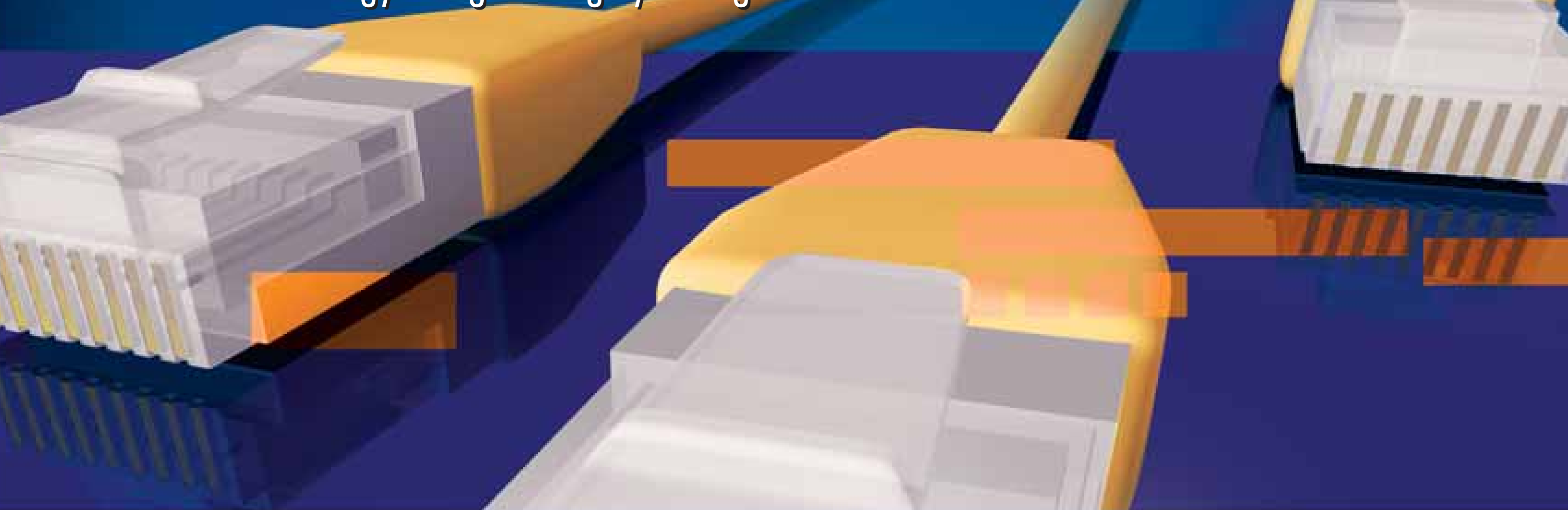
Speeding Up Control Automation with EtherCAT

You can combine FPGA-based control automation devices with EtherCAT industrial Ethernet technology using the highly configurable Beckhoff EtherCAT IP core.



by Erik Vonnahme
Hardware Engineer
Beckhoff Automation GmbH
e.vonnahme@beckhoff.com

Current control automation networks are characterized by increasing requirements regarding network and application performance, special features, and fast development cycles. EtherCAT is a high-speed industrial Ethernet control automation network that addresses these requirements. EtherCAT achieves very fast I/O cycles – 30 µs for 1,000 digital I/Os – and provides highly accurate distributed clocks for I/O synchronization with jitter and synchronicity significantly below 1 µs.

EtherCAT is standardized by IEC, ISO, and SEMI. It uses a master-slave communication schema, with a software master running on standard (PC) mainboards with a standard Ethernet network interface controller supporting 100 Mbps full-duplex communication. EtherCAT slave controllers (ESC) are processing EtherCAT frames on the fly, as shown in Figure 1; thus they introduce very few delays. The EtherCAT IP core represents such an ESC – a building block for your own EtherCAT slave devices. In this article, I'll introduce the features, benefits, and general usage of the EtherCAT IP core.

In many cases you as an EtherCAT developer are already planning to use an FPGA inside a new EtherCAT slave device such as a motion controller. This FPGA integrates application-specific I/O interfaces, performs encoding and decoding, data preprocessing, and might even contain a Xilinx® MicroBlaze™ soft-core processor to execute your software application and high-level protocols.

