Optimized interaction of control hardware and software relocates complex control loops into the central controller

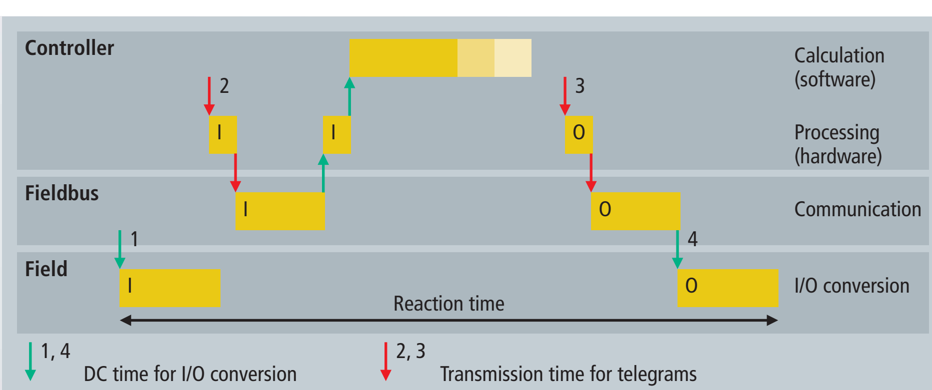
XFC: control technology in the sub-100 µs range

Closing fast control loops with very fast reaction times in a centralized controller is a challenge even for the very latest controllers. A high speed fieldbus system and high performance hardware platform alone are not sufficient to accomplish this. To actually achieve genuine reaction times of significantly less than 100 µs, all software and hardware components employed must be perfectly matched to one another. With its eXtreme Fast Control (XFC) technology, Beckhoff set this standard at an early stage. TwinCAT 3.1, the latest software generation from Beckhoff, continues on this path even further by allowing 100 % utilization of processor power. Through the intelligent use of processor cores for real-time applications, greater computing power is now available for applications equipped with multi-core PCs. TwinCAT recognizes CPU capacity not used by Windows and can integrate each available core into the real-time system by configuration. This leads to optimized scalability and full utilization of system resources for each application task and permits the assignment of complex control loops to the central controller without any loss of performance for further real-time tasks.

Modern Ethernet-based fieldbus systems such as EtherCAT enable higher-level controllers to communicate with the connected components in the sub-100 µs range. Beckhoff has already presented cycle times of 12.5 µs at the Hannover Messe 2012 and a further reduction is theoretically possible. XFC technology from Beckhoff is the "missing link" to successfully achieve reaction times significantly under 100 µs and serves as the foundation to take full advantage of a fast fieldbus and high performance control components.

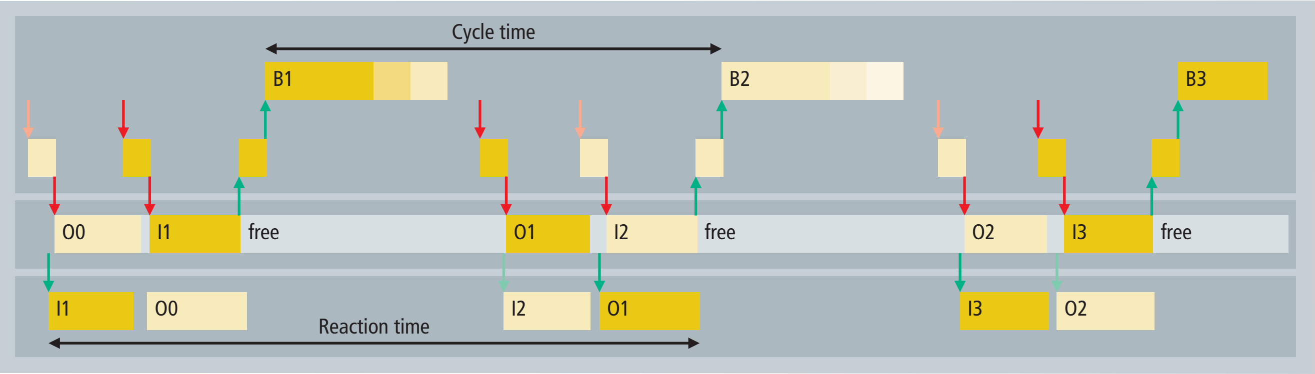
The "reaction time" is defined as the time between the occurrence of an external signal and the output of a reaction to an actuator. It contains the acquisition and conversion of a physical signal in the sensor, the time taken to communicate with the controller, the processing within the controller, the time required for transmission to the actuator and the conversion into a physical signal. The reaction time is crucial for the efficiency of many control processes: the shorter it is, the faster the controller can react to deviations from the desired condition.

Reaction time and time control



Control loops in automation technology that require very fast reactions are therefore closed locally by special devices and not in the higher-level controller. Typical examples are drive amplifiers, which close the current control loop, the velocity control loop and often the position control loop directly in the drive amplifier and only receive the set position from the higher-level controller. Granted, this is a practical solution for many applications. However, such decentralized controllers frequently must adjust up to 100 parameters to the specific task.

