CODESYS V3.5 SP17
Features and Improvements
1. Engineering
2. Runtime
3. Visualization
4. Motion CNC Robotics
5. Fieldbus
6. Communication
Overview

- CFC improvements
- Security improvements
- Signed Libraries / Library Manager
- Further improvements
- Modularization of the CODESYS Development System
CFC improvements

- General space optimization

Before V3.5 SP17

As of SP17
CFC improvements

- Dark theme

- Pin rearrangement by drag and drop
CFC improvements

- Increased drop area for connections
- Keyboard shortcuts
- Dropping objects and symbols into the canvas
Signed libraries

- Supported since CODESYS V3.5 SP15
  - Simplification of former signing process with public/private key files
  - Now based on certificates

- Unsigned libraries: now with restrictions in CODESYS
  - No scripting in library documentation (since SP15)
  - Embedded files from unsigned libraries cannot be opened (new in SP17)

- Signed CODESYS libraries
  - In SP16: some
  - In SP17: all
    (Exception: individual libraries with dependencies to libraries < SP17)
Library Manager

- New function in the Library Manager: Display of all transitively referenced libraries in the project
Further improvements

- Compare: Allow the comparison of two arbitrary objects
- Logger: Filter by component ➞ Improved usability

- Preparation for new language features
  - List of reserved keywords as of SP17 e.g. LDATE, CHAR, WCHAR
  - Usage ➞ Compiler warnings
  - Compiler errors as soon as new language features are available
Further improvements

- **Acceleration of project load**
  - Language model stored as additional file (similar to compile information)
  - 15% - 35% faster loading of project files

- **General acceleration**
  - Blocking of certain operations avoided
  - Significantly faster reaction after project load / download

- **Re-work of the Build menu**
  - Easier usage
  - Former commands still available (via Customize command)
Further improvements

- **Watch list:**
  Structured display of inherited variables of an FB

- **Force list:**
  Display of recent process value of a forced variable
Further improvements

- Memory view: Search for specific values in the memory of the IEC application
  - Supported in Core Dump, too

- Trace
  - Simplified adding and editing of variables
  - Improved work with multiple diagrams
  - Improved usage of cursor
Modularization of the CODESYS Development System

- **CODESYS Development System**
  - Unmodifiable core system: "CODESYS Essentials“
    - Mandatory in order to work
  - Various individual modules
    - Installed with the standard setup
    - Own versioning (4.x.x.x)
  - Update of individual modules possible without changing “CODESYS Essentials”
    - Functional updates at any time
    - Immediate benefit from improvements
Modularization of the CODESYS Development System

- **CODESYS Installer**
  - New executable delivered with the setup
  - Grouping of required add-on components into so-called "installations"
  - Creating customized versions with only desired components
  - Simultaneous usage of versions for productive and experimental use
  - Reload of missing components possible ➔ Deployment server
Signed add-on Packages

- All packages with native CODESYS add-ons are signed
  ➔ Packages can be trusted.
- Check within the CODESYS Installer and the Package Manager
- Result similar to signed libraries
  - Green: Certificate / time stamp certificate by a CA
    ➔ Both certificates are valid and not expired.
    ➔ Package can be installed without any concern.
  - Yellow: Certificate is self-signed, not signed or expired, timestamp is not trustworthy.
    ➔ Package can only be installed after user confirmation.
  - Red: Certificate does not match the content.
    ➔ Package cannot be installed.
Overview

- OPC UA Server/Client
- Firmware update
- Modularization
- Update CODESYS Control RTE SL
- Further improvements
OPC UA Server / Client

- Support of OPC UA Methods
- Support of OPC UA Alarms & Conditions
- Support of user-defined information models / companion specifications
- Implementation of new security profiles defined in OPC UA 1.04

Find detailed information in the section Communication
Firmware update

- Backup / restore / update of the complete runtime system
  - Main use cases
    - Update of the CODESYS runtime with backup of application, certificates, and licenses
    - Backup / restore of the CODESYS runtime including backup of application, certificates, and licenses
  - Introduction of a new „Device management service“ running stand-alone or as component of a runtime with backend for access to local and remote PLCs
  - Introduction of a generic archive format (including manifest and packages)
    - Other firmware parts (OS parts, drivers, external software) could be included, but not scope of development so far
  - Client implementation in development
Modularization

- Independent releases of Linux-based single license runtimes
  - Release of new features / improvements independent from main version (CODESYS Essentials)
  - Version changed to V4.x.x.x
  - No compatibility break
  - Runtimes still based on V3.5.x.x Runtime Toolkit
Update CODESYS Control RTE SL