At this point, you would typically attach an EtherCAT slave controller ASIC to your FPGA for realization of the network interface. Such a two-chip solution increases board space, development time, and costs. Additionally, you might want to introduce special features or solutions that have additional requirements not covered by EtherCAT ASICs, since they are optimized for general applications.

The EtherCAT IP core allows you to integrate both your application-specific logic and EtherCAT slave controller functionality into a single FPGA, and also provides features beyond the EtherCAT ASIC properties. The EtherCAT IP core is designed to be FPGA family-independent and can easily be transferred from one FPGA family to another, so FPGA end-of-life becomes less important.

Inside an EtherCAT Slave Controller

To perform high-speed frame processing on the fly, EtherCAT slaves must be equipped with special Ethernet MAC and frame-processing functions that standard Ethernet network interfaces can not supply.

An EtherCAT slave controller (ESC), shown in Figure 2, provides these functions and serves as an interface between the network and slave application. Typically it has two ports for building line topology blocks. Infrastructure ESCs with three or four ports are used to realize more complex hierarchical topologies. All frame-processing tasks are completely and independently performed by the ESC, thus relieving a local application processor, which can even be omitted for simple digital I/O slaves. Incoming frames are directed by the autoforwarder to the main EtherCAT processing unit or to the appropriate output port.

Process data is exchanged between master and application using the dual-ported RAM inside the EtherCAT processing unit. Consistency between both input and output process data is achieved by hardware SyncManagers. These SyncManagers are

