I/O Terminal

Installation Manual

Revision 1.2, May 2012
Valid for AKD PDMM Hardware Revision DB

AKT-ECT-000-000 ETHERCAT Coupler

Keep all manuals as a product component during the life span of the product.
Pass all manuals to future users / owners of the product.

Because Motion Matters™

Record of Document Revisions

Revision	Remarks
1.0	Preliminary edition
1.1	Added mechanical drawing to Appendix A. For more information, see “Appendix A” page 55.

IMPORTANT NOTICE

Copyright© Kollmorgen™2010-2012. All rights reserved. Kollmorgen holds the copyright to this manual. All rights are reserved and no part of this publication may be reproduced or transmitted in any form or by any means without prior written consent from Kollmorgen.

Disclaimer

The information in this manual was accurate and reliable at the time of its release. However, Kollmorgen. reserves the right to change the specifications of the product described in this manual without notice at any time.

This document contains proprietary and confidential information of Kollmorgen. The contents of the document may not be disclosed to third parties, translated, copied or duplicated in any form, in whole or in part, without the express written permission of Kollmorgen.

Registered Trademarks

Kollmorgen is a registered trademark of Danaher Corp.

Technical changes which improve the performance of the device may be made without prior notice! Printed in the Federal Republic of Germany. All rights reserved. No part of this work may be reproduced in any form (by printing, photocopying, microfilm or any other method) or stored, processed, copied or distributed by electronic means without the written permission of Kollmorgen. All other proprietary names mentioned in this manual are the trademarks of their respective owners.

May 2012

1 SAFETY PRECAUTIONS

This chapter provides safety information for the I/O terminal.

1.1 Safety Rules

The appropriate staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

1.2 State at Delivery

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify any liability from Kollmorgen.

1.3 Personnel Qualification

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.

1.4 Description of Notes and Warnings

The following notes and warnings are used in this manual. They are intended to alert the reader to the associated safety instructions.

Danger — This note is intended to highlight risks for the life or health of personnel.

Warning — This note is intended to highlight risks for equipment, materials or the environment.

Note — Indicates information that contributes to better understanding.

2 BASICS

This chapter provides an overview of the EtherCAT basics.

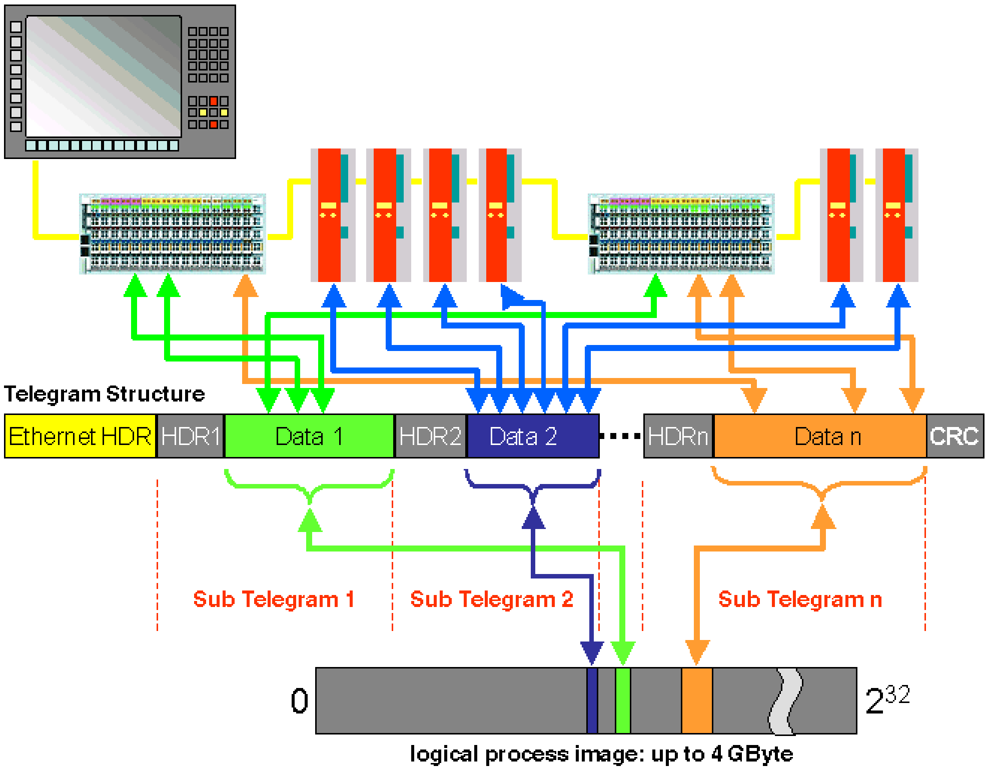
2.1 System Properties

This section covers the basic system properties.

2.1.1 Protocol

The EtherCAT protocol is optimized for process data and is transported directly within the Ethernet frame thanks to a special Ether-type. It may consist of several sub-telegrams, each serving a particular memory area of the logical process images that can be up to 4 gigabytes in size. The data sequence is independent of the physical order of the Ethernet terminals in the network; addressing can be in any order. Broadcast, Multicast and communication between slaves are possible. Transfer directly in the Ethernet frame is used in cases where EtherCAT components are operated in the same subnet as the control computer.

