Technical Article

Industrial Drive Control Architectures: Part 1

Brian Fortman

Many industrial inverter and servo drive manufacturers have traditionally relied upon field-programmable gate-array (FPGA) or ASIC technology to complete functions that are not supported by commercial, off-the-shelf (COTS) products, like 32-bit microcontrollers (MCUs). However, adding FPGAs and ASICs to software programmable controllers, in order to support position sensor feedback or sigma-delta filtering, as examples, will increase system cost and add development complexity.

In this series of blogs we will take a look at the history of how industrial drive and servo control architectures have evolved, some of the challenges of introducing an FPGA into these architectures and how new capabilities of industrial drive control SoCs (COTS MCUs) shift the cost-benefit model away from using FPGAs in today's industrial drives.

So how did FPGAs became common practice in inverter drive and servo control architectures? When new system functions were not achievable using the off-the-shelf controllers available at the time, many developers had to implement their specific PWM/IGBT protection scheme, for example, outside of the MCU. Others may have felt that their current loop timing was too short to be handled by a programmable MCU and that it could only be accomplished in logic gates vs. software. Once an FPGA is used in the system, it then becomes a bit of a system design crutch, albeit a logical crutch, to integrate support for the new technologies introduced by an evolving drives and servo market. For example, the FPGA functions expanded to integrate clockwise/counter-clockwise (CW/CCW) and pulse train output (PTO) ports for communications with PLCs and motion controllers. Next, FPGAs became the in-system destination for supporting emerging standard and proprietary position sensor interfaces like EnDat and BiSS. Further, the interfaces to the modulated outputs of isolated sigma-delta ADCs, typically based on sinc filters, were also integrated into the FPGA device. In addition, a few of the industrial Ethernet standards have made their MAC controllers available in FPGA gates as well. All of these different functions quickly add up to hundreds of thousands of gates and a very expensive FPGA.

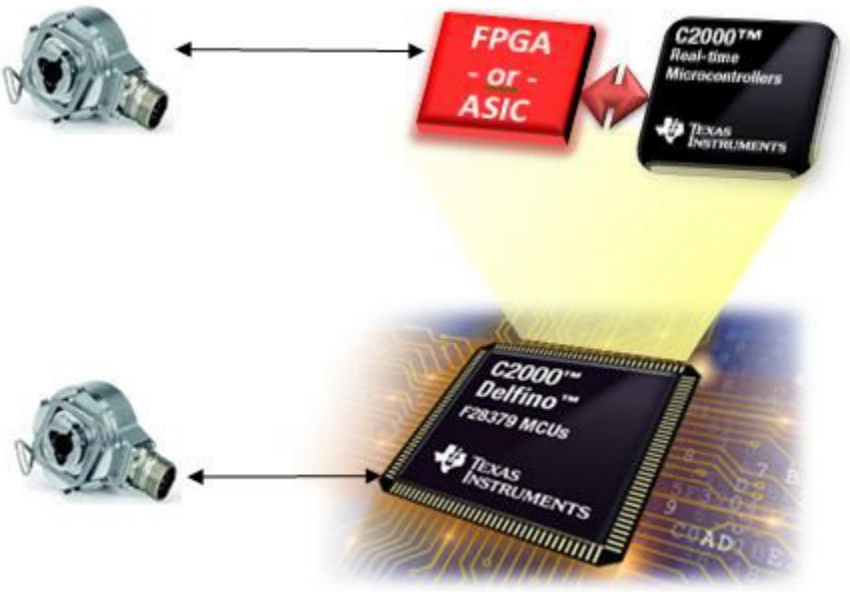

Figure 1. SoC vs. MCU plus FPGA Architectures

Copyright © 2023 Texas Instruments Incorporated

While this expansion of drive functions was being absorbed by the FPGA, an interesting new market dynamic was also taking place. Catalog controllers started bringing these functions on-chip delivering off-the-shelf features for any drive developer to use. The difference here is substantial: The on-chip functions are available for the developer to use immediately – that is, to purchase an MCU available in a catalog without the need to *construct* these solutions using an FPGA. This correlates directly as a decrease in system cost and board space savings that are not possible with an FPGA solution. With the birth of new drive control SoCs, like C2000™ F28379 MCUs with DesignDRIVE Position Manager technology, developers can now avoid the many drawbacks cited above.

In the next part in this series we will dig a bit deeper into some of the other challenges associated with bringing FPGAs into drives and servo control architectures.

Read More:

Industrial drive control architectures: Part 2

Industrial drive control architectures: Part 3

Order the C2000 DesignDRIVE development kit for industrial motor control now

Copyright © 2023 Texas Instruments Incorporated

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2023, Texas Instruments Incorporated