If the controlled system becomes more complex and depends on many influencing factors that are not available at the local device, central regulation by a powerful central controller is more efficient. All information is simultaneously available to it, and states that possibly originate from other control loops or are detected by other sensors can also be taken into consideration. A further important advantage of a central controller is the possibility to adapt the control algorithm to each specific application – directly by the user in a familiar programming environment and language. However, the advantages of this approach are usually lost again if the central controller has a significantly slower reaction time due to the additional required communication.



Reaction time versus cycle time

Fieldbus communication

An essential prerequisite for the shortest possible reaction time of a distributed control system is fast and deterministic communication. EtherCAT offers a very good basis for this with its efficient transmission and the Distributed Clocks feature in particular. The transformation of input signals must be initiated in such a way that the result is ready and has arrived at the controller directly before the next communication cycle. The output communication cycle should then be initiated directly following the calculation of the set values so that the values have just arrived in the actuator when their conversion is due. For the output data, the applicable criterion is that they are sent "early enough;" the input data, however, may not be communicated "too early," otherwise, their conversion will not be completed. In order to initiate the common communication cycle for inputs and outputs at the end of the control cycle, the standard mode of operation of an EtherCAT Master is not sufficient, since valuable time would elapse unused.

Time-controlled double transmit mode

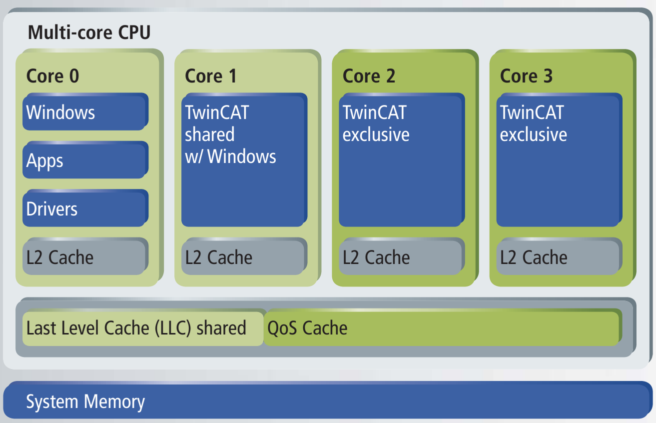
By means of separating the input and output communication, it is possible to calculate and implement the optimum time for each corresponding transmission time. Strict time control of communication permits temporally optimized dispatching of the input and output telegrams and the setting of Distributed Clocks in the sensors and actuators to match it.

A further important property of EtherCAT – the parallel operation of different communication cycles and cycle times on a single system – should also be retained. Apart from the extremely fast control, it should also be possible to exchange other signals in "more normal" cycle times on the same EtherCAT node. The remaining bandwidth can therefore be used efficiently in order to cover the communication needs of other tasks in the controller.

In the configuration phase, the TwinCAT system configurator calculates the precise transmission times of all EtherCAT telegrams for a period that equates to the lowest common multiple of all cycle times involved and transfers this table to the EtherCAT master. Following this period the process repeats itself, so that the table can be continually processed by the master. The master can then independently fill remaining temporal gaps in the table with acyclic communication requirements – e.g. with parameter data via CoE (CAN Application Protocol over EtherCAT).

Strict time control is, however, not easy to achieve. A great advantage of EtherCAT, which allows the master operation on commercially available processors (Commercial Off-the-Shelf, or COTS for short) and Ethernet controllers (MAC), opposes the demanded determinism to a certain extent. An all-software solution is insufficient, since the running time of the code to be executed cannot be fully predicted. Apart from the obvious influence of the conditional execution of the software created by the user, the processing time is additionally subject to fluctuations that depend on the processor architecture. Since these are not negligible, they will be dealt with in particular once again below.

The support of the subordinate Ethernet hardware, which dispatches the individual telegrams at precisely calculated times, is crucial for adherence to the telegram transmission times. Since this problem also plays a role outside of automation technology, widely available hardware can still be used. The first MACs with time-controlled transmission queues have already been brought onto the market by major manufacturers and this property seems to be included in many newer MACs. The Ethernet MACs employed from the Beckhoff CX Embedded PC series, as well as the external CU2508 as an Ethernet port multiplier provide hardware support for time-controlled transmission and can therefore be used for this specific application.



Exclusive CPU cores for TwinCAT real-time tasks

Modern processors can be a blessing and a curse for fast control tasks

PC control technology benefits from rapid technical progress in the IT and computer science worlds. However, this progress is not a gift which is "ready for use" without some effort: technical extensions must be made in hardware and software in order to properly use it in the world of automation. The real-time extension for Windows in TwinCAT and the use of Ethernet as a fieldbus in the form of EtherCAT strike the ideal balance for this.