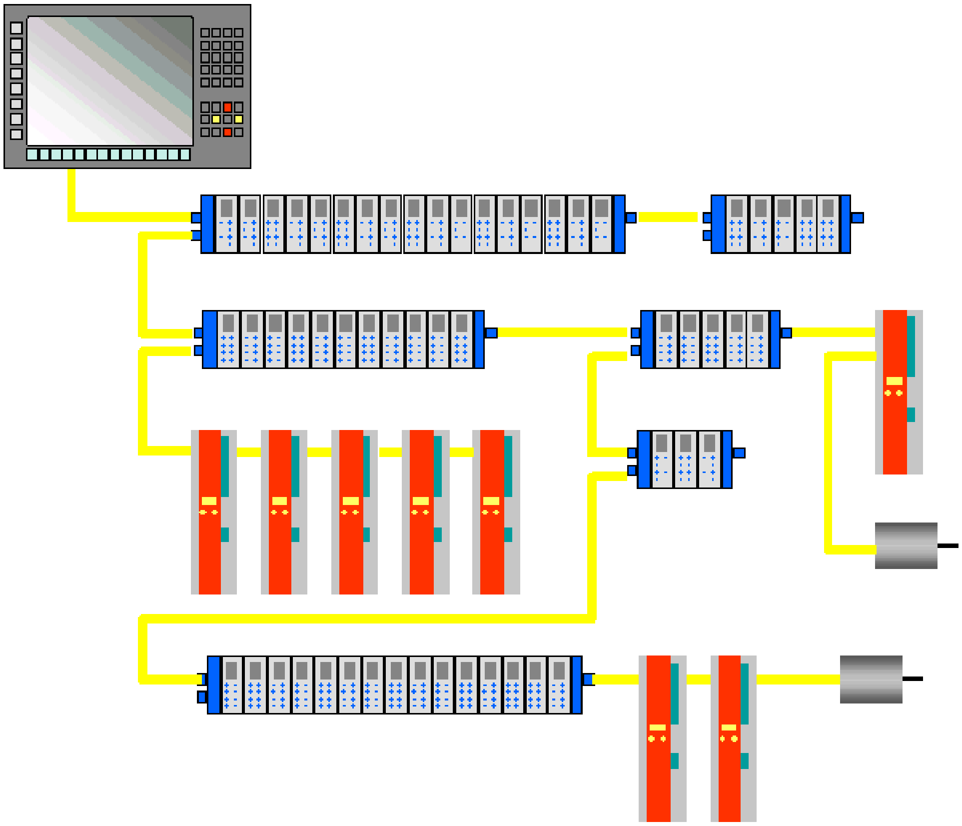
However, EtherCAT applications are not limited to a subnet: EtherCAT UDP packs the EtherCAT protocol into UDP/IP datagrams. This enables any control with Ethernet protocol stack to address EtherCAT systems. Even communication across routers into other subnets is possible. In this variant, system performance obviously depends on the real-time characteristics of the control and its Ethernet protocol implementation. The response times of the EtherCAT network itself are hardly restricted at all: the UDP datagram only has to be unpacked in the first station.



Protocol structure: The process image allocation is freely configurable. Data are copied directly in the I/O terminal to the desired location within the process image: no additional mapping is required. The available logical address space is with very large (4 GB).

2.1.2 Topology

Line, tree or star: EtherCAT supports almost any topology. The bus or line structure known from the fieldbuses thus also becomes available for Ethernet. Particularly useful for system wiring is the combination of line and branches or stubs. The required interfaces exist on the couplers; no additional switches are required. Naturally, the classic switch-based Ethernet star topology can also be used.



Maximum wiring flexibility — With or without switch, line or tree topologies, can be freely selected and combined. The complete bandwidth of the Ethernet network - such as different optical fibers and copper cables - can be used in combination with switches or media converters.

2.1.3 Distributed Clocks

Accurate synchronization is particularly important in cases where spatially distributed processes require simultaneous actions. This may be the case, for example, in applications where several servo axes carry out coordinated movements simultaneously.

The most powerful approach for synchronization is the accurate alignment of distributed clocks, as described in the new IEEE 1588 standard. In contrast to fully synchronous communication, where synchronization quality suffers immediately in the event of a communication fault, distributed aligned clocks have a high degree of tolerance vis-à-vis possible fault-related delays within the communication system.

With EtherCAT, the data exchange is fully based on a pure hardware machine. Since the communication utilizes a logical (and thanks to full-duplex Fast Ethernet also physical) ring structure, the mother clock can determine the run-time offset to the individual daughter clocks simply and accurately - and vice versa. The distributed clocks are adjusted based on this value, which means that a very precise network-wide time base with a jitter of significantly less then 1 microsecond is available.

However, high-resolution distributed clocks are not only used for synchronization, but can also provide accurate information about the local timing of the data acquisition. For example, controls frequently calculate velocities from sequentially measured positions. Particularly with very short sampling times, even a small temporal jitter in the displacement measurement leads to large step changes in velocity. With EtherCAT new, extended data types are introduced as a logical extension (time stamp and oversampling data type). The local time is linked to the measured value with a resolution of up to 10 ns, which is made possible by the large bandwidth offered by Ethernet. The accuracy of a velocity calculation then no longer depends on the jitter of the communication system. It is orders of magnitude better than that of measuring techniques based on jitter-free communication.

2.1.4 Performance

EtherCAT reaches new dimensions in network performance. Protocol processing is purely hardware-based through an FMMU chip in the terminal and DMA access to the network card of the master. It is thus independent of protocol stack run-times, CPU performance and software implementation. The update time for 1000 I/Os is only 30 µs - including terminal cycle time. Up to 1486 bytes of process data can be exchanged with a single Ethernet frame - this is equivalent to almost 12000 digital inputs and outputs. The transfer of this data quantity only takes 300 µs.

The communication with 100 servo axes only takes 100 µs. During this time, all axes are provided with set values and control data and report their actual position and status. Distributed clocks enable the axes to be synchronized with a deviation of significantly less than 1 microsecond.

