



KPA EtherCAT Studio is a second generation of a user-friendly tool for EtherCAT bus configuration and diagnostics. This lightweight, high-performance standalone Windows application natively supports all the advanced features of KPA EtherCAT Master.

Key Features

- For configuring EtherCAT bus, EtherCAT slave devices and for online diagnostics
- Intuitive, user-friendly interface
- Based on KPA Studio Development Framework (SDF)
- Three basic types of license: Basic, Standard or Premium

	Basic	Standard	Premium	+	Extensions
Studio					
Scaling Signal Values		Automation interface			
Slaves Library Studio/Emergency Output Watch Viewer		Master Output			
Plug-ins for configuring specific devices:					
CANopen Master/slave (EL6751) Profibus slave (EL6731) Bridge terminals (EL6692/EL6695)		Modular Devices	Virtual Serial COM		
Tools					
Topology Viewer Snapshot Viewer Chart Viewer Diagnostic scanner S2S Communication Editor		EEPROM Editor	Comparing tools Data logging tools		
Common					
DC Tasks/Sync Units			Power calculation via LVDS		
Master					
Cyclic Commands ENI Export External task		PI Alignment Master Watchdog Master Mailbox. AoE Master Mailbox. EoE ENI Export. XSLT Scheme support	Cable Redundancy		
Slave					
Mailbox. CoE Mailbox. FoE		Slave Watchdog Slave Mailbox. AoE Slave Mailbox. EoE Mailbox. SoE Configuring Init Commands, FMMU/SM Direct Memory access Direct EEPROM access Explicit identification	Hot Connect		

MDP Profibus Master
Gateway Profile 3100

Core Features

Scaling signal values

It is a possibility to change a representation of signal's value. Scaling signal values is available for integer signals with 16- or 32-bit length. This attribute is used to specify the range of the input data.

Studio/Master/Emergency Output

This feature transmits additional information about processes in Studio, Master and on the bus.

Studio Output displays messages from KPA Studio (Studio core, plug-ins) with possibility of customization and Master/slave traces split by severity.

Master Output displays raw Master traces that is got from Master.

Emergency Output displays messages which are sent by slaves in case of emergency (transmitted via CoE emergency service).

Watch Viewer

Watch Viewer allows to monitor changing of selected signals, that allow to control them in separate window.

Slaves Library

Slaves Library is a list of available ESI files which can be used to build up Project configuration. There is an opportunity to work with several Slaves Library collections, change active collection by using custom collection and manage Slaves Library view by setting user attributes for slaves. It is possible to use as a source for the collection: KPA ESIs Storage, ESI file, ESIs folder, or Installed ESIs folder and save it as KPA ESIs Storage.

Configuring specific devices

- EL6751:
 - CANopen Master plug-in - This plug-in allows the user to adapt CANopen Master configuration got by scanning CAN bus.
 - CAN Interface plug-in is intended to configure device in CAN Interface (raw CAN) mode to be used with/without CAN DBC driver on master side. Data exchange through the driver can be done either through the specified signals (adding messages from a DBC file) or by sending/receiving unspecified messages via driver's queue.
- EL6731:
 - MDP plug-in for Profibus Master Gateway Profile 3100 - to configure a slave supporting Profibus Master (3100) profile. Позволяет конфигурировать девайсы посредством гсд файлов. Конфигурирование мб выполнено в автоматическом режиме если устройство поддерживает AoE протокол.
- EL6692/EL6695
 - M2M plug-in - is intended to configure the both sides of the EtherCAT bridge terminal (Primary and Secondary).
- EL6731-0010
 - Profibus slave plugin - is intended to configure device for communication between EtherCAT and PROFIBUS.

Modular devices

MDP Modules is intended to configure slave supporting Modular Device Profile with modules specified in its ESI file.

Virtual Serial COM

KPA offers an easy way of data exchange with devices with serial interface (RS232/RS422/RS485).

KPA EtherCAT Master supports Virtual COM communication. It allows you to communicate with the serial device (connected via Serial to EtherCAT gateway device) from your Windows application via Virtual COM port. Master transmits data got from the serial terminal / the Virtual COM device to the recipient.

