IPC-supported control of decentralized Servo Drives using XFC

The FPGA-based Servo Drive

An important feature of the EtherCAT-based XFC (eXtreme Fast Control) technology is that the functionalities of dedicated special controllers can be integrated in the central control unit. In addition to cost savings, significantly higher performance algorithms are possible, which can be easily adapted to other known constraints in the central control unit. From an automation system perspective, Servo Drives are subordinate special controllers. The following research article shows the feasibility and benefits of taking a Servo Drive apart into its individual components and including the control loop directly in TwinCAT using XFC.

In the following scientific application, Dr.-Ing. Jens Onno Krah and M. Sc. Christoph Klarenbach from Cologne University of Applied Sciences introduce a new kind of approach in which an FPGA programmed in VHDL is used in a Servo Drive in place of a dedicated µController. The solution is principally based on the outstanding technical features of EtherCAT and XFC technology.

Motion Control with Servo Drives

For a long time, dynamic Servo Drive control was firmly linked to the direct current motor. The power stage consisted of four power switches and current control was implemented in an analog circuit using operational amplifiers. The velocity control was also often implemented using operational amplifiers, whereby the actual velocity was measured by an analog tacho-generator attached to the motor. Usually a dedicated motion controller, which influenced the target velocity of the servo controller via its command variable, was employed for controlling the position. The actual positional value was provided to the motion controller by an encoder.

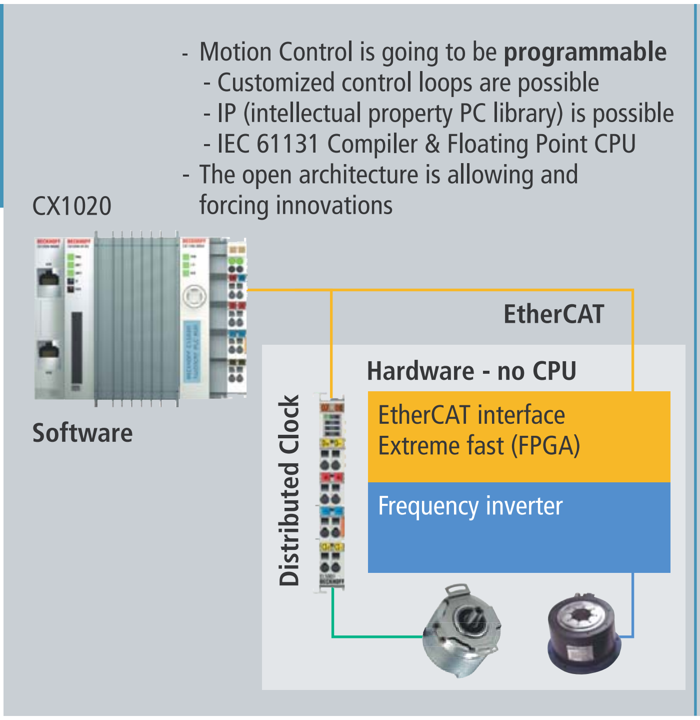
The "typical Servo Drive" has changed step by step over the course of time:

- In order to increase reliability and to reduce motor size, the servo motors are commutated electronically (six-step). (Brushless direct current – BLDC)
- In order to reduce the torque ripple, the servomotors are commutated sinusoidally. (Brushless alternating current – BLAC)
- Further size reduction is achieved by the development of motors with single tooth windings.
- The numerically differentiated positional signal from the encoder or resolver is used instead of the analog tacho signal for measurement of the velocity.
- Current and velocity control are realized using digital processors as sampling control. Due to this, control parameters are precisely reproducible instead of "adjusting a potentiometer to 1:00 pm". The parameters can also be changed via the fieldbus.
- The analog set value (±10 V) is replaced by fieldbus process data.