The extremely high performance of the EtherCAT technology enables control concepts that could not be realized with classic fieldbus systems. For example, the Ethernet system can now not only deal with velocity control, but also with the current control of distributed drives. The tremendous bandwidth enables status information to be transferred with each data item. With EtherCAT, a communication technology is available that matches the superior computing capacity of modern Industrial PCs. The bus system is no longer the bottleneck of the control concept. Distributed I/Os are recorded faster than is possible with most local I/O interfaces. The EtherCAT technology principle is scalable and not bound to the baud rate of 100 MBaud – extension to GBit Ethernet is possible.

2.1.5 Diagnostics

Experience with fieldbus systems shows that availability and commissioning times crucially depend on the diagnostic capability. Only faults that are detected quickly and accurately and which can be precisely located can be corrected quickly. Therefore, special attention was paid to exemplary diagnostic features during the development of EtherCAT.

During commissioning, the actual configuration of the I/O terminals should be checked for consistency with the specified configuration. The topology should also match the saved configuration. Due to the built-in topology recognition down to the individual terminals, this verification can not only take place during system start-up, automatic reading in of the network is also possible (configuration upload).

Bit faults during the transfer are reliably detected through evaluation of the CRC checksum: The 32 bit CRC polynomial has a minimum hamming distance of 4. Apart from breaking point detection and localization, the protocol, physical transfer behavior and topology of the EtherCAT system enable individual quality monitoring of each individual transmission segment. The automatic evaluation of the associated error counters enables precise localization of critical network sections. Gradual or changing sources of error such as EMC influences, defective push-in connectors or cable damage are detected and located, even if they do not yet overstrain the self-healing capacity of the network.

2.2 The Bus Terminal System

Up to 64 Bus Terminals each having 2 I/O channels for each signal form

The Bus Terminal system is the universal interface between a fieldbus system and the sensor / actuator level. A unit consists of a Bus Coupler as the head station, and up to 64 electronic series terminals, the last one being an end terminal. For each technical signal form, terminals are available each having two I/O channels, and these can be mixed in any order. All the terminal types have the same mechanical construction, so that difficulties of planning and design are minimized. The height and depth match the dimensions of compact terminal boxes.

Decentralized wiring of each I/O level

Fieldbus technology allows more compact forms of controller to be used. The I/O level does not have to be brought to the controller. The sensors and actuators can be wired decentrally, using minimum cable lengths. The controller can be installed at any location within the plant.

Industrial PCs as controllers

The use of an Industrial PC as the controller means that the operating and observing element can be implemented in the controller's hardware. The controller can therefore be located at an operating panel, in a control room, or at some similar place. The Bus Terminals form the decentralized input/output level of the controller in the control cabinet and the subsidiary terminal boxes. The power sector of the plant is also controlled over the bus system in addition to the sensor/actuator level. The Bus Terminal replaces the conventional series terminal as the wiring level in the control cabinet. The control cabinet can have smaller dimensions.

Bus Couplers for all usual bus systems

The Bus Terminal system unites the advantages of a bus system with the possibilities of the compact series terminal. Bus Terminals can be driven within all the usual bus systems, thus reducing the controller parts count. The Bus Terminals then behave like conventional connections for that bus system. All the performance features of the particular bus system are supported.

Assembly on standardized mounting rails

The easy, space-saving, assembly on a standardized mounting rail (EN 50022, 35 mm), and the direct wiring of actuators and sensors, without cross-connections between the terminals, standardizes the installation. The consistent labeling scheme also contributes.

The small physical size and the great flexibility of the Bus Terminal system allows it to be used wherever a series terminal is also used. Every type of connection, such as analog, digital, serial or the direct connection of sensors can be implemented.

Modularity

The modular assembly of the terminal strip with Bus Terminals of various functions limits the number of unused channels to a maximum of one per function. The presence of two channels in one terminal is the optimum compromise of unused channels and the cost of each channel. The possibility of electrical isolation through potential feed terminals also helps to keep the number of unused channels low.

Display of the channel state

The integrated LEDs show the state of the channel at a location close to the sensors and actuators.

Standard-Bus

The Standard Bus is the data path within a terminal strip. The Standard Bus is led through from the Bus Coupler through all the terminals via six contacts on the terminals' side walls. The end terminal terminates the Standard Bus. The user does not have to learn anything about the function of the Standard Bus or about the internal workings of the terminals and the Bus Coupler. Many software tools that can be supplied make project planning, configuration and operation easy.

Potential feed terminals for isolated groups

The operating voltage is passed on to following terminals via three power contacts. You can divide the terminal strip into arbitrary isolated groups by means of potential feed terminals. The potential feed terminals play no part in the control of the terminals, and can be inserted at any locations within the terminal strip.

Up to 64 terminals can be used within one terminal strip. This count does include potential feed terminals, but not the end terminal.

The Interfaces

A Bus Coupler has six different methods of connection. These interfaces are designed as plug connectors and as spring-loaded terminals.

2.3 CoE Interface

All intelligent EtherCAT slaves (with micro-controller) supports the CoE interface. It includes the following components.

2.3.1 EtherCAT State Machine

The state of the EtherCAT slave is controlled via the EtherCAT State Machine.

2.3.2 Object Directory

All EtherCAT slaves supporting the CoE interface have an object directory containing all parameter, diagnostic, process or other data that can be read or written via EtherCAT.

The object directory can be read via the SDO information service. It is included in the device description file. All EtherCAT slaves should support the SDO information service at least to such an extent that the compact object description of each object can be read (from index 0x1000). This object description contains the data type, the length, the access rights and information as to whether the object can be mapped in a PDO (and therefore can be used as process data).