- **Multicore support**
  - Distribution of the application on multiple CPU cores
  - Selection of cores during setup
  - Task group configuration and core assignment within IDE

- **New drivers**
  - Support of new Peak CAN PCI Express cards
  - Support of Innodisk CAN PCI Express cards
Further improvements

- **CodeMeter®: Support of CodeMeter Universal Firm Code**
  - New generic firm code introduced by WIBU for all container types (dongle, SoftContainer, cloud)
  - Necessary for CodeMeter Runtime Support for Linux ARM
  - Due to compatibility issues only released for Linux ARM runtime

- **User Management**
  - Enable online user management by default

- **Integration of redundancy component in single-licensed runtimes**

- **Major update of CODESYS Control for PLCnext SL**
  - Bug fixes within Axioline configuration
  - Support of retain memory
  - Support of Smart Elements modules
  - Bug fix for problems with thread priorities

- **VxWorks**
  - Support of 64-bit targets on VxWorks 7
Further improvements

- **Logging of user actions**
  - Login successful / failed / logout
  - Download / create / start / stop / reset / delete application
Overview

- Overlay function in the CODESYS TargetVisu
- OPC UA Client support in CODESYS HMI SL
- HMI Redundancy
- Improvements of trend / alarm management
- Further improvements
Overlay function in the CODESYS TargetVisu

- **Dynamic internal rotation for all elements** (includes groups, frames, and native controls)
- **Time animations**
  - Smooth flying in of a menu independent of the visualization task
  - Smooth transitions for dialog fade in/out
  - Update of animated images such as GIFs/SVGs independently of the visualization task
- **Free positioning of elements in all dimensions**
  - Dialog opens over browser control
  - Crosshairs can be placed over camera image with visualization elements
- **Hardware accelerated drawing of TargetVisu (using GPU)**
- **Enabled by default in new projects - if supported by device**
- **Precondition: Qt installed on the target device**
- **For details, see video at** [https://youtu.be/C-7RQW-jmDk](https://youtu.be/C-7RQW-jmDk)
CODESYS HMI SL with OPC UA Client

- CODESYS HMI SL
  - Connecting to available OPC UA server
  - Browsing the published data
  - Read / write access to the data via visualization
HMI redundancy

- Synchronization of WebVisu client information between HMI master and slave
- In case of failure of HMI master: All WebVisu clients are automatically switched to the other HMI
  - Clients remain on pages/frames as before.
  - Login remains
- Immediate start of data transfer from the new HMI master to the controllers
- Central display of alarms either in HMI1 or HMI2 using remote alarm configurations
- Remote trend recording possible
- First release: no synchronization of recipes
HMI redundancy - configuration

- Visualization redundancy is not bound to CODESYS HMI SL.
- Enabled within the general redundancy configuration
- Automatic enabling via libraries and generated code possible
Improvements of Trend

- Interruptions in the trend recording (PLC Start/Stop) visible as gaps
- Possibility to stop the recording of single variables
- Dynamic limits for minimum / maximum of Y axes via IEC variables
- Trend legend with optional checkboxes to hide / show individual variables
- Possibility to reset the trend recording in the database
- Search / replace within the trend recording now possible
Improvements of alarm management

- Possibility to readout all information of a selected alarm
  Both active and historical alarms
- Complete alarm record can be reset via IEC application
Further improvements

- Support of the LDATE, LTOD, LDT, LTIME data types within visualization, alarm management, and recipe management
  - Support in the Date and Date / Time Picker elements as well as in the formatted output / input with %t
  - Support of alarm conditions and latch variables
  - Support of variables in recipe definitions

- Option to display scrollbars for scrollable elements like frame, table, combo box and alarm table when "Multitouch handling" is enabled.

- WebVisu FileTransfer - dialogs for the file transfer can be customized in size and design.
Further improvements

- Interface/callback: For tracking all inputs on elements
- New format string for TIME variables to enter REAL values
  Examples:
  - \%t[hh4] or \%t[HH4], hours with 4 decimal places
  - \%t[mm2] or \%t[m2], minutes with 2 decimal places, e.g. 40.25 minutes
  - \%t[ss0], seconds without decimal places.
- Recipe management: new option
  Recipes are / are not overwritten during download.
Further improvements

- Output of the number of used variables within all visualizations
1. Engineering
2. Runtime
3. Visualization
4. Motion CNC Robotics
5. Fieldbus
6. Communication
Overview

- SoftMotion on Standard PLCs
- Robotics improvements
- New supported drives
SoftMotion on Standard PLCs