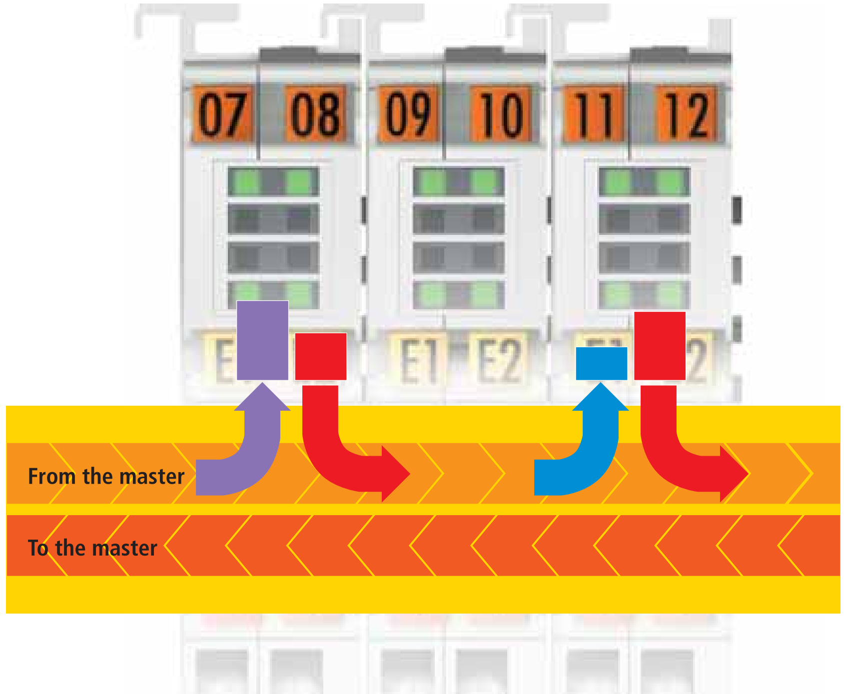


Figure 1 – EtherCAT slaves processing data on the fly

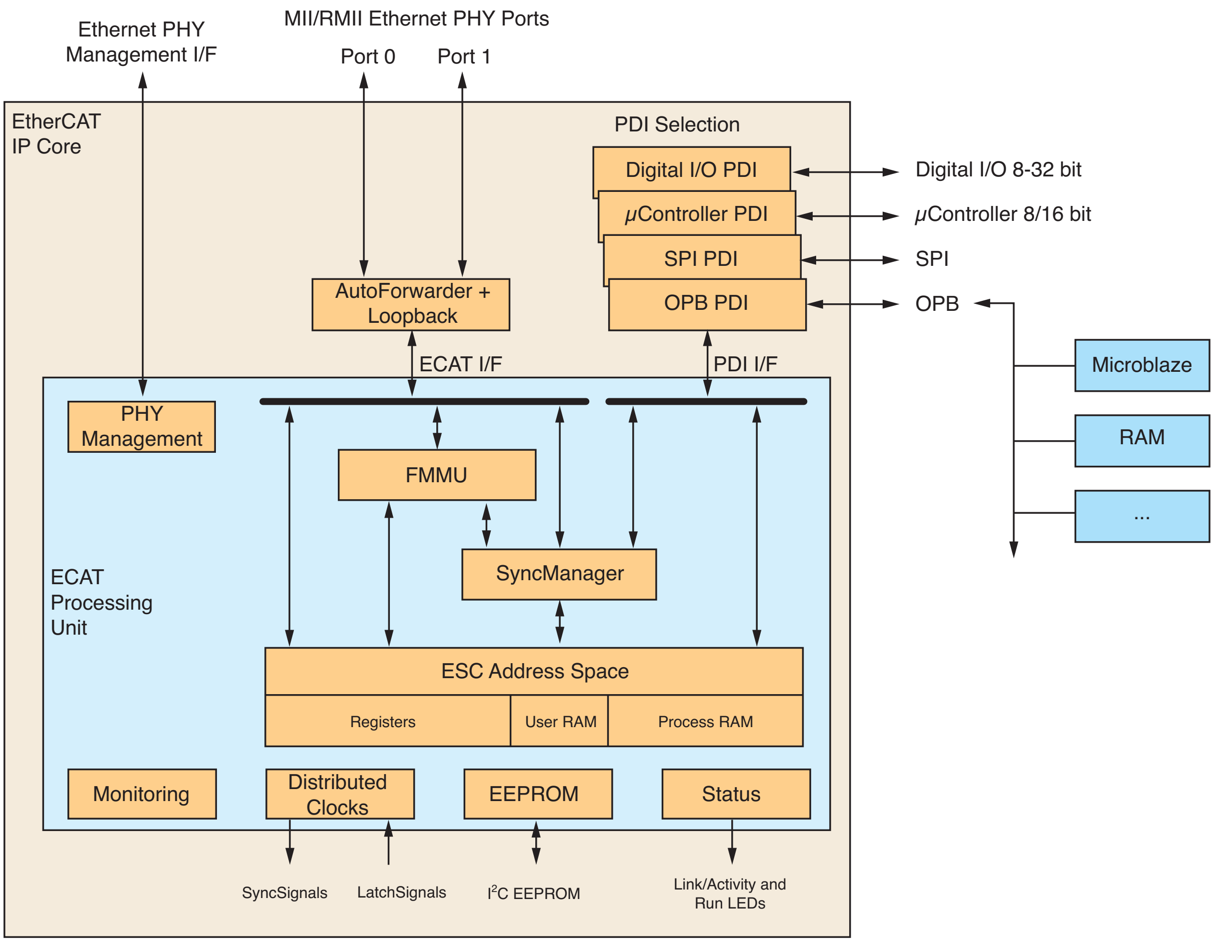


Figure 2 – EtherCAT slave controller block diagram

also used for coordinated data exchange between the EtherCAT master and a local application processor. The SyncManagers keep track of handshaking and provide error-handling functionality.

EtherCAT slave controllers provide several fieldbus memory management units (FMMU), which are used for mapping process data into a logical address space. The process data can even be mapped bit-wise, reducing the need for copying process image data inside the master application and reducing CPU time for data copying up to 30%. Logical addressing with combined read-write commands increases network performance and reduces overhead.

The local application interface is called the process data interface (PDI), which can be as simple as digital I/Os or provide a processor interface for access to the dual-ported RAM and the ESC registers containing control and status information.

Distributed clocks (DC) are another important feature of ESCs. With DCs, a precise network-wide synchronization of input and output events and time stamping is integrated into the ESCs.

Highly Configurable

In contrast to netlist-based, pre-synthesized IP cores, the EtherCAT IP core contains encrypted source code, which allows for a wide range of user-assigned configuration options and parameters (HDL generics/parameters) so you can fully customize your EtherCAT slave features and tailor it to your application requirements and FPGA resources. Because you have complete control of the synthesis and optimization process, you can achieve an optimal integration of your application-specific logic and the EtherCAT IP core.