2.3.3 Process Data

The EtherCAT data link layer is optimized for fast transfer of process data. It contains information about how the device process data are linked to the EtherCAT process data and how the application on the device is synchronized with the EtherCAT cycle.

The process data are allocated (mapped) via PDO mapping and sync manager PDO assign objects. They describe which objects are transferred as process data from the object directory via EtherCAT. All EtherCAT slaves support reading of the PDO mapping and the sync manager PDO assign objects as a minimum. If the EtherCAT slave process data mapping can be configured by the EtherCAT master, PDO mapping or sync manager PDO assign objects should also be write-enabled.

Sync manager communication objects define the cycle time for transferring the associated process data via EtherCAT and how they are synchronized for the transfer.

2.3.4 EtherCAT Start-Up

The EtherCAT start-up describes the interrelationship between EtherCAT state machine, process data mapping and device parameter settings during start-up of EtherCAT network.

2.3.5 Command Objects

Command objects can be used to start actions in a device for which a result is only available after a certain time.

2.3.6 Emergencies

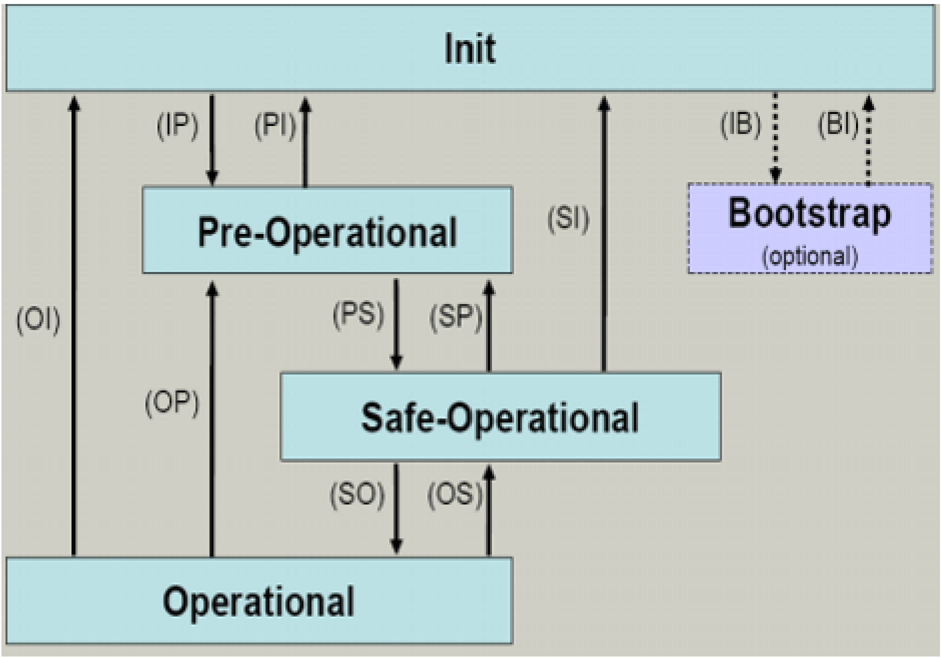
Emergencies can be used to transfer diagnostic information or process events with time stamp for recording by an event logger, for example.

On the other hand, status messages indicating the current state of the device, which is required at the same time as the actual process data in the control application, should be transferred directly with the process data.

2.4 EtherCAT State Machine

The state of the EtherCAT slave is controlled via the EtherCAT State Machine (ESM). A distinction is made between the following states (see. Fig. 1):

- Init
- Pre-Operational
- Safe-Operational and
- Operational
- Boot



2.4.1 Init

After switch-on the EtherCAT slave in the *Init* state. No mailbox or process data communication is possible. The EtherCAT master initializes sync manager channels 0 and 1 for mailbox communication.

2.4.2 Pre-Operational (Pre-Op)

During the transition between *Init* and *Pre-Op* the EtherCAT slave checks whether the mailbox was initialized correctly.

In *Pre-Op* state mailbox communication is possible, but not process data communication. The EtherCAT master initializes the sync manager channels for process data (from sync manager channel 2), the FMMU channels and, if the slave supports configurable mapping, PDO mapping or the sync manager PDO assignment. In this state the settings for the process data transfer and perhaps terminal-specific parameters that may differ from the default settings are also transferred.

2.4.3 Safe-Operational (Safe-Op)

During transition between *Pre-Op* and *Safe-Op* the EtherCAT slave checks whether the sync manager channels for process data communication and, if required, the distributed clocks settings are correct. Before it acknowledges the change of state, the EtherCAT slave copies current input data into the associated DP-RAM areas of the EtherCAT slave controller (ECSC).

In *Safe-Op* state mailbox and process data communication is possible, although the slave keeps its outputs in a safe state, while the input data are updated cyclically.

2.4.4 Operational (Op)

Before the EtherCAT master switches the EtherCAT slave from *Safe-Op* to *Op* it must transfer valid output data.

In the *Op* state the slave copies the output data of the masters to its outputs. Process data and mailbox communication is possible.

2.4.5 Boot

In the *Boot* state the slave firmware can be updated. The *Boot* state can only be reached via the *Init* state.

In the *Boot* state mailbox communication via the *file access over EtherCAT* (FoE) protocol is possible, but no other mailbox communication and no process data communication.

3 OVERVIEW

This section provides an overview of the ETHERCAT Coupler.

Note: For information about configuring the ETHERCAT Coupler, see the Kollmorgen Automation Suite™ IDE software and online help system.

3.1 EtherCAT Coupler (AKT-ECT-000-000)

The ETHERCAT Coupler connects EtherCAT, the real-time Ethernet system, with the modular, extendable electronic terminal blocks. A unit consists of a ETHERCAT Bus Coupler, any number (between 1 and 64) of terminals (255 with Standard Bus extension) and one end terminal.

