



Software breathes life into the PC-based controller

Beckhoff hit upon the idea of using a PC as a controller in 1986. 25 years ago a floppy disk drive was to be integrated into the controllers with Motorola processors developed up to that point by Beckhoff. Since this task was not quite so easy to accomplish, it was decided to use the PC as the CPU, whereby the floppy disk drive was automatically included. As further developments demonstrated, the use of the PC as the CPU (Central Processing Unit) for control and Motion Control offers many more advantages: once written, the software for control, motion and human-machine interface (HMI) can simply be adopted from one processor generation to the next. As a result, the increasing performance of processors on the PC market provides the user with ever increasing performance without requiring wholesale changes to the automation software. A PC-based controller needs a PC, which functions as the CPU, and a fieldbus, with which the input and output data are read and written. However, it is the software that first breathes life into the PC.

S1000 and S2000 – PLC and Motion Control under DOS

With the first PC-executable control software platforms, S1000 and S2000, Beckhoff had already made PLC and Motion Control functions, including CNC, available to its customers starting 1988. Thousands of machines for completely different fields were automated with the S1000 and S2000 software. The real-time capability of the PCs under the DOS operating system was made possible by a real-time extension developed by Beckhoff in-house. The programming of the actual controller took place in a programming language similar to Step-5. Even at that time it was possible to write parts of the application in C, such as the visualization, for example.

TwinCAT – The Windows Control and Automation Technology

Beckhoff switched to the Windows operating system and a new programming philosophy began in 1995. The whole world of automation, from the I/O level to PLC, Motion and CNC, has been at the disposal of the user since the introduction of TwinCAT in 1996. TwinCAT implements the real-time, i.e. the deterministic, virtually jitter-free execution of tasks on the PC with a Windows operating system. Beckhoff has taken care of the real-time development in-house from the outset and has always kept the know-how in the company's capable hands. Only the high deterministic of real-time make it possible to accomplish the control of an axis in less than a millisecond on a non-real-time operating system such as Windows XP.

On the basis of real-time, the drivers represent the connection to the outside world for the different fieldbuses. In addition to the Beckhoff fieldbus Lightbus, the first software versions already supported PROFIBUS, CANopen and DeviceNet; over 18 different fieldbus systems are supported today. On account of the concept of mapping between different process images that already existed in the first versions, it was simple to implement the development and integration of new fieldbuses. Each fieldbus has its own process image with inputs and outputs. The same applies to PLC and Motion Control: linking of variables must be carried out between software and hardware. These mappings are then updated in the cycle of the associated tasks, i.e. the values of the variables are copied from one process image to the other. The decoupling of software devices and fieldbuses makes it possible to change from one fieldbus to another simply by "re-wiring" – without having to adapt the software.

The actual application software is usually implemented in the PLC. Programming takes place with the standardized IEC 6113-3 languages. In addition to the two textual languages Instruction List (IL) and Structured Text (ST), the graphic languages Ladder Diagram (LD), Function Block Diagram (FBD) and Sequential Function Chart (SFC) are also available. On top of that, the IEC 61131-3 standard provides a software architecture that can be combined extremely well with the real-time characteristics of the TwinCAT system. The standard defines tasks and programs that can be connected directly to the real-time tasks.

Motion Control under TwinCAT means first of all the complete control of individual axes. Each axis is represented in the software by a virtual axis object, which takes care of setpoint generation, scaling and control. In the case





Thanks to the option to execute TwinCAT modules on specific cores of multi-core CPUs, TwinCAT 3 is future-proof and represents the foundation for the foreseeable future of PC-based control technology.

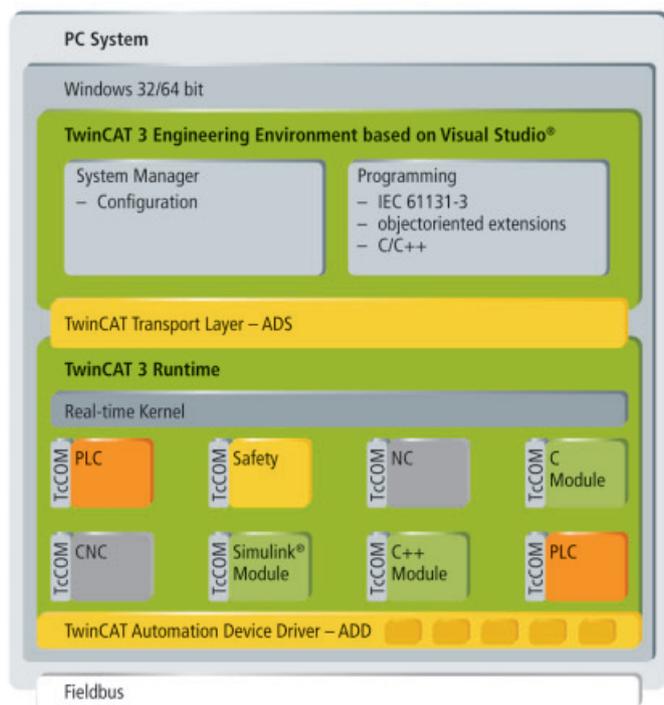
of simulation – which is always automatically available with TwinCAT – the output values of the axes are set the same as the input values. The axis object also assumes the task of coupling the real axis. Axis objects can abstract different axis types with different fieldbus interfaces. Abstraction layers enable the TwinCAT user to simply switch between different types of axes without modifying the PLC code. The movement of an axis, from the PLC application, can take place in the same way on a servo axis via EtherCAT or on a stepper axis via PROFIBUS.