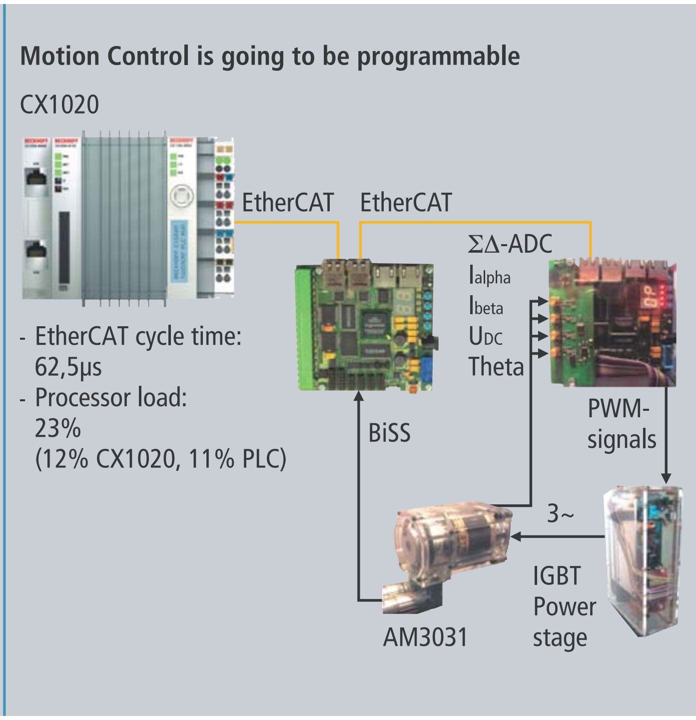
Digitalization also allows the use of more complex control algorithms:

- field-oriented control of synchronous and induction motors
- predictive control algorithms to reduce the switching frequency without reducing the bandwidth (Smith Predictor)
- model-based control algorithms to increase the dynamics (Luenberger Observer)
- consideration of non-linear behavior, such as iron saturation of the motor inductance in the case of high current operation
- Motion Control functions are implemented more and more often as a software module either inside the servo controller or the machine controller/ Industrial PC.

Using fieldbus systems such as the SERCOS interface, closing all control loops locally virtually became the standard in the machine tool industry. From a control point of view, this is a great advantage because the fieldbus cycle times do not



- Motion Control is going to be **programmable**
  - Customized control loops are possible
  - IP (intellectual property PC library) is possible
  - IEC 61131 Compiler & Floating Point CPU
- The open architecture is allowing and forcing innovations



Motion Control is going to be programmable

- EtherCAT cycle time: 62,5µs
- Processor load: 23% (12% CX1020, 11% PLC)

limit the bandwidth of the control loops due to dead time. This approach is used successfully in many applications today.

One disadvantage of digitalization using processors remains the system-related dead time of sampling control with $T_T = T_A/2$. The sampling times mostly lie within a range between 50 to 250 µs. This results in only a very small impairment at switching frequencies of up to 10 kHz. At higher switching frequencies, however, the sampling control can significantly limit the achievable current control bandwidth.

Is the intelligent Servo Drive a "dead end"?

A competitive situation has arisen between the producers of Servo Drives and the controller manufacturers concerning the various Motion Control functions. As a result, the product managers of the Servo Drive manufacturers have ensured that more and more "intelligence" is built into the Servo Drives. Today, most manufacturers offer more or less powerful PLC functionality (IEC 61131) inside the Servo Drive.

Due to different requirements, clear tendencies have arisen in only a few applications:

- In CNC and robotic applications the Motion Control functionality is almost always managed inside the machine controller (IPC):
  + The algorithms for the coupled servo axes are calculated by at least one fast CPU – often with a floating-point-unit (FPU).
  - High fieldbus bandwidth is required.
- In special technology applications such as a "flying saw" or a cross cutter, the Motion Control function is often managed inside the Servo Drive:
  - The Servo Drive requires a fast µController with a large memory capacity.
  + High fieldbus bandwidth is not required.
- Conversely, for cam plates, a clear trend is not visible. Closing the control loops inside the servo controller is traditionally the preferred solution in Europe – even with analog Servo Drives. In the USA, however, it is typical to close only the current loop inside the drive. Velocity and position loops are mostly closed in the controller.

Especially in applications with several axes, the installed computing power of the individual drives is utilized very inefficiently:

- There is no provision for the even distribution of the required computing power over the individual drives.
- The execution of the programs in the servo µControllers is more similar to that of an interpreter than to executing a compiled program.
- A significant portion of the computing power is required by some fieldbus interfaces.

Another disadvantage of closing all loops inside the Servo Drive is that the user can only parameterize the control loops, but cannot modify the control structure itself. **Motion Control** is **configurable** and **parameterizable**, but *not* **freely programmable**! This has lead to the challenge that some drives, which are configured via almost 1,000 parameters, are too complex for the user.