The ETHERCAT Bus Coupler recognizes the connected Bus Terminals and automatically allocates them into the EtherCAT process image. The ETHERCAT Bus Coupler is connected to the network via the upper Ethernet interface. The lower RJ 45 socket may be used to connect further EtherCAT devices in the same strand.

In the EtherCAT network, the ETHERCAT Coupler can be installed anywhere in the Ethernet signal transfer section (100BASE-TX) - except directly at the switch.

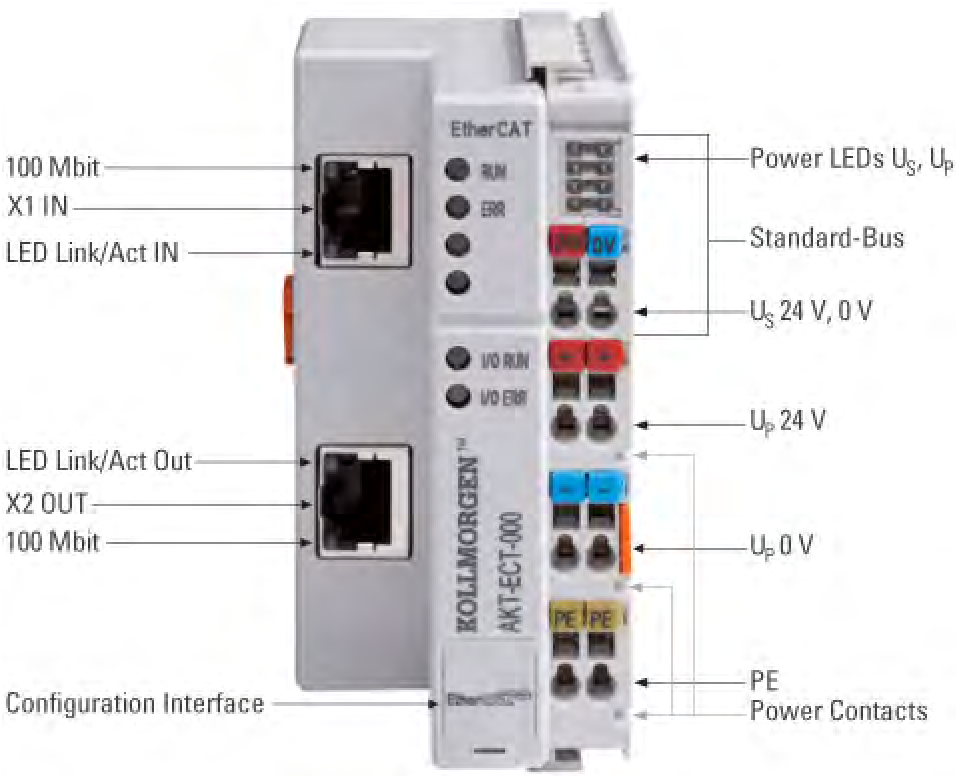


Figure 2.1 EtherCAT Coupler (Front View)

3.1.1 Pin Assignment

The following table provides the pin assignment for the ETHERCAT Coupler.

Terminal Connection		Description
Designation	**Number**	
+24 V U_S	1	Supply for Bus Coupler and Performance Bus circuit
+24 V U_P	2	Supply for power contacts (internally connected with terminal point 6)
0 V U_p	3	Supply for power contacts (internally connected with terminal point 7)
PE	4	PE connection (internally connected with terminal point 8)
0 V U_S	5	Supply for Bus Coupler and Performance Bus circuit
+24 V U_P	6	Supply for power contacts (internally connected with terminal point 2)
0 V U_P	7	Supply for power contacts (internally connected with terminal point 3)
PE	8	PE connection (internally connected with terminal point 4)

3.2 Technical Data

This section provides the technical details for the ETHERCAT Coupler.

Parameters	AKT-ECT-000-000
Task in the EtherCAT-System	Coupling of Standard-Bus Terminals on 100BASE-TX EtherCAT networks
Number of Bus Terminals	64 (255 with Bus extension)
Max. number of bytes fieldbus	1024 byte input and 1024 byte output
Digital peripheral signals	8192 inputs/outputs
Analog peripheral signals	256 inputs/outputs
Protocols	EtherCAT (Direct mode)
Baud rate	100 MBaud
Configuration	EtherCAT (ADS)
Bus connection	2 x RJ45
Power supply	24 V_{DC} (-15%/+20%)
Input current	70 mA + (Standard Bus current)/4, max. 500 mA

Parameters	AKT-ECT-000-000
Standard-Bus supply current (5 V)	Max. 1750 mA
Power contacts	Max. 24 V_{DC}, maximal 10 A
Dielectric strength	500 V_{rms} (power contact/supply voltage/Ethernet)
Weight	~170 g
Operating temperature	0°C ... + 55°C
Storage temperature	-25°C ... + 85°C
Relative humidity	95%, no condensation
Free Fall	up to 1 m in the original packaging
Dimensions (W x H x D)	~49 mm x 100 mm x 70 mm
Mounting	On 35 mm mounting rail (conforms to EN 50022)
Vibration/shock resistance	Conforms to EN 60068-2-6 / EN 60068-2-27, EN 60068-2-29
EMC resistance burst/ESD	Conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP 20
Installation position	Variable
Approvals	CE, UL
Protection class in accordance with IEC 536 (VDE 0106, Part 1)	A protective conductor connection to the mounting rail is necessary!
Protection class according to IEC 529	IP20 (protection against contact with a standard test finger)
Protection against foreign objects	Less than 12 mm in diameter

4 MOUNTING AND WIRING

This section provides mounting and wiring information for the ETHERCAT Coupler.