In addition, further functions are available in TwinCAT for the coupling of axes: linear couplings are realized with the Gearing, non-linear with the TwinCAT libraries “Camming” or “Flying saw.” All of these couplings are controlled from the PLC via function blocks for Motion Control applications, which were standardized by the international organization, PLCopen. Beckhoff actively collaborated in the design of this virtual standard. The worldwide uniform appearance and behavior of these function blocks has led to rapid global acceptance of this standard. Work is currently proceeding on further standardizations.

The ultimate discipline in the field of Motion Control is the movement of interpolating axes, which involves the interpolating movement of several axes in a group. The fields of application are, for example, the classic machining centers for wood or metal, as well as portals and, of course, robots. Two levels are available for interpolation in the TwinCAT system. With TwinCAT NC I up to three axes can be interpolated at the same time and up to five auxiliary axes can be carried along on the path. The programming of the path, i.e. the movement of the axes in space, usually takes place according to the DIN 66025. Here, segments of the path are defined by means of so-called G-commands. However, programming can also be done in the PLC in the case of the NC I system and has the advantage that it is easier to learn for the PLC programmer. With the TwinCAT CNC platform, up to five axes can be interpolated at the same time. The CNC software package provides a series of options that are often used in machining centers, such as transformation functionality and high-speed cutting technology.

TwinCAT 3 – Revolution and Evolution

A completely new architecture – called eXtended Automation (XA) – was unveiled in 2010 with TwinCAT 3, the latest and most advanced generation of TwinCAT. The use of Microsoft Visual Studio® as a framework for the TwinCAT engineering tools is a revolution in the field of controls engineering. In TwinCAT 3, the globally renowned and accepted Visual Studio® solution is integrated with TwinCAT components for system configuration and PLC programming, enabling the use of existing languages such as C and C++ for real-time tasks. As a result, extensive existing source codes in C/C++ can be used to great effect. The .NET programming languages such as C# or VB.NET are also available in the same programming environment for non-real-time applications. That means universal engineering for the user: Visual Studio® forms the common framework for all languages and all configurators. In addi-



Architecture of the TwinCAT 3 automation software

tion, a series of add-ins is available for Visual Studio®, such as the source code control option. All source codes of all programming languages, including the configuration data, can be administered entirely in a source code database. This can significantly simplify version control and debugging for an entire team of programmers.

As other examples of well-known and respected programming solutions, the academic field is inconceivable without Matlab®/ Simulink®.

TwinCAT 3 offers the option to directly execute code from Matlab®/Simulink® in real-time. Several different toolboxes are available for Matlab®/Simulink® for controller development or for simulation, significantly simplifying development. C or C++ code is produced by the TwinCAT Target via the real-time workshop and compiled with the aid of the Microsoft C compiler. An XML-based description file is additionally generated. The description file and the compilation can now be integrated into any TwinCAT 3 system, even without having Matlab®/Simulink® installed. In the TwinCAT system the structure of the Matlab®/Simulink® network can be viewed and parameters can be modified.

Object orientation has only been present in PLC programming in a rudimentary form until now. This will completely change with TwinCAT 3: all functions are now available in order to successfully use object-oriented PLC code.



Dr. Josef Papenfort, TwinCAT Product Manager, Beckhoff Automation

Inheritance and virtual classes (interfaces) are also available in addition to the definition of classes and methods. If used correctly, these constructions should ensure that the quality of the PLC software increases and that engineering costs are lowered. Object orientation will also become manifest as a global standard in the third edition of IEC 61131-3.

The tried-and-tested configuration of the I/Os and axes using the TwinCAT system manager and the Motion Control system in the previous version has been adopted to a large extent in TwinCAT 3. The Beckhoff real-time extension for the Windows operating systems has likewise been adopted, but has been extended with the option to distribute functions to the various cores of a multi-core CPU. By means of configuration the user can intelligently decide which function should run on which core. The performance of the new multi-core CPUs is thus entirely at the disposal of the user.

With TwinCAT 3 Beckhoff has also pushed the standardization of real-time modules further. In a similar fashion to EtherCAT, all software modules are given standardized interfaces and a standardized state machine. The TwinCAT runtime modules obey the TwinCAT Component Object Model (TtCOM). Comparable to the COM for Windows programming, real-time modules can place their methods at the disposal of other modules. This standardization ensures that modules that were written in different languages can interact with one another in a runtime. Runtime modules can be directly and cyclically executed by tasks or can also be called by other modules. For example, a PLC module can directly use a controller that was written in Matlab®/Simulink®.

Summary

Beckhoff has been providing its customers with software packages for the automation of machines and plants for 25 years. In doing so, justice has been done to the progress associated with each new operating system and the increasingly powerful processors. With TwinCAT 3 in particular, Beckhoff has set a further milestone. Here, particular importance has been attached in the engineering to the efficient creation of application software. The optimum programming language can be selected for each problem, thus reducing the engineering costs.

TwinCAT www.beckhoff.com/TwinCAT