The user-configurable options cover three main parts of the IP core (Figure 2):

- Physical Ethernet interface
- Process data interface
- EtherCAT processing unit features

With the physical Ethernet interface options, you can select either MII or RMII as the interface to the external Ethernet PHYs. Two EtherCAT ports are supported. The availability and features of the Ethernet PHY MII management interface depends on your requirements.

You can choose one of these process data interfaces for your application side:

- Digital I/O interface, with as many as 32-bit unidirectional input or output signals (with byte-wise direction settings). Synchronous input and output events can be specified, and the digital outputs are watchdog-controlled.
- Asynchronous µController interface, with 8- or 16-bit data bus and 16-bit address bus.
- SPI interface, with a clock rate of up to 30 MHz with several modes of operation.
- OPB interface for PowerPC or MicroBlaze processors, with an OPB clock frequency specified in multiples of 25 MHz.

On top of the process data interface, as many as 64-bit general-purpose inputs and 64-bit general-purpose outputs are available with the IP core.

The features of the EtherCAT processing unit are individually set depending on your application demands (Figure 3). The main parameters influencing IP core size are:

- Number of SyncManager and FMMU instances (up to eight each)
- Distributed clocks
- Internal dual-ported RAM size (1 to 60 KB)
- EtherCAT and PDI processing options, application interface features
- Diagnostic functions

You can also specify a product ID reflected in the ESC registers for easy identification of your EtherCAT device.

Simple Design Flow

A small configuration tool named IPCore_Config (Figure 3) is used to set up the EtherCAT IP core. This program generates an HDL wrapper, which reduces the IP core signals to those you really use and sets the generic values reflecting your chosen features in the configuration tool. You can use this C# tool in a Windows envi-

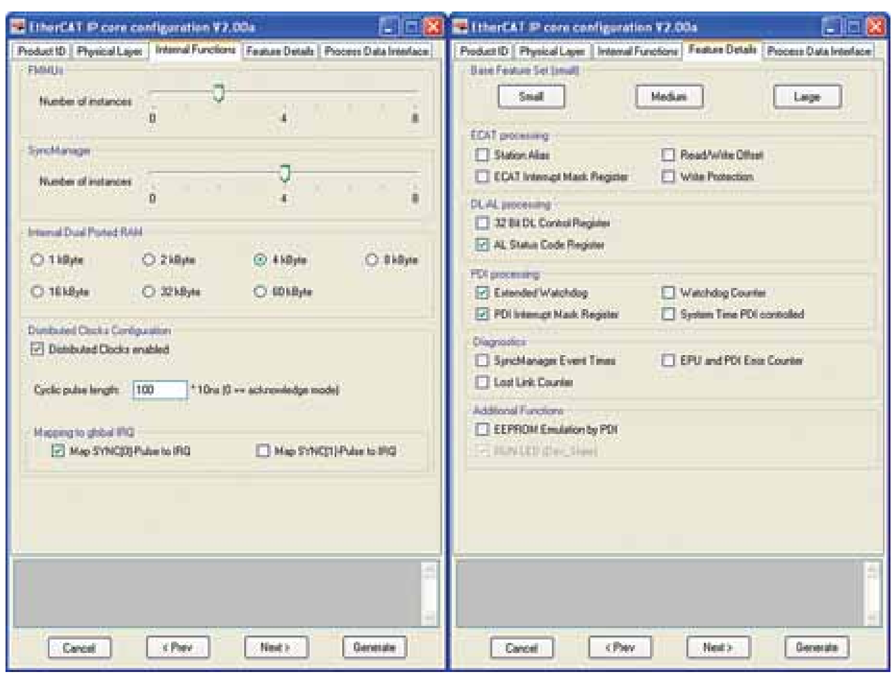


Figure 3 – Configurable options of the EtherCAT processing unit

ronment and on Linux platforms using the Mono .NET framework.

The generated HDL wrapper, together with the encrypted EtherCAT IP core itself and an encrypted vendor-specific package containing your company's vendor ID, are the basic EtherCAT design files for your ISE™ software project. You only have to add a DCM entity for clock generation and connect the signals according to your hardware setup to finish a simple EtherCAT slave design ready for synthesis. The EtherCAT IP core requires synchronous 25-MHz and 100-MHz clocks and an additional 50-MHz clock if you are using RMII Ethernet PHYs. A reset controller is recommended for proper DCM initialization. Design files for DCM instantiation and a reset controller are included in the EtherCAT IP core library.