Note: For information about configuring the ETHERCAT Coupler, see the Kollmorgen Automation Suite™ IDE software and online help system.

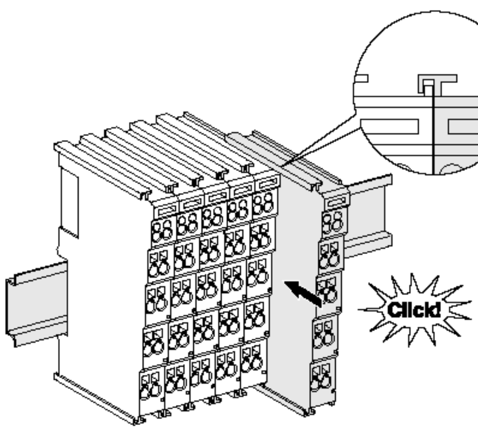
4.1 Installation of Bus Terminals on Mounting Rails

DANGER!! Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the Bus Terminals!

4.1.1 Assembly

The ETHERCAT Bus Coupler and Bus Terminals are attached to commercially available 35 mm mounting rails (DIN rails according to EN 50022) by applying slight pressure:

1. First attach the Fieldbus Coupler to the mounting rail.
2. The Bus Terminals are now attached on the right-hand side of the Fieldbus Coupler. Join the components with tongue and groove and push the terminals against the mounting rail, until the lock clicks onto the mounting rail.



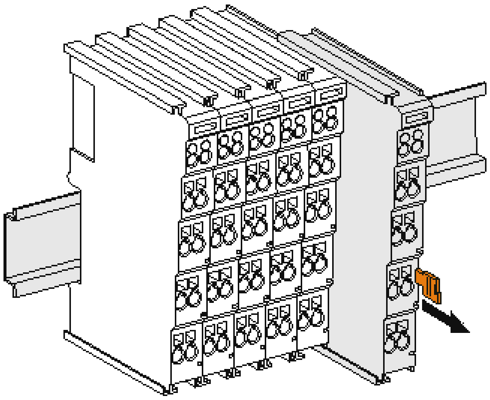
If the Terminals are clipped onto the mounting rail first and then pushed together without tongue and groove, the connection will not be operational! When correctly assembled, no significant gap should be visible between the housings.

Note: During the installation of the Bus Terminals, the locking mechanism of the terminals must not come into conflict with the fixing bolts of the mounting rail.

4.1.2 Disassembly

Each terminal is secured by a lock on the mounting rail, which must be released for disassembly:

1. Carefully pull the orange-colored lug approximately 1 cm out of the disassembled terminal, until it protrudes loosely. The lock with the mounting rail is now released for this terminal, and the terminal can be pulled from the mounting rail without excessive force.



2. Grasp the released terminal with thumb and index finger simultaneous at the upper and lower grooved housing surfaces and pull the terminal away from the mounting rail.

4.1.3 Connections Within a Bus Terminal Block

The electric connections between the ETHERCAT Bus Coupler and the Bus Terminals are automatically realized by joining the components:

- The six spring contacts of the Standard Bus/Performance Bus deal with the transfer of the data and the supply of the Bus Terminal electronics.
- The power contacts deal with the supply for the field electronics and thus represent a supply rail within the bus terminal block. The power contacts are supplied via terminals on the ETHERCAT Bus Coupler (up to 24 V) or for higher voltages via power feed terminals.

Note: During the design of a bus terminal block, the pin assignment of the individual Bus Terminals must be taken account of, since some types (e.g. analog Bus Terminals or digital 4-channel Bus Terminals) do not or not fully loop through the power contacts. Power Feed Terminals interrupt the power contacts and thus represent the start of a new supply rail.

4.1.4 PE Power Contact

The power contact labeled PE can be used as a protective earth. For safety reasons this contact mates first when plugging together, and can ground short-circuit currents of up to 125 A.

WARNING!! Note that, for reasons of electromagnetic compatibility, the PE contacts are capacitatively coupled to the mounting rail. This may lead to incorrect results during insulation testing or to damage on the terminal (e.g. disruptive discharge to the PE line during insulation testing of a consumer with a nominal voltage of 230 V).

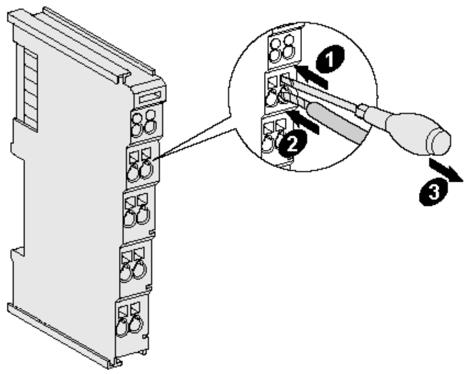
For insulation testing, disconnect the PE supply line at the Bus Coupler or the Power Feed Terminal! In order to decouple further feed points for testing, these Power Feed Terminals can be released and pulled at least 10 mm from the group of terminals.

DANGER!! The PE power contact must not be used for other potentials!

4.1.5 Wiring

Up to eight connections enable the connection of solid or finely stranded cables to the Bus Terminals. The terminals are implemented in spring force technology. Connect the cables as follows:

1. Open a spring-loaded terminal by slightly pushing with a screwdriver or a rod into the square opening above the terminal.



2. The wire can now be inserted into the round terminal opening without any force.
3. The terminal closes automatically when the pressure is released, holding the wire securely and permanently.