Due to physical limits, the development of processors for increasing the computing power has shifted from continuously increasing clock frequencies to a multiplication of the number of computing cores per chip. We speak here of "chip multiprocessing" (CMP). In addition to dual core CPUs, four-core or eight-core units are also available today at a reasonable price. This development very much benefits software-based automation solutions such as TwinCAT 3, because they are able to distribute individual tasks depending on the number of available CPU cores. This means that major functions such as HMI, PLC control, PLC runtime and NC can be distributed to dedicated cores with less effort than ever today. TwinCAT 3 facilitates the utilization of multi-core systems through corresponding configuration and diagnostics tools. For example, the TwinCAT System Manager enables monitoring of real-time task runtimes and the manual configuration of priorities or task sequences. Tasks can be allocated statically to a particular core via configurable core affinities.

Due to the widespread use of main memory and processor cache (here: Last Level Cache), however, today's multi-core systems are subject to the limitation that cores can mutually affect each other. Under certain circumstances this may have an effect on the processing times of the code in the real-time environment, which must be taken into consideration when configuring the cycle times of real-time tasks in the sub-100 µs range. TwinCAT provides suitable measurement tools to evaluate the respective system behaviors.

The common use of system resources by CPU cores can also be disadvantageous in non-real-time applications. CMP systems were originally designed for high-performance processing by means of the optimal use of all system resources by multi-threading single applications. Due to the available computing performance, however, they can also run several operating systems on one computer in parallel with the aid of virtualization technologies. Mutual interference may under certain circumstances be disruptive here. In order to find a remedy for this problem, new hardware and software solutions for processors and operating systems were and are being developed and implemented by the manufacturers of the processors and operating systems. The decisive system properties to measure the performance of a computer are above all the size of the processor cache and the bandwidth of the main memory bus (in addition to the clock frequency of the processor core).

A software solution to optimize cache usage is the so-called "Cache Coloring" or "Page Coloring" tool. Here, the virtual memory management of the CPU generates a contiguous memory from the point of view of the cache, allowing more efficient use of the available capacity. This function is implemented, for example, in the dynamic memory management of an operating system or is contained in the virtual memory management of a hypervisor.

Of particular interest for the prioritization of applications running in parallel is a "Last Level Cache" that offers a QoS function (Quality of Service). Using this hardware property, cache areas can be reserved for high-priority applications or the cache size can be limited for low-priority applications. Currently, the ultimate resource in scalability and resource allocation in SMP systems (Symmetric Multiprocessing) is a "genuine" multiprocessor system in NUMA architecture (Non Uniform Memory Access). Not only does each processor have its own "Last Level Cache" here, but also its own local main memory where all processors have a common address space and are connected to one another by a bus. However, this is also the most complex and thus the most expensive of all solutions.

From version 3.1 on, TwinCAT offers the possibility to use processors or processor cores of a PC system exclusively for real-time applications. TwinCAT recognizes CPUs not used by Windows and can integrate them into the real-time system by configuration. For TwinCAT and its real-time applications, such as PLC and Motion Control, this takes place completely transparently, while the selected CPU is "invisible" to Windows. TwinCAT configures the real-time CPUs in such a way that the total address space is visible from the real-time CPUs but, conversely, private memory areas can be "hidden" from Windows. Virtualization functions have been consciously dispensed with for the implementation of this functionality in order to avoid an additional software layer, which could possibly

Author: Dr. Dirk Janssen,
Manager Software
Development System,
CNC and I/O, Beckhoff

Author: Ramon Barth,
Manager Software
Development System,
HMI and Real-Time, Beckhoff

lead to performance losses or latency times. This function can be used in all Windows versions starting from Windows XP SP2.

The advantages of this new TwinCAT property are obvious: primarily, the real-time applications are allotted more computing power, since the processor core concerned can be used by up to 100 %. Local resources of a core, e.g. the "Second Level Cache" or the memory of a NUMA architecture can be clearly assigned to the real-time part of the software. In addition, the static configuration of QoS caches of future processors is simplified. Based on the hardware properties described above, this solution permits the optimum scaling and usage of system resources for the respective application purpose. Cycle times as low as those in the 10-microsecond range illustrate the possibilities of technologies available today for users of TwinCAT and EtherCAT.

Conclusions

The relocation of fast control loops with very short reaction times into the central controller represents a challenge even to modern communication systems and control hardware. TwinCAT, the control software that can run on general-purpose PC hardware, and EtherCAT, the high-speed communication system based on standard Ethernet MACs, form the basis of the solution. However, it is only the optimum interaction between software and hardware and the optimum utilization of the latest hardware properties by TwinCAT 3 that create the necessary conditions for the reliable, deterministic operation of extremely fast control loops in a centralized controller. For the user the advantages include new possibilities to design advanced controller architectures in-house, to access considerably more information, to use a familiar development environment and – last, but not least – to reduce hardware costs.

Release of TwinCAT 3.1

With the release of TwinCAT Version 3.1 at Hannover Messe 2013, users now have the new "exclusive real-time cores" feature at their disposal. The time-controlled double transmit mode will be available as an update to TwinCAT 3.1 by the middle of the year. As a result, the cycle times and performance values described in this paper can be achieved on multiprocessor systems. The first systems with QoS cache will be available towards the end of 2013.

Further Information:
www.beckhoff.com/XFC
www.beckhoff.com/TwinCAT3