All in all, the manufacturers can only "accommodate" the costs of intelligence in larger drives (from around 1 kW). However, since the drives can almost never completely replace the PLC, the strategy of installing more and more microcontroller computing power inside the Servo Drives leads to a dead end.

A new approach: the FPGA-based Servo Drive

In this article, a new approach is presented in which a dedicated µController is no longer used in the Servo Drive. A Field Programmable Gate Array (FPGA) – programmed in VHDL – is used instead.

This innovative approach is essentially based on the following new technologies:

- EtherCAT fieldbus with distributed clocks (DC) and XFC
- ∑Δ modulator for analog-to-digital conversion
- digital encoder interfaces (EnDAT 2.2, BiSS, A quad B)
- digital signal processing using FPGAs
- flexible 6-phase PWM in VHDL
- soft core µController in FPGAs (system on a programmable chip: NIOS II)
- Embedded PC with floating-point-unit

Summary

In this article, a concept is presented wherein the control loops of a Servo Drive are closed centrally via EtherCAT with XFC technology and distributed clocks. The advantages of this EtherCAT/FPGA servo concept are:

- The advantages of analog and digital control technology are combined due to digital signal processing in an FPGA.
- The position feedback from motor and/or load using a resolver or an encoder can be realized modularly via one or more separate EtherCAT Terminals.
- A standard frequency converter with a fast FPGA-based EtherCAT interface suffices as an IGBT power output stage.
- Customer-specific control structures can be created easily in an IEC 61131 programming environment using floating point algorithms.
- There is no limitation of the control loop bandwidth at switching frequencies up to 8 kHz → T_a = 62,5 µs.
- The open architecture makes use of the Motion Control IP of machine manufacturers or third-parties possible, which promotes innovation.

EtherCAT fieldbus with distributed clocks and eXtreme Fast Control Technology

EtherCAT is an Ethernet-based, real time fieldbus. The special features of EtherCAT are briefly described below:

- standard Ethernet hardware: RJ 45 connectors, transformer, cable and PHY (standard MAC also in the IPC/EtherCAT Master)
- The slave interface connection needs only an ASIC or FPGA (I/O, drive, …).
- extremely optimized and extremely fast (only one Ethernet frame for several drives)
- distributed clocks (DC): high precision synchronization (<< 1 µs) due to the continuous adjustment of distributed clocks
- XFC: Within one sampling period
  - the actual values are first read via EtherCAT,
  - then the control algorithm is calculated by the IPC
  - and the new target values are written immediately via EtherCAT.

The EtherCAT slave interface is available as a configurable FPGA IP core (Intellectual Property). Only two PHYs (in & out), two transformers and two RJ 45 connectors are additionally necessary outside the FPGA.

The process data, set values and actual values are processed inside the FPGA using a VHDL program. The service data, control parameters and monitor functions (EtherCAT: Mailbox) are processed by an FPGA-internal soft core processor (NIOS II).

∑Δ modulator for analog-to-digital conversion

The quality of the analog-to-digital converter is of great importance to servo controllers. The phase current measurements are especially critical. Traditionally, these currents are first converted by hall effect current transducers into an electrically isolated voltage that can then be digitalized by SAR ADCs (successive approximation). Sampling is usually carried out at certain harmonic-free times in order to minimize aliasing effects without using special low pass filters. Additional analog comparators are often used to detect overcurrent very rapidly so that the power stage can be switched off immediately in order to protect them.

Thanks to the use of ∑Δ-ADCs, the control quality can be improved with considerably less effort. Several semiconductor manufacturers offer integrated circuits specifically designed for potential-free current measurement. The differential analog input of these ICs can be directly connected to a shunt for current measurement; the electrically isolated, digital bit-stream is connected to an input of the FPGA. Signal transmission, filtering and sampling are carried out digitally. If the ∑Δ modulator is placed directly at the shunt, disruption of signal transmission, filtering or signal processing, as a matter of principle, is not possible due to the steep switching flanks of the output stage.

The bit-stream of the current signal is conditioned and processed further in the FPGA in three channels:

1. very fast for overcurrent detection (approx. 2 % precision)
2. fast with approx. 12-bit precision for the proportional component of the current controller
3. high precision (integrated over one switching frequency period) for the integral component of the current controller

The patent for this new current control scheme is pending.

