



WP398 (v1.0) August 15, 2011

Agile Mixed Signal Addresses Analog Design Challenges

By: Anthony Collins

The design of analog interfaces presents many challenges for analog, digital, embedded, and system designers. Addressing the often diverse requirements of these analog interfaces and the verification of the design can be costly and time consuming. Xilinx's Agile Mixed Signal technology offers a better way to scale and customize common analog interfaces requirements. This technology is a unique combination of a flexible analog interface (XADC block) and the programmable logic capability of 7 series FPGAs and Zynq™-7000 Extensible Processing Platform (EPP).

Introduction

Many diverse electronic systems contain significant analog functionality. The system might need to interface directly to the real world through sensors to monitor and control the environment, for example, in an industrial automation application. Other applications might need to monitor the operating environment to ensure safe and reliable operation, for example, monitoring temperatures and power supply conditions in a communications infrastructure application. The challenges range from designing the analog conditioning circuits that need to be implemented prior to analog-to-digital conversion to verifying the analog and digital implementation as early as possible in the design phase. There is also the need to reduce costs and shorten time to market. The designer can mitigate many of these challenges by using a combination of the XADC block and programmable logic resources of 7 series FPGAs.

Agile Mixed Signal technology brings flexible digital signal processing and local intelligence to the data acquisition front end. This approach allows large portions of the analog signal processing and conditioning to be moved into the digital domain which can reduce development and product costs. Better support for analog interfaces in the digital or embedded design environment enhances verification early in the development phase and shortens time to market. In addition, the ability to integrate and customize these interfaces can greatly reduce component count and enhance the reliability, safety, and security of the system.

The Need for Analog Signal Conditioning

A wide range of sensor types are required to sense the physical environment. These sensors produce an electrical output in response to stimuli from the environment, for example, temperature, moisture, and mechanical stress. The output signals from these sensors vary greatly and must be manipulated or conditioned to map the sensor output signal into the input range of the analog-to-digital converter before it can be processed by the system. Much of this conditioning is often done in the analog domain, where functions like gain, offset, and filtering are used to scale, calibrate, and even linearize the output from these various sensors. **Figure 1** illustrates typical analog signal conditioning for a RTD based temperature sensor.

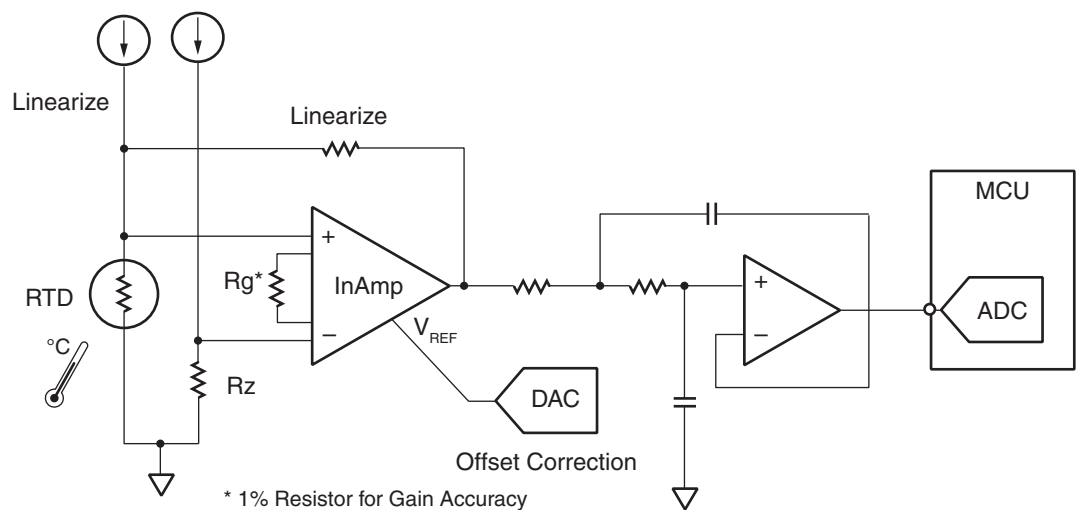


Figure 1: Analog Signal Conditioning for an RTD

This circuit performs basic linearization in the analog domain but other more complex linearization methods can be deployed [Ref 1]. However, the approach results in an analog interface that is often highly customized for a given sensor. These interfaces often need to be redesigned and verified for a range of sensor types and applications.