- Previously: SoftMotion only on devices with special target type = 0x1006
- New: SoftMotion can be run on all PLCs.
- Activation via new command "Enable SoftMotion"
  - SoftMotion objects (CAM editor, CNC editor, axis groups) can now also be inserted on standard PLCs.
- Advantage for users: easier upgrade
  - of CODESYS compatible devices
  - of SoftPLCs from the CODESYS Store ("SL Runtimes")
Robotics improvements

- **Improved jogging**
  - Jogging = moving robot in manual mode
  - Improved orientation jogging: Robot rotation more predictable
  - Fixing of numerous bugs
  - Improvement of test coverage
  - Improved documentation and error messages

- Note: SMC_GroupJog2 replaces SMC_GroupJog (will be discontinued)
Robotics improvements

- **SMC_GroupInterruptAt**
  - Stopping at a certain position on the path
  - Otherwise like SMC_GroupInterrupt (pausing on the path)
New supported drives

- Bosch Rexroth Indradrive CS
- Delta Remote IO
- Delta ASDA A3/B3
- Beckhoff EL7041/7047
1. Engineering
2. Runtime
3. Visualization
4. Motion CNC Robotics
5. Fieldbus
6. Communication
Overview

- General improvements
- EtherNet/IP
- CAN / CANopen / J1939
- EtherCAT
- PROFINET
- MODBUS
General improvements

- Implicitly generated fieldbus task:
  - Warning when deleted
  - Easy way for restoring deleted task
EtherNet/IP

### Scanner
- LargeForwardOpen for Scanner and Adapter: Up to 65527 bytes per connection (before: 512 bytes)!
- EtherNet/IP Scanner: Possibility of enabling filling bytes in I/O mapping
- Stability improvements for multiple connections per slave
- GUI: Many bug fixes and performance improvements ➔ Update recommended!

### Adapter
- Possibility for sending outputs on start of bus cycle: Possible reason for doing so: less Jitter
- Extension of Ethernet Link Object: Conformance test with 1 GBit possible
General rework of online help with focus on diagnosis

- New structure: Configuration, Functions, Libraries, Diagnosis
- Diagnosis chapter added
  - Diagnosis in application
  - Diagnosis in user interface
  - Troubleshooting

CAN (CANopen/J1939)
CAN (CANopen/J1939)

Troubleshooting

**CANbus**

<table>
<thead>
<tr>
<th>Problem</th>
<th>Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>The CANbus is not in the device tree after the application download.</td>
<td>Check the status page and logger page of the CAN bus. You will find more information here.</td>
</tr>
<tr>
<td>Possible causes:</td>
<td></td>
</tr>
<tr>
<td>• The driver could not be opened.</td>
<td>- The network ID is set too large. In the CAN bus, a network ID was set for which there is no driver instance, if the interfaces, then valid values for the network ID are only 0 and 1.</td>
</tr>
<tr>
<td></td>
<td>- The driver component was not loaded in the runtime. Check whether the CAN driver component has been loaded correctly (see the Configuration).</td>
</tr>
<tr>
<td></td>
<td>- The memory is insufficient. Opening the CAN driver requires memory. Too little memory may be available.</td>
</tr>
<tr>
<td></td>
<td>• A bus error has occurred.</td>
</tr>
<tr>
<td></td>
<td>- The CAN chip signals a bus error. See the chapter “Diagnosis of Hardware Failures.”</td>
</tr>
</tbody>
</table>

**Diagnosis of Hardware Failures**

Most CAN problems are traced back to incorrect wiring or faulty CAN devices. Potential errors include the following:

- Missing or incorrectly dimensioned terminal resistors
- The bus has to be terminated exactly with a 120 Ohm terminal resistance at the beginning and at the end.
- Different baud rates
- All subscribers have to use the same baud rate.
- Short circuit between CANLow, CANHigh, CANGround, V+ CANLow and CANHigh interchanged
- Different lengths of CANHigh and CANLow
- CANHigh and CANLow are not a twisted pair. As a result, signals can be interrupted more easily.
- Different grounding sources for CANHigh
- Two-good grounding of the CAN bus cable
- When both ends of the shielding of the CAN bus cable are grounded, grounding loops can form which can cause interruptions.
- Bus cable too long

The maximum bus length depends on the set baud rate (see table).

