial connection and is able to self-program its own Flash memory. This provides the ability to emulate EEPROM within the Flash and upgrade your system while it’s in the field.

The PSoC has a variety of parts that range from six to more than sixty I/O, 2K–64K of Flash, and multiple combinations of digital and analog blocks.

My Experience with the PSoC Family
I’ve been able to work on many designs with the Cypress PSoC family from conception to production. These designs have various functions including signal processing, user interface, motor control, and analog measurements. These designs have given me an opportunity to try out the capabilities of the PSoC in different situations.

I have also had the opportunity to teach the Cypress PSoC at Utah State University. It has been a marvelous teaching tool for students not only to learn the basics of microcontroller design, but also to work with the flexibility of reconfiguration and the growing pains of working with an analog system. Many of my students have abandoned their previous choices for micros in favor of the PSoC due to its ease of implementation, its great development tools, and powerful capabilities.

Getting Over Those Speed Bumps
The PSoC family of processors was breaking into some new territory when their parts were first designed. I found that I had some unique learning curves as I was working
on making this microcontroller do things that I had not as yet tried to accomplish in any other device. The growing pains were, at the time, somewhat difficult, but the successes far outweighed the disappointments for me.

I have heard some criticism about the limitations in the PSoC family. I believe that too many can get caught up pointing out the limitations of the parts and have lost sight of the myriad of problems that this family of micros can solve. I will discuss some of the limitations that I found in the PSoC along with methods and workarounds to smooth out your speed bumps and get your design up and running as quickly as possible. The limitations of the PSoC shouldn’t discourage you from considering it for your next project.

My first designs utilized the 25xxx/26xxx series as these were the only parts available at the time. There has since been a release of a new flavor of PSoC processors (24xxx/27xxx) during the development of this book that eliminate many limitations of the previous family and adds additional features that enhance the capabilities of the previous parts. The 25xxx/26xxx parts are still available and are still the best choice for some designs that I’m involved with. However, the newer PSoC parts are filling the gaps quickly and Cypress intends to phase out the older parts with their idiosyncrasies.

The scope of this book is to give the reader a reference to use in concert with the data sheet and user manual. I intend to show methods of design that will help you to utilize the PSoC to its greatest potential with a minimum of heartaches and late nights. I will include specific examples that will highlight the PSoC’s unique capabilities and have put each of those examples on the companion CD-ROM.

**A True System on a Chip**

Since the advent of the transistor and later the first integrated circuit (IC) that stored many transistors onto a single chip, there has been a continual drive to see how much we can fit into a single piece of silicon. The densities have been increasing and the processes have been shrinking away almost to nothing. The days of saving your pennies to get that math coprocessor for your 386 computer at home are long forgotten. Probably a good half of the technical population doesn’t remember actually using an 8088 or even a 286. Moore’s law has long outstripped those days of yore.
Microcontrollers aim for integration. The goal isn’t, for example, to have a system to which you can add more RAM. The object of the game is to have one chip serve as the CPU, RAM, ROM, and I/O necessary to get the entire project done. Over the years, designers have been bringing more and more onto the microcontroller to make it more powerful. They started to add such things as comparators, analog-to-digital converters, timers, interrupt controllers, and even multipliers/accumulators. In the last few years, I’ve even started to see some other micros build in amplifiers.

The PSoC development team wanted to take the integrated microcontroller system a step further. Even though there are a myriad of projects that you can accomplish with the microcontrollers of today, there was a large group of projects that required some analog manipulation of signal external to the microcontroller to process the signals to a point that they could be handled correctly by the microcontroller. The PSoC designers strived to design a microcontroller that would allow you to perform the necessary analog functions within the microcontroller itself and eliminate the need for the external analog components. At the same time, they wanted to keep all of the integration of other microcontroller systems. Thus, they created a true system on a chip.

The digital and analog capabilities are controlled via configurable blocks. The blocks are divided into sections according to design. Some blocks are designed to perform analog functions. Other blocks are used for digital purposes. These blocks are configured via a series of registers. These registers are initialized on power up using the PSoC’s built-in Flash memory. There isn’t a need for a separate configuration memory.

Embedding this analog functionality within the microcontroller poses several advantages. There is of course the space advantage. Less real estate means a smaller PCB and pennies in your pocket. The integration typically means a simpler design since you don’t have to route power and ground to multiple ICs across the board. There is also the advantage of not having to route the signal lines from one chip to another, since the routing is handled within the microcontroller.

There is advantage in the noise realm. The electrical signal paths within single ICs are much shorter than they are going to be if you have to route them externally between components. This presents smaller antennas. Smaller antennas mean you gain in both noise immunity and noise emissions. Everyone I know who has spent
Chapter 1

time in the mystical realm of trying to understand the voodoo characteristics of noise appreciates the opportunity to avoid a noise problem.

Previous designs that I’ve worked on required additional external circuitry to power down the analog section of the circuit in order to conserve power during periods of inactivity. The PSoC micro has the ability to power down the analog section by simply writing to a register. This provides a simple way to minimize power in a mixed digital and analog design. In addition, Cypress allows you to choose different power settings within its analog systems which allow you to easily optimize your design for maximum battery life and still maintain acceptable performance.

Since the configurability of the PSoC micro is all controlled by a series of registers, the PSoC is able to change its configuration on the fly. Cypress helps to facilitate this changeover by allowing the user to specify multiple configurations in the Device Editor of PSoC Designer™. If a small change is needed, then the user is allowed to modify these registers directly, which can be performed quickly and without an extensive use of program space.

A Work in Progress

There have been great ideas for improvements for the PSoC over the last couple of years, and happily some of them have been implemented. As Cypress works to resolve existing glitches and improve subsystems of the PSoC family, the PSoC will find its way into more existing designs and will open the way for new designs in areas that have never seen the benefits of a microcontroller.

Each day as I work with the various micros in my line of work, I always know that I’ll be in for a day of excitement as I work with the PSoC. I look forward to the PSoC designs that I will do in the future.