Challenges in Verifying the Analog Interface

It is common for the analog subsystem to be developed independently of the rest of the system. As a consequence, verification of the entire system design is often not as robust or thorough as it should be. For example, the analog designer might use a simulation tool like SPICE to verify the analog subsystem, and the digital or FPGA designer can use a tool flow like ISE® Design Suite 13.2 for development. The problem in the past has been the two design tool flows did not always communicate very well with each other, thus making mixed mode simulation of the analog and digital functions very difficult. Typically, system-level issues might not be found until the first prototypes are built. This late discovery can result in significant rework of the analog and/or digital design.

More Components Increase Cost and TTM

When much of the signal processing is carried out in the analog domain, the diverse needs of applications means system designers can find themselves dealing with a complex BOM. In addition, many of these components can be proprietary due to their analog nature. This can pose significant risk in the form of long lead times or even obsolescence of older analog ASSPs.

Leveraging Agile Mixed Signal

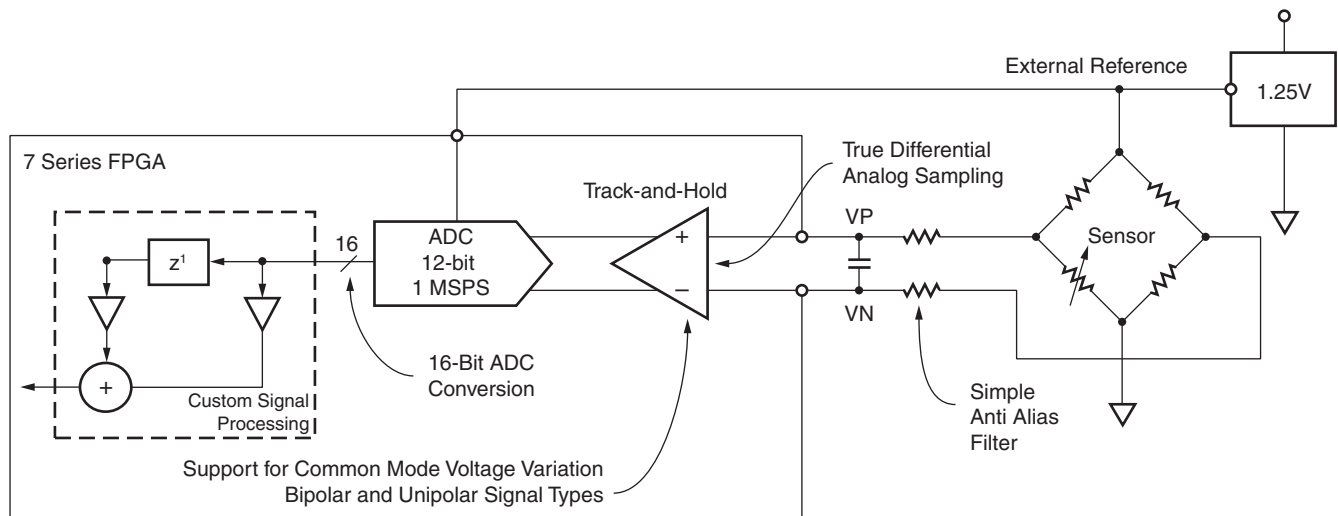
Xilinx's Agile Mixed Signal technology can help address many of the challenges described in the [Introduction](#). Typical application examples are presented in this section and the unique benefits of Agile Mixed Signal are described.

Designing a Platform for Sensor Interfacing

Many sensor applications have a number of functions in common. For example, there is often the need to calibrate (i.e., remove offset and gain errors) the analog signal processing chain and perhaps remove non linearity associated with the sensor itself (see [Figure 1](#)). While many of these functions are commonly implemented in the analog signal processing chain or sometimes in a microcontroller, they can be more efficiently implemented with the digital signal processing capabilities of FPGAs. This is especially true for high channel counts or where complex filtering is also required. Calibration in the FPGA can significantly reduce cost by eliminating components and reducing the PCB footprint of analog circuits. It can also provide a platform that is easily reprogrammed to address a range of sensor and application needs. However, the analog-to-digital converter (ADC) is key to enabling such an approach and the XADC in 7 series FPGA provides unique flexibility and functionality beyond that typically found in ADCs integrated into microcontroller units (MCUs) for example.