<table>
<thead>
<tr>
<th>Bit Rate</th>
<th>Cable Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 kbps</td>
<td>6.7 km</td>
</tr>
<tr>
<td>20 kbps</td>
<td>3.3 km</td>
</tr>
<tr>
<td>50 kbps</td>
<td>1.6 km</td>
</tr>
<tr>
<td>125 kbps</td>
<td>500 m</td>
</tr>
<tr>
<td>250 kbps</td>
<td>250 m</td>
</tr>
<tr>
<td>500 kbps</td>
<td>125 m</td>
</tr>
<tr>
<td>1 Mbps</td>
<td>25 m</td>
</tr>
</tbody>
</table>

**Detecting hardware failures**

[Diagram showing CAN-H, CAN-L, CAN-HI, CAN-LI connections]
CAN (CANopen/J1939)

- Modularization
  - Stack library placeholders:
    Now part of the CANopenManager, CANopenSlave, and J1939Manager device
  - Using new protocol stack:
    Before: „Update device“ for PLC necessary ➔ New runtime required
    New: „Update device“ on the corresponding stack device (e.g. CANopenManager) possible
CANopen

- **CANopen Manager**
  - Improved COB-ID collision detection and intelligent suggestions
  - Additional check for EMCY, SYNC and TIME COB-ID
  - Check & Fix dialog suggests COB-ID formulas where possible.

- **CANopen Slave**
  - PDO/SRDO communication settings of I/O areas can be edited.
J1939

- **Import function for DBC**
  - Proprietary file format from Vector Informatik for defining CAN messages and ECUs
  - Widely used by Commercial Vehicle Manufacturers
  - Different use cases (see following slides)
- Use case 1: Import a complete network

**J1939**
Use case 2: Install as database
- **Removal of standard database**
  - Tooltip and online help notice the user!
  - User has to buy a DBC database e.g. from [https://www.csselectronics.com/](https://www.csselectronics.com/)

  Discount Code (5%): *J1939DBC_CODESYS*
EtherCAT

- Online page shows Link-Status
- Routing of EoE
- Overview page for device status
PROFINET

- **Performance optimizations**
  Up to 40% lower bus cycle times

- **Support dual- / multi-port Ethernet interfaces**
  ➔ Build controllers and devices with up to 8 Ethernet ports, e.g. for „Daisy Chain“ use case

- **Create PROFINET Device *.devdesc from GSDML**
  ➔ No more manual steps for manufacturers of PROFINET Device
PROFINET

- **PROFINET Device: programmatic configuration**
  - No Device-objects in static device tree
  - Completely programmatic, function block-based configuration (Library „ProfinetDeviceConfig“)

```c
VAR
  pnDevice : ProfinetDeviceConfig.ProfinetDeviceInstance;
  deviceIdent : ProfinetDeviceConfig.DeviceIdentification;

// Startup:
IF (NOT pnDevice.xInit) THEN
  deviceIdent.deviceID := 16#1017;   // Device Config
  deviceIdent.vendorID := 16#02CC;
  ... pnDevice.Plug(1n32);
  //Plug Modules
  pnDevice.Plug(out32x8);
  pnDevice.Plug(out32_param);
  ... pnDevice.ConfigureDevice(interfaceName, deviceIdent); //Configure
ELSIF (NOT (pnDevice.xStartup OR pnDevice.xOnline)) THEN
  pnDevice.Start();
ENDIF

//Acyclic services of device:
 pnDevice.ProcessServices();
```
- **IO-Link integration**
  - Editor for settings of IO-Link Device (IODD based)
  - Pre-processed IO data
  - Diagnosis with textual information from IODD
  - Scan/import of IO-Link sensors
MODBUS

- MODBUS function block library
  - Completely programmatic, function block-based configuration and communication
  - Client + server
  - TCP + serial
  - Various examples in CODESYS Forge
Overview

- OPC UA Server/Client improvements
- OPC UA PubSub communication
- IIoT Library Bundle improvements
OPC UA Server/Client improvements

- **Support of OPC UA methods**
  - Use case: call functions and methods of IEC applications and start complex functional processes via OPC UA

- **Support of OPC UA alarms & conditions**
  - Use case: send and acknowledge alarms to and with a capable OPC UA client
  - Events signal a change of the alarm status to the client
  - The client uses methods to interact with a specific instance of an alarm or condition.

- **Implementation of new security profiles defined in OPC UA 1.04 (CDS-62076)**
  - Basic125Rsa15 and Basic256 message security policies have been deprecated.
  - Two new security profiles: Aes128Sha256RsaOaep (for medium security needs) and Aes256Sha256RsaPSS (for high security needs)
OPC UA Server/Client Improvements

- Support of OPC UA Client configuration via Data Sources
OPC UA information models / companion specifications

- Support of user-defined information models / companion specifications
IoT Library Bundle improvements

- **Extension of the functional scope**
  - Now with function blocks for data exchange with Google Cloud IoT Core
  - License for Google Cloud IoT Core Client SL included in the **IIoT Libraries SL**
Thank you for your attention!