To set such communication Master requires the KPA Virtual COM driver being installed in the Windows and the corresponding feature must present in Master's license. KPA Virtual COM driver is provided as a separate module and can be installed independently of Master. It is not required additional configuring with COM settings (baud rate, data bits, parity etc.).

Tools

- Topology Viewer - displays Network and Master configurations simultaneously. The tool displays slaves' state. If there is a problem with slave(s) in configuration, it also will be marked by a corresponding icon and the problems will be displayed in the slave tooltip.
 - Snapshot Viewer - Signal values are gathered during operation and stored to make a snapshot. The viewer then displays the snapshot. Taking snapshots reduces overall load on the system in order not to affect hard real-time performance.
-

- Chart Viewer allows monitoring multiple channels from one or several slaves simultaneously.
- Diagnostic Scanner - scans EtherCAT network and executes diagnostics for possible slave and connection problems.
- S2S Communication Editor allows to map input process data of one device to output process data another device without wiring (via the Master). Copying of the data is handled by the Master stack.
- EEPROM Editor allows to read EEPROM, to generate ESI from a binary file or directly from the slave, view it by bits or in Hex Editor, to navigate through the EEPROM using quick access to certain parameter, to edit it. Besides working with read data, the user may create new EEPROM categories and add new data. Modified EEPROM can be saved to a binary file, ESI or directly to the slave.
- Comparing tools:
 - Projects Comparator. Comparing two projects. It is possible to compare the project opened in Studio with a project loaded from file, or just two projects from files. Found differences are highlighted with different colors which are the user can set by yourself.
 - ESI and EEPROM Comparator. The tool enables comparison of EEPROM content acquired directly from a Slave device or provided in a binary file against the content of a file either assigned to the Slave or chosen by a user. Differences are highlighted with customizable colors; properties values are displayed.
- Data logging tools:
 - Run-time Data Logger. Manages the data logging on the Master side. The Studio retrieves the stored data and enables building of charts for signals picked from a defined set.
 - Data Logger manages the data logging on the Master side.
 - Frame logger. In order to get time information on EtherCAT packets: duration of packet assembling, time of received frame parsing, time of setting frame to queue for sending, etc. This information may be visualized with the help of Frame Logger tool.

Distributed Clocks

The feature enables synchronization of the Master and all slave devices on the bus. It is configured on the both sides Master and slaves.

Task/Sync Units

Provides an opportunity to configure Process data update by using tasks and sync units and update particular blocks of Process image in certain time.

LVDS power capabilities

Monitoring of power supply and consumption of LVDS slaves on the bus in order to avoid power lack issues in the configuration.

Cyclic commands

Studio allows to configure and set the values of cyclic commands to send and receive data from bus.

There are three types of cyclic commands available for selection:

- Common logical read/write command;
- Separate logical read and logical write commands;
- Separate physical read and physical write commands.

ENI Export

Creation of configuration file for master (according to EtherCAT Technology Group (ETG) specification (ENI)).

Using KPA Studio it is possible to prepare Master configurations to be used in other systems (for example CODESYS).

KPA Studio provides the following types of Master configuration export:

- Export Master Configuration KPA – creates an XML file, which can be imported for further usage by the Master, KPA extensions are included;
 - Export Master Configuration ETG Standard – creates an XML file according to the ETG specification;
 - Export CODESYS PLC Configuration – creates a file, which can be imported into CODESYS configuration (includes the Master module configuration);
 - Export CODESYS PLC Configuration (master module) – creates a file, which can be imported into CODESYS configuration (without PLC configuration);
 - Export encoded configuration – creates an XML file in UTF-8 encoding;
 - Export C header file – creates a C header file with definitions of offsets and lengths of process image signals, which can be used to simplify procedure of process image usage while development of control application/PLC;
-

- Export C header file including slave properties – creates a C header file with definitions of offsets, lengths and types of signals and slaves' properties (VendorID, ProductCode, etc.); also, can be used to simplify procedure of process image usage while development of control application/PLC;
- Export IEC61131-3 PLC POU (Program Organization Unit) – creates a file, which can be imported into other Master configurations using software which supports IEC61131-3 standards (for example, Structured Text language);
- Export XML Format for IEC 61131-3 PLCopen – creates a file, which can be imported into other Master configurations using software which supports IEC61131-3 PLCopen standard;
- Export Master Configuration (comments without CDATA) – creates an XML file as Export Master Configuration KPA, but without CDATA in comments.

