PIC® and AVR® Microcontrollers Simplify Sensor-Based Application Design

FUTURE ELECTRONICS

www.FutureElectronics.com/m/Microchip

MICROCHIP

Introduction

Modern sensor-based embedded designs using a microcontroller (MCU) have common requirements. They must take up little real-estate, often needing to operate remotely from a battery or alternative power source and frequently deal with harsh and/or noisy environments – all while providing reliable data acquisition from various sensor types. In response to these requirements, recent PIC® and AVR® MCUs focus on integrating more analog functionality while delivering low-power, robust and low-cost operation in very small packages. In this article, integrated analog on new PIC and AVR devices like the PIC18F56Q71 and AVR64DD32 will be discussed in the context of sensor applications. This article will demonstrate how these flexible products are an invaluable tool in the application developer's arsenal to help reduce design complexity and, ultimately, time to market.

Reducing Jitter and Power Consumption

New PIC and AVR devices feature more intelligence integrated into analog hardware. Peripherals like the Analog-to-Digital Converter with Computation (ADCC) add hardware based mathematic and computation capabilities that not only replace common software routines the designer would otherwise need to write, debug, and then rewrite but also eliminates latencies that waste precious CPU time jumping to different locations in memory to execute subroutines. When a task is automated in hardware, these latencies are fixed and can be reduced to merely the propagation of signals internal to the MCU through components like gates in as little as a few nanoseconds. Faster signal acquisition optimizes the MCU's ability to quickly respond; thereby reducing system jitter. This could certainly be solved using an MCU with faster operational characteristics and a purely software-based approach, but this increase in speed oftentimes comes at a cost – increasing power requirements which may not be ideal for the application at hand. It's important to also consider that there are processing speed limitations that restrict how small an MCU can physically be. Faster speeds on smaller process technologies increase the likelihood of electro-mechanical crosstalk within the device that could affect signal integrity, requiring a larger device and leading to more real-estate taken up in the application as well as an associated increased cost. Integrated analog such as the ADCC can do many of these tasks in lower power states which not only conserves power but also further reduces noise by eliminating digital processes like clocks when acquiring a signal, which could further improve accuracy.

Robust Operation

Certain environments, such as those in industrial settings, can present harsher temperatures and introduce noise which could complicate the development of sensor applications. PIC and AVR MCUs offer solutions with extended temperature ranges of -40°C to 125°C, with some specified up to +150°C, and feature up to 5.5V operation. Higher operating voltages mean that signal data acquisition is less susceptible to noise within the system. Consider an application using a sensor that outputs an I2C serial signal in a noisy environment. The threshold which is used by the MCU to determine whether a signal is a logic HIGH or logic LOW level is affected by the bus voltages used. The lower the bus voltage, the smaller the threshold between logic levels. Noisier environments could trigger false reads by corrupting values transmitted to the receiving device. Raising the voltage to 5V is a viable option to pull logic threshold levels out of the noise, reducing the need for signal conditioning either in hardware or through software routines. An additional feature on the AVR64DD32 family of devices called Multi-Voltage I/O (MVIO) is meant to accommodate operation in multiple voltage domains independent of the device VDD voltage domain. The implication here is that if a design must interface with multiple sensor types that operate at different voltage levels, the MVIO can accommodate using two power sources to the device without the need for level shifters. As an example, the MCU could read an output voltage from one sensor operating at 5V, and then still read a sensor outputting serial data on a bus operating at 1.8V without additional hardware though the use of MVIO. This multi-voltage operation also facilitates interconnectivity with other intelligent devices. For example, the AVR64DD32 device could be used as a peripheral controller, acquiring sensor signal data in a harsh environment while operating at 5V and in a lower-power state taking advantage of its robust analog capabilities. It can then send updates to a 32-bit MCU operating at 3.3V for advanced processing. Localized processing like this, or edge-computing, can improve local response to system changes and reduce costly transactions in IoT applications that communicate with cloud computing services.

Integrated Analog Reduces Design Complexity

There are many ways that integrating analog onto an intelligent device like a microcontroller can reduce design complexity, but let's discuss two here. The first, and likely most obvious way, is the reduction of external components. Newer PIC and AVR MCUs continue to integrate an increasing amount of analog – including stable voltage references, operational amplifiers, digital-to-analog converters and more thanks to the smaller core architecture found on these devices. Smaller architectures mean that a reduced overall device size can be achieved without process shrinks that make it difficult to add analog capabilities. Shrinking a process technology further introduces the risk of electro-magnetic crosstalk due to coupling capacitances and interference from adjacent signal paths corrupting data integrity.

Second, a key consideration when designing any sensor-based system is to understand possible operating conditions that could have a negative impact on the system. These could be anything from environmental conditions such as temperature extremes, voltage spikes or any other event that could occur predictably, or randomly, and require critical and specific configuration of analog within the application. Although these conditions may only occur infrequently or not at all, analog with fixed discrete components and circuitry will need to be configured in such a way to always operate in response to that possible condition. At the cost of optimized performance and power conservation. Microcontrollers that integrate analog capabilities can make use of the central processing unit (CPU) and user application code to reconfigure the integrated analog on the fly at runtime, potentially providing a solution in these situations. As an example, some newer PIC and AVR MCUs feature integrated operational amplifiers with programmable gain options that can be reconfigured dynamically as system conditions change. In this way, the system can operate in its most efficient mode and detect when a condition requiring a change in analog setup occurs using other integrated analog on the device. From there the CPU once notified or flagged could then, based on user firmware, reconfigure the analog accordingly in response. Once the condition passes and things return to "normal," the CPU could again reconfigure the analog back to a more optimized, power-conserving state. Recent integrated operational amplifiers such as those found on the PIC18F56Q71 device family, also feature hardware-based override controls which can automate reconfiguration of the op amp gain to preconfigured modes based on whether an input signal is logic HIGH or logic LOW. Developers have the flexibility to use signals internal to the MCU from integrated sources such as other analog or digital peripherals, internal clocks, software sources or signals from external circuitry. In addition, integrated capabilities such as customizable configurable logic and advanced signal routing could be used to apply fundamental intelligence in hardware to combinations of these input signals (internal and external) that can also be used to reconfigure analog, often while the device is in a lower power mode.

Summary

Integrated analog peripherals on PIC and AVR devices discussed in this article are often the right solution for many sensor-based applications. New tools like Microchip Technology's MPLAB® X IDE and MPLAB Code Configurator (MCC) graphical programming tool plugin can further reduce development time through hardware abstraction, making it easier than ever to understand how new features work on these devices. However, there are design considerations that must always be made to pick the right tool for the job and these devices may or may not fit the bill. There will be times the electrical specifications of the integrated features of the PIC and AVR MCU may fall outside of what the application requires. In those instances, perhaps a dedicated ASIC or discrete component that offers higher performance and more appropriate electrical specifications will be needed to accommodate design requirements. These alternatives can take advantage of the entire die being dedicated to that fixed analog capability, which further mitigates electro-mechanical crosstalk as discussed earlier. In the cases where more advanced processing capabilities are a requirement, a more software-centric device like a 32-bit MCU or microprocessing unit (MPU) with faster operating capabilities will best suit the task at hand. These are all decisions that must be made by the designer to use what's best for the application. This is why companies like Microchip Technology offer ASIC, 8-/16-/32-bit Microcontroller and MPU technologies to provide the sensor designer with the tools they need that best suit the application at hand.