7 Series XADC Functionality

The XADC contains two independent Track-and-Hold (T/H) circuits that can accommodate a range of input signal types. The T/Hs also have true differential sampling inputs that support variation in the common input range and can mitigate the impact of electrically noisy environments by rejecting common mode noise (see the Analog Inputs section [Ref 2]). The ADCs in the XADC block have fully specified for 12-bit performance and are factory tested. However, the ADCs carry out a full 16-bit conversion, which significantly reduces the quantization error or noise. By oversampling and using the signal processing capabilities of the FPGA, the resolution of the analog channel can be greatly enhanced. The additional precision can also be leveraged to implement digital calibration techniques as outlined in [Digital Calibration and Processing](#). See [Figure 2](#).



WP398_02_072011

Figure 2: XADC Flexible Analog Inputs

Digital Calibration and Processing

After the analog signals have been accurately converted by the ADCs, the logic resources in the FPGA, which include DSP slices, block RAMs, and Logic function, can be applied to correct or calibrate the transfer function of the sensor. See [Figure 3](#).

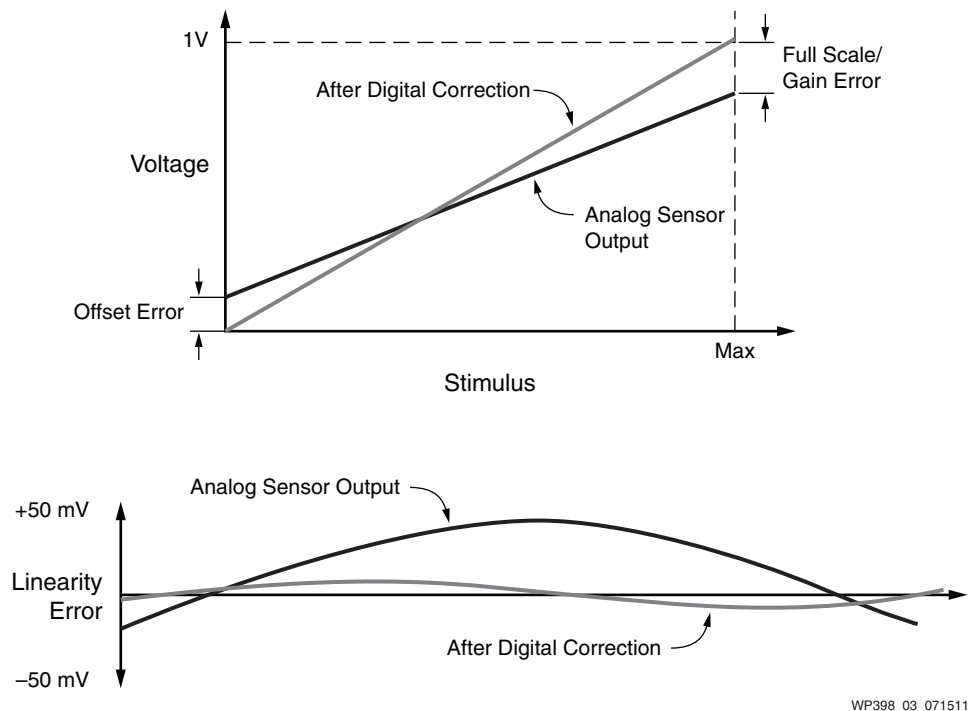


Figure 3: Digital Correction of Offset, Gain, and Linearity Errors

Offset and gain calibration of the sensor transfer function are also easily implemented. Truncation or rounding errors are mitigated by the additional conversion precision of the ADCs. To linearize a sensor transfer function, one approach might be to use an interpolated look-up table in the form of a single 1K x 18 bit block RAM. Linearization of a large number of channels is very efficient to implement in an FPGA because the same resources can be shared across all channels due to the high clock rates that can be achieved in an FPGA.