Wire Size Width	Wire Stripping Length
0,08 ... 2,5 mm^2	8 mm

Note: Analog sensors and actors should always be connected with shielded, twisted paired wires.

4.2 Power Supply and Potential Groups

This section provides power supply information for the ETHERCAT Coupler.

4.2.1 ETHERCAT Bus Coupler Power Supply

The EHERCAT Bus Coupler requires a 24 V DC supply for their operation. The connection is made by means of the upper spring-loaded terminals labeled 24 V and 0 V. The supply voltage feeds the EHERCAT Bus Coupler electronics and, over the Standard Bus/Performance Bus, the Bus Terminals. The power supply for the EHERCAT Bus Coupler electronics and that of the Standard Bus/Performance Bus are electrically separated from the potential of the field level.

4.2.2 Input for Power Contacts

The bottom six connections with spring-loaded terminals can be used to feed the supply for the peripherals. The spring-loaded terminals are joined in pairs to a power contact. The feed for the power contacts has no connection to the voltage supply for the ETHERCAT Bus Coupler. The design of the feed permits voltages of up to 24 V. The assignment in pairs and the electrical connection between feed terminal contacts allows the connection wires to be looped through to various terminal points. The current drawn from the power contacts must not exceed 10 A for long periods. The current carrying capacity between two spring-loaded terminals is identical to that of the connecting wires.

4.2.3 Power Contacts

On the right hand face of the ETHERCAT Bus Coupler there are three spring contacts for the power contact connections. The spring contacts are hidden in slots so that they can not be accidentally touched. By attaching a Bus Terminal the blade contacts on the left hand side of the Bus Terminal are connected to the spring contacts. The tongue and groove guides on the top and bottom of the ETHERCAT Bus Coupler and of the Bus Terminals guarantees that the power contacts mate securely.

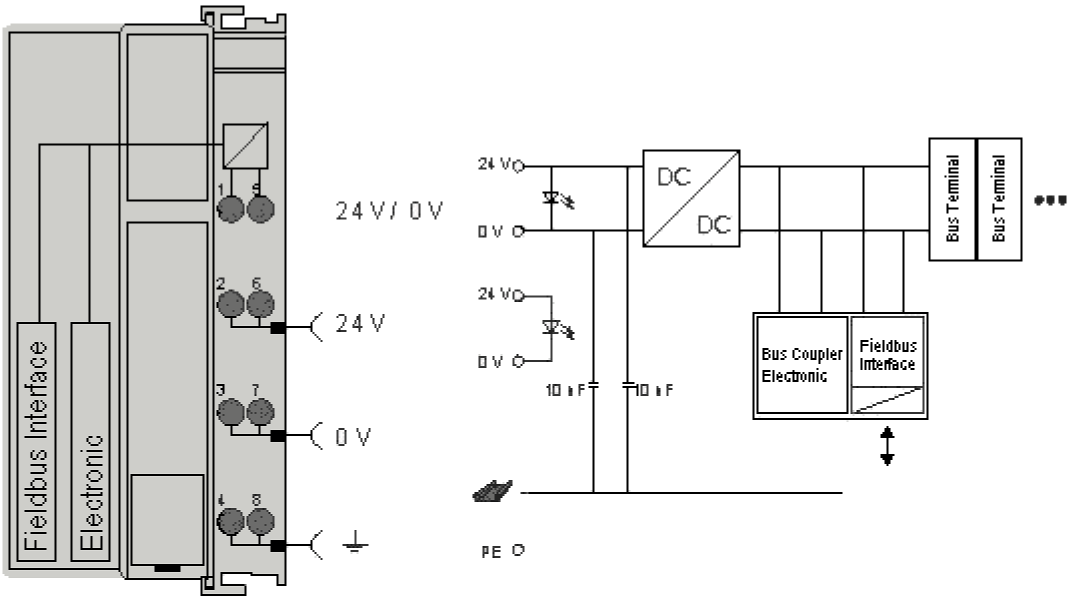
4.2.4 Configuration Interface

The standard ETHERCAT Bus Couplers have an RS232 interface at the bottom of the front face. The miniature connector can be joined to a PC with the aid of a connecting cable and the configuration software. The interface permits the Bus Terminals to be configured, for example adjusting the amplification factors of the analog channels. The interface can also be used to change the assignments of the Bus Terminal data to the process image in the ETHERCAT Bus Coupler. The functionality of the configuration interface can also be reached via the fieldbus using string communication facility.

4.2.5 Electrical Isolation

The ETHERCAT Bus Couplers operate by means of three independent potential groups. The supply voltage feeds the Standard Bus/Performance Bus electronics in the ETHERCAT Bus Coupler and the Standard Bus/Performance Bus itself, which are electrically isolated. The supply voltage is also used to generate the operating voltage for the fieldbus.

Note: All the Bus Terminals are electrically isolated from the Standard Bus/Performance Bus. The Standard Bus/Performance Bus is thus electrically isolated from everything else.



4.3 EtherCat Wiring

The cable length between two EtherCAT devices must not exceed 100 m.

4.3.1 Cables and Connectors

For connecting EtherCAT devices only Ethernet cables that meet at least the requirements of category 5 (CAT5) according to EN 50173 or ISO/IEC 11801 should be used. EtherCAT uses 4 wires for signal transfer.

EtherCAT uses RJ45 connectors. The pin assignment is compatible with the Ethernet standard (ISO/IEC 8802-3).

Pin	Core Coloring	Signal	Description
1	Yellow	TD+	Transmission Data +
2	Orange	TD-	Transmission Data -
3	White	RD+	Receiver Data +
4	Blue	RD-	Receiver Data -

4.3.2 Performance Bus Supply

WARNING!! The same ground potential must be used for the Performance Bus supply of all EtherCAT terminals in a terminal block!