This feature allows export custom ENI files based on custom XLST scheme support.

External Task

KPA Studio allows running external tasks (specific applications) on Master side. Sawtooth application, which simulates saw-shaped signals on outputs to show a snapshots diagram. A signal from slave's analog input is multiplied by a special factor and sent to output without wiring (through memory). The signal sent to output is shaped as a saw.

KPA Studio allows running external tasks (specific applications) on Master side to manage EtherCAT bus. External tasks are custom developed depending on customer needs.

Process Image alignment

Studio allows you to set different types of PI alignment depending on your configuration. You may choose between the five options in the **Alignment mode** list:

- No alignment;
- Bit alignment;
- CODESYS (PowerPC) alignment;
- CODESYS (PowerPC) with Bit-Byte alignment;
- CODESYS (x86) alignment.

The options listed above present the ways of addresses types alignment in Process Image as different PLC systems acquire different addressing modes. For example, a four-byte signal offset should start with a four-fold address value. Bit alignment enables grouping of one or more bit long signals into one or several sequential bytes.

Alignment is set on Process Image Tab, that allows the user to set variables' values, get a snapshot and watch the fluctuation of variables' value in Chart window. Process Image Tab – allows the user to set and force/unforce variables' values, get a snapshot and watch the fluctuation of variables' value in Chart window.

Master/Slave Watchdog

Master

Master watchdog – enables/disables the Master watchdog function. If there is PLC program crash or it is hanging and no output value has been set within this timeout, a default or last value (depending on the signal's settings or Master settings) will be set by Master to all outputs.

Slave

The slave's watchdogs are used for observing communication and returning to a safe state in case of an error. The ESCs (EtherCAT Slave Controller) support up to two internal watchdogs:

- SM – Sync Manager watchdog, which is used for monitoring process data accesses by Master (e.g. Master read process data).
- PDI – Process Data Interface watchdog, which is used for monitoring of firmware viability.

Cable Redundancy

Cable redundancy makes possible several things that used to be bottle-necks of conventional field bus systems built up on closed ring principles, such as cable feedback for problem finding and system availability in case of cable break. If the ring is interrupted at some point, both branches can be still reached.

AoE

ADS over EtherCAT (AoE) is a standard, client-server Mailbox protocol defined by the EtherCAT specification. It is used to build storage area networks (SANs) with low-cost, standard technologies.

AoE is routable and supports the contemporary handling of parallel services, and therefore is particularly suitable for the access to sub-networks via gateway devices (master devices which are required to operate gateway slave devices should therefore support AoE). Due to its routability, AoE is also defined as standard protocol for the communication between an EAP (EtherCAT Automation Protocol) network and single EtherCAT fieldbus segments.

Both master and slave stacks exist which support basic AoE services, enabling a light and fast implementation of this mailbox protocol in EtherCAT devices.

CoE

Event driven standard of access to CANopen Object dictionary and exchange of CANopen Emergency and PDO/SDO messages. With the CoE protocol (CAN application protocol over EtherCAT), EtherCAT provides the same communication mechanisms as in CANopen®-Standard EN 50325-4: Object Dictionary, PDO Mapping (Process Data Objects) and SDO (Service Data Objects) – even the network management is similar. This makes it possible to implement EtherCAT with minimal effort in devices that were previously outfitted with CANopen, and large portions of the CANopen Firmware are even reusable. Optionally, the legacy 8-byte PDO limitation can be waived, and it's also possible to use the enhanced bandwidth of EtherCAT to support the upload of the entire Object Dictionary. The device profiles, such as the drive profile CiA 402, can also be reused for EtherCAT.