The oversampling capability of the ADC (up to 1 MSPS) and the ability to implement digital filters with a range of response types along with any stop and pass band requirement can greatly reduce the analog filter requirements. Simple passive low pass filters (as shown in Figure 2) can be sufficient to suppress higher frequencies and prevent aliasing effects.

Implementing DSP Functionality in an FPGA

Implementing DSP functionality in an FPGA has become a lot more accessible in recent years—even for non-digital designers or analog engineers with limited FPGA experience. For example, Xilinx's System Generator for DSP [Ref 3] is the industry's leading high-level tool for designing high-performance DSP systems using FPGAs. The tool, which is based on Simulink® and MATLAB® from The Mathworks, Inc.:

- Uses the Xilinx blockset that contains functions for signal processing (e.g., FIR filters), arithmetic, and digital logic.
- Provides automatic code generation of VHDL or Verilog
 - Easily targets Xilinx FPGAs with the user's DSP design.
- Performs hardware co-simulation
 - Ability to validate working hardware and accelerate simulations in MATLAB.

Analog designers with a deep understanding of the sensor type and channel requirements can easily implement all the signal processing requirements by optimizing the analog and digital signal processing elements of their design.

Verifying the Analog Interface Early in the Design Phase

Implementation of the analog interface or data acquisition is just one aspect of designing the entire system. The data acquisition system must also be integrated into what is often a complex digital or embedded control system. FPGAs are commonly used to implement sophisticated signal processing algorithms or act as co-processors to the microprocessor in such systems. Verifying the data acquisition together with the digital control system can be a complex task.

Figure 4 shows the block diagram of the data acquisition required to implement sensorless field orientated control (SFOC) of a permanent magnet synchronous motor (PMSM).