Digital encoder interfaces

Digital encoder interfaces allow fast transmission of the high resolution angle data of modern sine-cosine encoders. Even long cables do not reduce the signal quality. In addition to the position data, service data such as an electronic type plate can also be transmitted. The evaluation of a motor coil’s temperature sensor can also take place via this transmission channel; fig. 2.

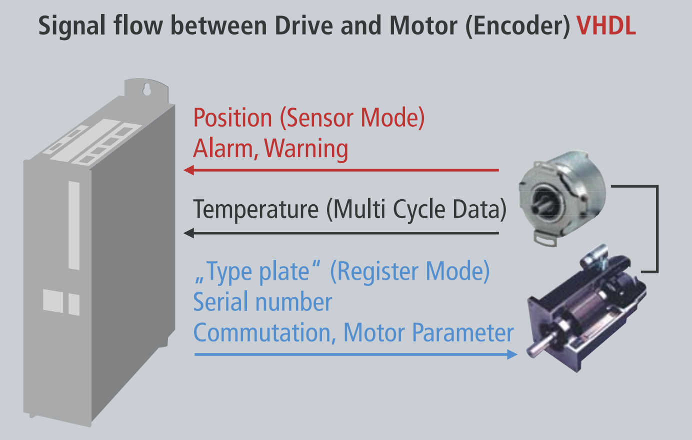
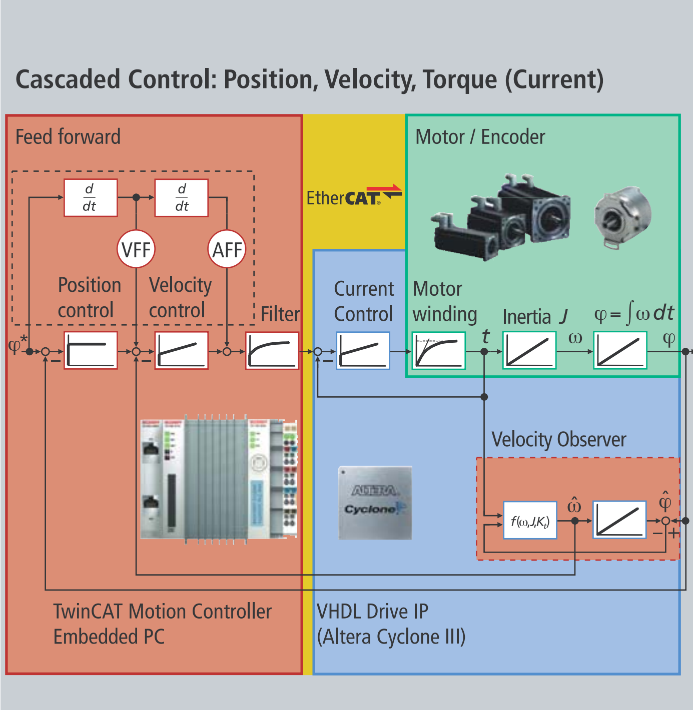
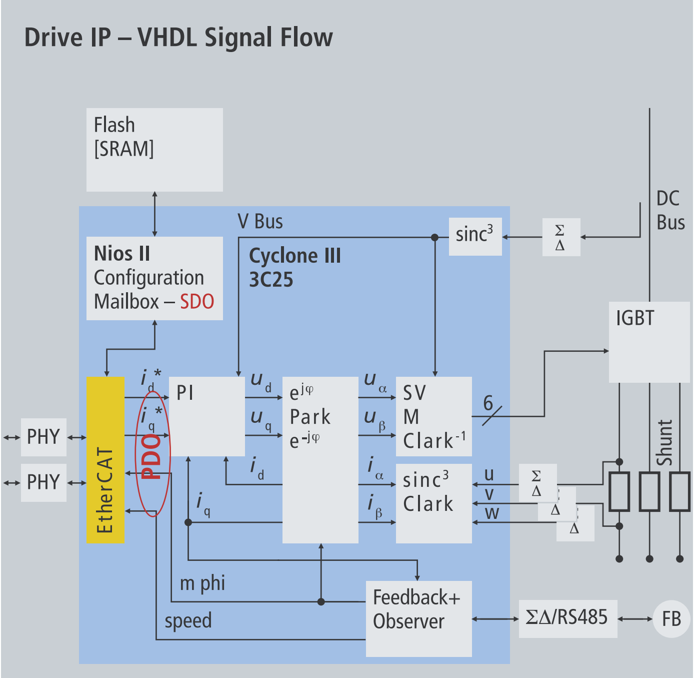


Fig. 2: Using a digital encoder interface, additional information can be transmitted along with the position. Analog signals are not necessary.