If you are targeting processor designs with PowerPC or MicroBlaze processors using Xilinx EDK, you will choose the OPB interface for the IP core. In this case, the configuration tool generates a complete microprocessor peripheral for your EDK user repository, ready for adding it to your processor design. Some of the IP core features that have no influence on the I/O signals can even be edited using the EDK, such as the number of SyncManagers and FMMUs.

Several example ISE software projects are part of the EtherCAT IP core installation. They are showing IP core instantiation using different top-level styles (VHDL, Verilog, or schematic entry), clock generation, the reset controller, and signal interconnection for ESCs with either a digital I/O or SPI interface.

An example EDK project includes a MicroBlaze processor with data and instruction memory, debugging interfaces, general-purpose I/O and the EtherCAT IP core. A short example program running on the MicroBlaze processor demonstrates the access of IP core registers and RAM.

The EtherCAT IP core requires Xilinx ISE software V8.2.03i or higher, and at least Xilinx EDK V8.2.02i if you want to set up a microprocessor design.

All example projects are developed and tested with the Beckhoff evaluation boards FB1130 (piggyback controller board) and EL9800 (main board). The FB1130 includes a Spartan™-3E FPGA (XC3S1200E), two Ethernet interfaces, and an I^2C EEPROM. The PDI is connected to the EL9800 main board.

Many Target FPGAs

The EtherCAT IP core does not contain any FPGA-specific instances. The internal RAM is inferred automatically during synthesis and the DCM is added by the user, so its outputs can also be used for other FPGA logic apart of the IP core. According to this philosophy, the EtherCAT IP core supports a wide range of Xilinx device families:

- Spartan-3, Spartan-3E, Spartan-3A, Spartan-3AN, and Spartan-3A DSP FPGAs
- Virtex™-II, Virtex-II Pro, and Virtex-II Pro X FPGAs
- Virtex-4 FPGAs
- Virtex-5 FPGAs

New device families are likely to be already supported by the EtherCAT IP core.

Resource Figures

The FPGA resources used by the EtherCAT IP core depend highly on your selected options. The minimum configuration takes approximately 2,200 slices (Spartan-3E FPGA slices with two LUT-4 and two flip-flops), and the maximum configuration occupies approximately 12,000 slices. Typical setups without DCs will need about 4,000-5,000 slices, while DCs will add about 2,100 slices. A SyncManager instance accounts for 350 slices and an FMMU entity takes about 400 slices. Table 1 shows a configuration example with resource usage.

The wide resource range of the EtherCAT IP core strengthens the demand for a user-configurable EtherCAT slave controller. The extended features of the EtherCAT IP core, in contrast to a functionally fixed external EtherCAT ASIC, enable you to easily integrate control automation networking and the application into only one FPGA, thus creating a unique solution with a manifold of opportunities for future development. For easy startup, the EtherCAT IP core helps you with example designs and base blocks for your FPGA design included in the installation files.

Configuration Option	Example Settings
FMMU	2
SyncManager	4
RAM	4 KB
Register Set	Small + AL Status Code, Extended Watchdog, PDI Interrupt Mask, Run LED, Management Interface
Distributed Clocks	Disabled
PDI	SPI Mode 3, Normal Sample

Table 1a – Example IP core configuration

Resources		XC3S1200E
Slices	4.402	50%
Slice FFs	3.750	21%
Four Input LUTs	6.435	37%
I/Os	43	22%
Block RAMs	3	10%
GCLKs	6	25%
DCMs	1	12%

Table 1b – Resource consumption for example configuration (ISE V9.1.03i)

Because the EtherCAT ASICs and the IP core share the same functional base – which has been approved for years – further development concentrates on the improvement of peripheral functionality. Since the first major update of the IP core has been released recently, we are now collecting new ideas and customer requests for the third release.

If you are interested in more information on EtherCAT technology, visit the EtherCAT Technology Group website at *www.ethercat.org*. The EtherCAT IP core, as well as the complete data sheet, is available from Beckhoff Automation GmbH at *www.beckhoff.com*.