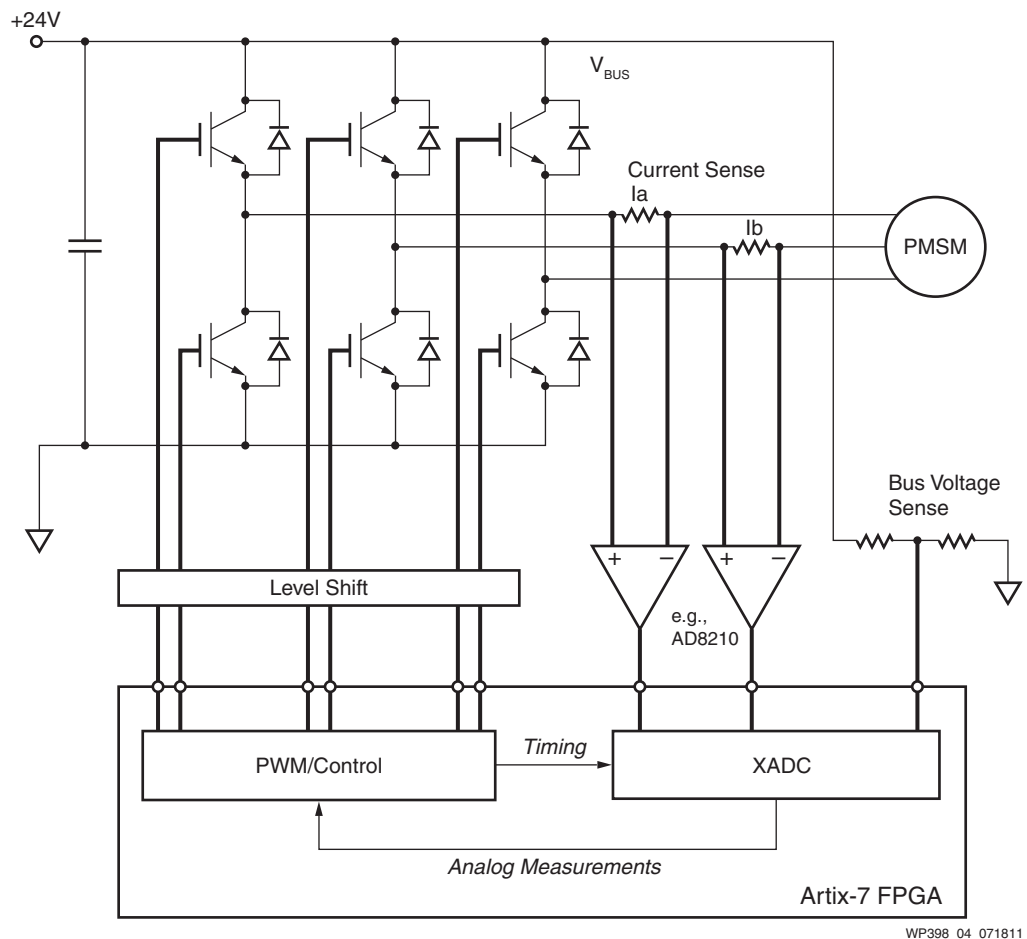


Figure 4: Data Acquisition for Motor Control

For an application like motor control, the signal processing capabilities of the FPGA are used to implement complex mathematical functions like Park and Clarke transforms at the data rate of the ADCs and close the control loop in the same rate.

An Advanced Data Acquisition System

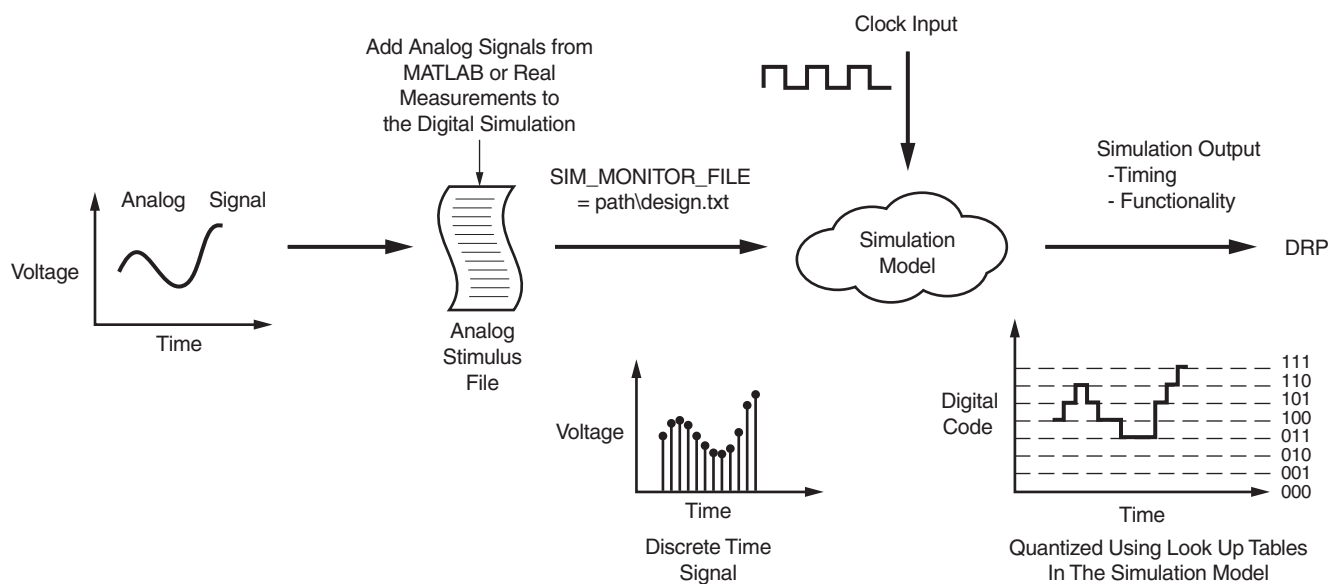
As seen in [Figure 4](#), the flexibility of the XADC block supports the complex data acquisition needs of an application like motor control. The simultaneous sampling capability of the dual ADCs in the XADC block allow simultaneous sampling of I_a and I_b (stator currents) to preserve the phase relationship, which is important for the FOC algorithm. Another benefit of the independent T/Hs is that they enhance the throughput rate of a multiplexed frontend. The T/Hs can start to acquire the next signal (i.e., V_{BUS}) to be converted by the ADC during the current conversion cycle. This greatly minimizes the impact of multiplexer crosstalk [[Ref 4](#)]. Thus, the ADCs are used to monitor a minimum of three signals I_a , I_b , and V_{BUS} .