EoE

The EtherCAT technology is not only fully Ethernet-compatible, but the Ethernet over EtherCAT (EoE) protocol tolerates other Ethernet-based services and protocols on the same physical network. The term Ethernet is also frequently associated with data transfer in IT applications, which are based on a TCP/IP connection. Using the EoE protocol any Ethernet data traffic can be transported within the EtherCAT segment. The Ethernet frames are tunneled through the EtherCAT protocol, similarly to the internet protocols (e.g. TCP/IP, VPN, PPPoE (DSL), etc.), which makes the EtherCAT network completely transparent for Ethernet devices. The device with the Switchport property takes care of inserting TCP/IP fragments into the EtherCAT traffic and therefore prevents the network's real-time properties from being affected. Additionally, EtherCAT devices may also support Ethernet protocols (such as HTTP) and can therefore behave like a standard Ethernet node outside of the EtherCAT segment. The master device acts as a Layer-2-switch that sends the frames via Eoe to the corresponding nodes according to their MAC addresses. This way, all internet technologies, such as an integrated web server, E-mail, FTP transfer, etc. can also be implemented in an EtherCAT environment.

FoE

FoE (File Access over EtherCAT) is a mailbox protocol generally intended to transfer file data on an EtherCAT network in both directions, and as such it can be used in any state where the mailbox communication is active (PreOP, SafeOP, OP). The most common use case for FoE is to download a new firmware for update, when the slave is in Bootstrap state.

EtherCAT provides the FoE protocol for simple file access. For example, the device runs in a boot loader state to support a firmware download to the host controller via the EtherCAT network. Standardizes firmware download to devices is therefore possible, even without the support of TCP/IP.

SoE

Servo drive profile over EtherCAT (SoE) communication services are used to access the IDNs of a SoE slave. The SoE drive profile is part of an extra master application, not of the master itself.

If the EtherCAT slave supports the Servo Profile over EtherCAT (SoE) protocol the additional tab "SoE" appears in the Mailbox.

Servo drive profile over EtherCAT® (SoE) is a protocol of EtherCAT interface for IEC 61491 servo drive profile.

SoE allows to integrate devices based on IEC 61491 technologies in an EtherCAT bus. It is executed by getting access to the drive parameters and data that defined in IEC 61491. Non-cyclic data exchange is implemented by SoE protocol with help of the EtherCAT SoE Service Channel (SSC) (equivalent of the IEC61491 Service Channel). It allows to access to identifiers (IDN) and their elements.

Configuring Initialization commands

Init commands initialize and control processes of communication with a slave (according to a specification). When a device is configured via Mailbox Initialization Commands, it is possible to set the Operational mode, speed of data exchange, a set of data for exchange, etc. Initialization commands is generated by Studio and can be disabled, added or deleted by customer.

Configuring FMMU/SM

4 Gb address space is one of the EtherCAT advantages. To use it the process data of the dedicated device must be mapped to the EtherCAT logical address space. This function is performed by FMMU (Fieldbus Memory Management Units).

On the other hand, the process data itself is located in the physical address space of the controller, access to which is controlled and synchronized by SyncManager (SM). Within the Studio the corresponding technological “pillars” can be configured in the way necessary for the user.

Direct Memory access

Studio allow directly read/write memory register of EtherCAT slave controller. также Studio allows mapping memory registers to Process Image by adding new or using known registers. This allows mapping registers to Process Image by adding new or using known registers.

Direct EEPROM access

EEPROM – Electrically Erasable Programmable Read Only Memory – energy-independent slave memory. The EtherCAT Slave Controller (ESC) loads its configuration data (e.g. PDI configuration) during start up from the Slave Information Interface (SII) which usually is implemented as EEPROM.

Studio allows read/write data about slave in/to EPROM and make changes to the slave description by using EEPROM editor tool.

Hot Connect

Hot Connect function allows to connect additional devices flexibly. They can be optionally removed from or added to the data traffic before the start or during operation of the system. Such actions will not cause the damage of the overall functionality. This is called “flexible topology” or Hot Connect.