Due to the new FPGA-based concept, only a few integrated circuits are necessary in addition to the FPGA in order to build a Servo Drive.



Using cascade control, the current control loop is closed inside the FPGA. Velocity and position loops are closed in the controller via EtherCAT with XFC (without additional dead time).

Classic µControllers can usually only evaluate two tracks from standard TTL encoders (A quad B or ROD). An interrupt must then be used for the evaluation of a zero pulse. EnDAT 2.2 or BiSS are not supported with special hardware by any µController.
Encoder evaluation with fine interpolation (+6 bit) of TTL encoders and/or the EnDAT or BiSS protocol or resolver evaluation can be realized easily and inexpensively as a VHDL module in an FPGA.
The encoder query is started, triggered or synchronized inside the FPGA using the sync signal, which is an EtherCAT distributed clocks function. The process data is hardware-routed internally inside the FPGA with no additional delay and is sent with the next EtherCAT frame to the IPC. It is not necessary to connect the positional feedback (encoder and/or resolver) via the servo controller. In fact, it is also possible to close the control loop via a resolver or encoder terminal via EtherCAT.

Digital signal processing using FPGAs

The functionality that was previously implemented using operational amplifiers can be realized today easily using so-called digital signal processing (DSP) blocks inside an FPGA. Since digital signal processing inside an FPGA takes place in parallel and not sequentially as in a µController, even complex algorithms can be executed extremely quickly, in less than 100 ns. As a result, the advantages of both technologies can be used, without any of their disadvantages:

analog control:	→ no dead time, no aliasing
µController-based control:	→ reproducible parameters, complex algorithms

Using the DSP functions, even a current controller with clock frequencies significantly greater than 20 kHz can be achieved completely digitally without the bandwidth of the control loop being reduced unnecessarily due to slow sampling.

Flexible 6-phase PWM in VHDL

Many µControllers developed for Motion Control offer a special 6-phase PWM for the 6 power semiconductors of the power inverter – usually IGBTs. Switching times are calculated according to the undershoot method, i.e. the desired output voltage is specified for each phase individually and the desired blocking time is specified for all phases in common. However, space vector modulation (inverse Clark transformation) must additionally be carried out in the software. Over-modulation or block commutation is not possible at all.
Many limitations disappear if the 6-phase PWM is realized in VHDL. In addition to a standard 6-phase PWM, the following features are supported:

- Space Vector Modulation (SVM)
- Modified Space Vector Modulation (MSVM – one phase does not switch)
- over modulation/block commutation
- boot strap gate driver voltage model for each phase
- optional PLL for hardware synchronization with the controller
- online configuration of the switching frequency in 1 Hz steps
- online configuration of the blocking time in 20 ns steps

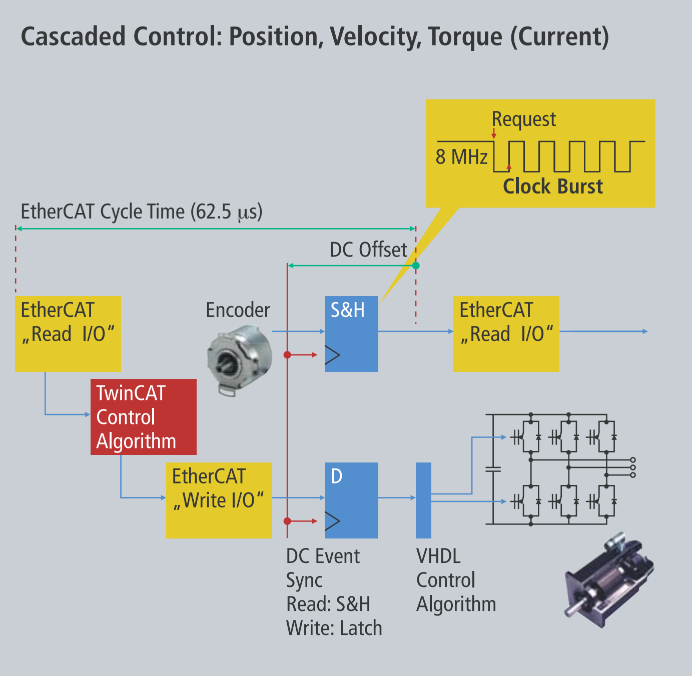


Fig. 5: Velocity and positional control sequence:

1. **EtherCAT 'Read I/O' reads the control variables/actual values.**
2. **TwinCAT calculates the control algorithms.**
3. **EtherCAT "Write I/O" writes the target value/current setpoint.**
   FPGA-internal synchronization takes place via the DC sync signal:
   - **latch of the position**
   - **accept the target (command) value**

Soft core controller in FPGAs

For initialization and to process the service data objects (SDOs), it is nonetheless still useful to employ a µController. A soft core CPU such as the NIOS II soft core µController ("system on a programmable chip") by Altera is ideally suited to this purpose. The program for the µController and the FPGA configuration can be permanently stored in a parallel flash memory. Depending on the type of FPGA and the application requirements, an external RAM can be necessary in addition to the FPGA's internal RAM. The clock frequency of the NIOS II soft-core µController is usually in the range of 50 MHz and is entirely sufficient for the processing of the service data.

Fig. 3 shows the block diagram of the control structure realized in the FPGA for a field-oriented, controlled drive. To reduce the complexity of the block diagram, not all signal paths for configuration are shown. The NIOS II can process Mailbox data and configure the individual VHDL modules inside the chip via the dual-port RAM of the EtherCAT module. In event of a fieldbus error, the CPU can shut down the servo motor in a controlled manner according to a preconfigured mode. This can be done using a velocity controller implemented by the NIOS II controller, or via a "controlled" short circuit motor current.

Prof. Dr.-Ing. Jens Onno Krah

Jens Onno Krah is employed in the Information, Media and Electrical Engineering faculty in Cologne University of Applied Sciences, Germany. He was appointed as professor in "General Control Technology" in 2004. After studying electrical engineering at the University of Wuppertal, Germany, he gained his PhD there in 1993 in the field of drive control under Prof. Holtz. He worked for Danaher Motion until 2004 as technical director and was responsible for the worldwide development of the Danaher Motion servo controller among other things. His particular fields of interest are digital signal processing and Motion Control.

Christoph Klarenbach, M. Sc.

Christoph Klarenbach studied electrical engineering at the University of Wuppertal, Germany, from 2001 to 2006 and gained his Master of Science in 2006. His main interests are the fields of control and drive technology as well as power electronics. At the moment he is working on his doctorate in the field of drive technology using FPGAs. This project is funded by Beckhoff.

Cologne University of Applied Sciences, Faculty of Information/Media Technology and Electrical Engineering

In the special subjects Automation Technology and Electrical Energy Technology, the study of Electrical Engineering offers two special fields of study that are concerned with the complex processes in power stations, energy and environmental technology and high tension applications. In the field of Information and Communication technology, the emphasis lies in the processing and transmission of information using the latest telecommunication systems, acoustical engineering, high frequency technology, and radio and television technology. Networking and digital transmission methods play an ever greater role here.

The course of study in Photo Engineering and Media Technology offered by the Cologne University of Applied Sciences is unique in Europe; the main course contents include the generation, storage, processing and playback of static and moving pictures.

→ Fachhochschule Köln www.fh-koeln.de

Embedded PC with floating-point-unit

Today, many compact controllers use Intel®-compatible processors with a floating-point-unit. The CX1020 Embedded PC from Beckhoff is such a device. Beckhoff TwinCAT PLC/NC software turns a PC with a Windows operating system into a real-time controller with cycle times from 50 µs, upwards. The task scheduler is configured in such a way that Windows XP Embedded only gets the remaining CPU time that is not required by the controller. Due to the very fast floating-point-unit, even complex control algorithms for several servo axes can be programmed very easily. For example, a velocity observer can also be implemented in TwinCAT in this way. The TwinCAT programming environment allows, among other features, Monitoring, Powerflow, Breakpoint, Single-Step and ScopeView.