The flexible timing of the XADC block also allows the ADCs, and in particular the sampling clock, to be synchronized to the correct pulse width modulation (PWM) state when driving the power stage (important for the algorithm).

The data acquisition and control algorithm have a complex set of interactions in an application-like motor control. The Xilinx tool chain supports verification of the design by providing a complex behavioral model of the XADC functionality, which can be incorporated in the digital verification. Xilinx also offers Hardware co-simulation capability through the iSim Hardware Description Language (HDL) simulator available in the ISE Design Suite 13.2.

Behavioral Simulation

The behavioral model for the XADC allows analog signals to be introduced into an HDL simulation without the need for mixed mode simulation capability. Simulators with VHDL or Verilog support are all that is required. Analog information is introduced into the simulation along with the correct timing using an analog stimulus file. This is basically a text file that contains "real" or analog information and the corresponding timing. The syntax is very straight forward, and analog information from a range of sources can be used. For example, the output from a MATLAB model or actual analog measurements taken in the lab by the analog designer can be easily added to the verification of the digital design. As a result, the digital design can be verified using a more appropriate testbench. See [Figure 5](#).



WP398_05_071811

Figure 5: XADC HDL Behavioral Model

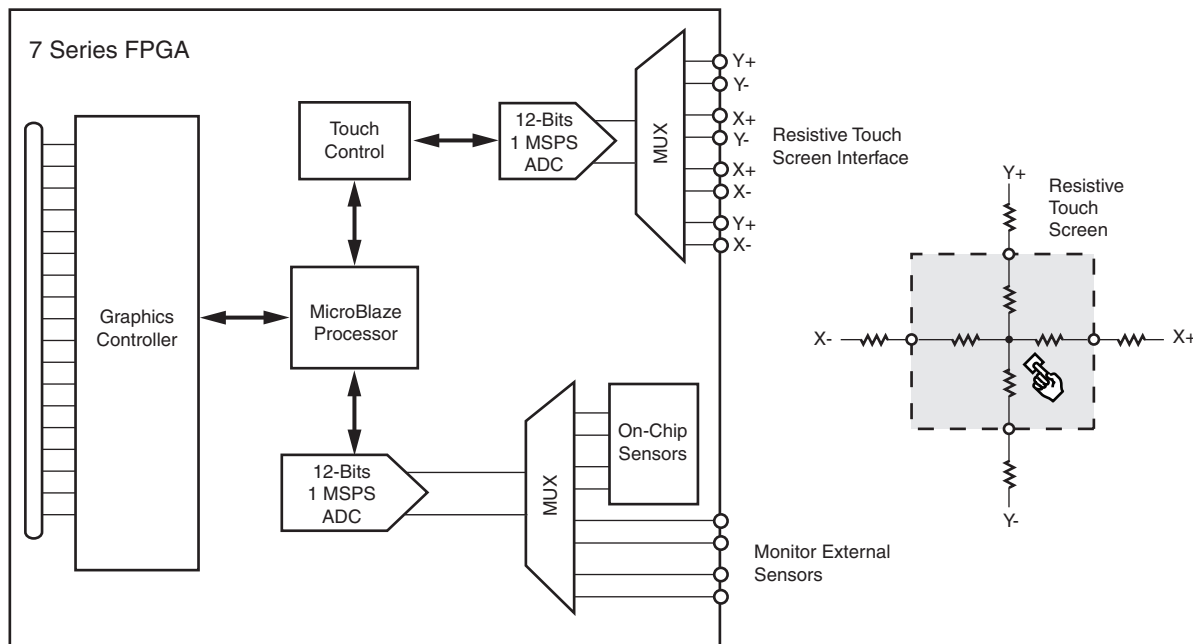
For an example of using the analog stimulus file, see the Applications section of UG480 [Ref 2].

Hardware Co-Simulation with ISim

Another approach supported by the tool set is to incorporate the actual hardware (XADC block) in the HDL simulations. Starting with ISE 13.1 software, hardware co-simulation [Ref 5] is integrated into ISim as a complementary flow to the software-based HDL simulation. This feature allows the simulation of a design or a portion of the design to be offloaded to hardware (FPGA). Thus, it is possible to run HDL verification with the simulator driving and collecting analog information from the actual motor.

Speeding Time to Market and Reducing Cost

There are many benefits to replacing aspects of the analog signal conditioning and calibration with digital calibration in the FPGA. The approach can reduce the cost and provide a platform on which to address the needs of a range of applications using the reprogrammability of the FPGA. The programmable logic can also customize the XADC functionality to eliminate the need for a range of analog ASSPs or proprietary analog components. The example shown in Figure 6 is an industrial HMI application in which the FPGA is typically used to implement the graphics controller and bitmap rendering for the display.



WP398_06_071911

Figure 6: Integrate HMI Touch Interface and Safety Sensors Using XADC

Resistive touch screens are still a common way for operators to interact with the HMI panel—especially where gloves need to be worn, making capacitive sensing less reliable. Figure 6 shows how the XADC can be used to replace an external touch screen controller ASSP. A simple state machine and one of the dual ADCs is used to implement such an interface. Another important requirement in environments like industrial automation is ensuring the safe operation of the equipment. One aspect of safety is monitoring of the physical environment (e.g., temperature and power supplies) and perhaps a range of sensors that ensure the integrity of the system. In

Figure 6, the second ADC has been deployed to monitor both on-chip and external sensors for this purpose. Thus, analog functions like thermal diode monitoring ICs and general purpose ADCs can also be integrated into the FPGA.

The integration of these analog functions reduces cost and mitigates the risk of long lead times or obsolescence of proprietary analog components.

Conclusion

This white paper describes the benefits of the Agile Mixed Signal technology enabled by the new XADC block in 7 series FPGAs. The key benefit is the ability to customize the data acquisition solution in the digital domain. This allows designers to define a platform that addressed a wide range of requirements to reduce development time and component costs. Other benefits include better support for verification of the analog interface in the HDL simulations and the ability to integrate a wide range of proprietary analog ASSPs into the FPGA.

For more information and background on Agile Mixed Signal technology, go to: <http://www.xilinx.com/ams>

References

1. Robert S. Villanucci, Wentworth Institute of Technology, Boston; Edited by Charles H. Small and Fran Granville -- EDN, February 7, 2008, Design an RTD Interface with a Spreadsheet:
http://www.edn.com/article/473143-Design_an_RTD_interface_with_a_spreadsheet.php
2. [UG480](#), 7 Series FPGAs XADC Dual 12-Bit 1MSPS Analog-to-Digital Converter User Guide.
3. System Generator for DSP:
<http://www.xilinx.com/tools/sysgen.htm>
4. Analog Devices: Understanding Crosstalk in Analog Multiplexers:
http://www.analog.com/static/imported-files/application_notes/177696709AN35.pdf
5. ISE Simulator (ISim):
<http://www.xilinx.com/tools/isim.htm>

Revision History

The following table shows the revision history for this document:

Date	Version	Description of Revisions
08/15/11	1.0	Initial Xilinx release.

Notice of Disclaimer

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available “AS IS” and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of the Limited Warranties which can be viewed at <http://www.xilinx.com/warranty.htm>; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in Critical Applications: <http://www.xilinx.com/warranty.htm#critapps>.

Automotive Applications Disclaimer

XILINX PRODUCTS ARE NOT DESIGNED OR INTENDED TO BE FAIL-SAFE, OR FOR USE IN ANY APPLICATION REQUIRING FAIL-SAFE PERFORMANCE, SUCH AS APPLICATIONS RELATED TO: (I) THE DEPLOYMENT OF AIRBAGS, (II) CONTROL OF A VEHICLE, UNLESS THERE IS A FAIL-SAFE OR REDUNDANCY FEATURE (WHICH DOES NOT INCLUDE USE OF SOFTWARE IN THE XILINX DEVICE TO IMPLEMENT THE REDUNDANCY) AND A WARNING SIGNAL UPON FAILURE TO THE OPERATOR, OR (III) USES THAT COULD LEAD TO DEATH OR PERSONAL INJURY. CUSTOMER ASSUMES THE SOLE RISK AND LIABILITY OF ANY USE OF XILINX PRODUCTS IN SUCH APPLICATIONS.