4.4 Ethernet Cable

This section provides the Ethernet cable transmission standards.

4.4.1 10Base5

The transmission medium for 10Base5 consists of a thick coaxial cable ("yellow cable") with a max. transmission speed of 10 MBaud arranged in a line topology with branches (drops) each of which is connected to one network device. Because all the devices are in this case connected to a common transmission medium, it is inevitable that collisions occur often in 10Base5.

4.4.2 10Base2

10Base2 (Cheapernet) is a further development of 10Base5, and has the advantage that the coaxial cable is cheaper and, being more flexible, is easier to lay. It is possible for several devices to be connected to one 10Base2 cable. It is frequent for branches from a 10Base5 backbone to be implemented in 10Base2.

4.4.3 10BaseT

Describes a twisted pair cable for 10 MBaud. The network here is constructed as a star. It is no longer the case that every device is attached to the same medium. This means that a broken cable no longer results in failure of the entire network. The use of switches as star couplers enables collisions to be reduced. Using full-duplex connections they can even be entirely avoided.

4.4.4 100BaseT

Twisted pair cable for 100 MBaud. It is necessary to use a higher cable quality and to employ appropriate hubs or switches in order to achieve the higher data rate.

4.4.5 10BaseF

The 10BaseF standard describes several optical fiber versions.

4.5 Short Descriptions of Cable Types

Twisted pair copper cable for star topologies, where the distance between two devices may not exceed 100 meters.

4.5.1 UTP

Unshielded twisted pair
This type of cable belongs to category 3, and is not recommended for use in an industrial environment.

4.5.2 S/UTP

Screened/unshielded twisted pair (screened with copper braid)
Has a general screen of copper braid to reduce influence of external interference. This cable is recommended for use with Bus Couplers.

4.5.3 FTP

Foiled shielded twisted pair (screened with aluminum foil)
This cable has an outer screen of laminated aluminum and plastic foil.

4.5.4 S/FTP

Screened/foiled-shielded twisted pair (screened with copper braid and aluminum foil)
Has a laminated aluminum screen with a copper braid on top. Such cables can provide up to 70 dB Shielded twisted pair reduction in interference power.

4.5.5 STP

Shielded twisted pair
Describes a cable with an outer screen, without defining the nature of the screen any more closely.

4.5.6 S/STP

Screened/shielded twisted pair (wires are individually screened)
This identification refers to a cable with a screen for each of the two wires as well as an outer shield.

4.5.7 ITP

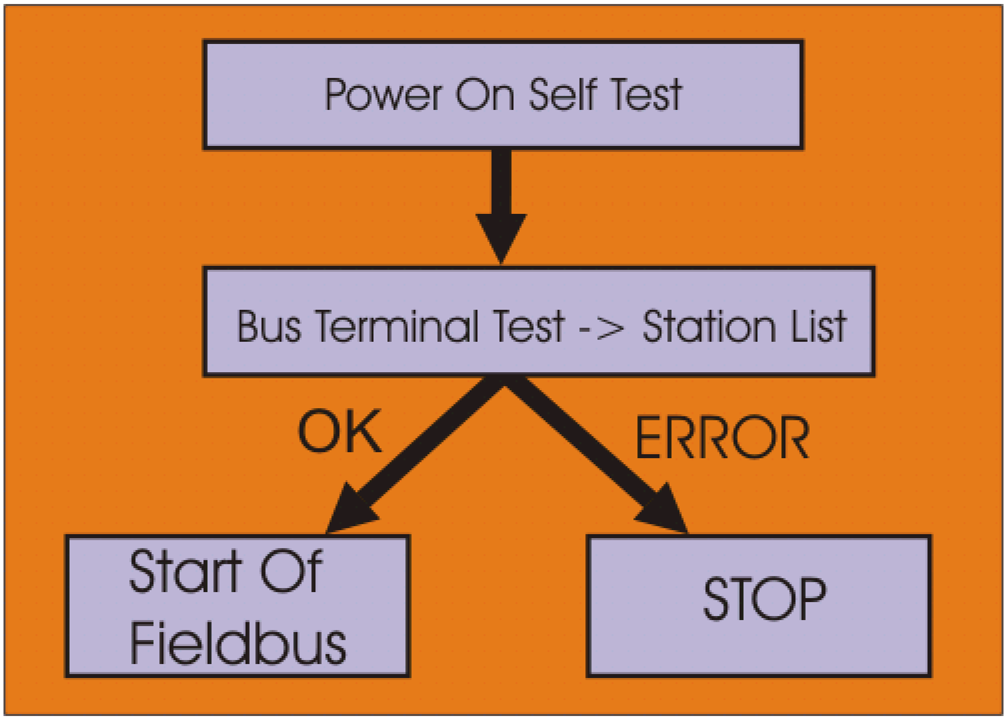
Industrial Twisted-Pair
The structure is similar to that of S/STP, but, in contrast to S/STP, it has only one pair of conductors.

5 PARAMETERIZATION AND COMMISSIONING

This section covers the parameterization and commissioning of the EtherCAT coupler.

5.1 Start-up Behavior of the Bus Coupler

Immediately after being switched on, the Bus Coupler checks, in the course of a self test, all the functions of its components and the communication on the Standard-Bus/E-Bus. The red I/O LED blinks while this is happening After completion of the self-test, the Bus Coupler starts to test the attached Bus Terminals (the Bus Terminal Test), and reads in the configuration. The Bus Terminal configuration is used to generate an internal structure list, which is not accessible from outside. In case of an error, the Bus Coupler enters the STOP state. Once the start-up has completed without error, the Bus Coupler enters the fieldbus start state.



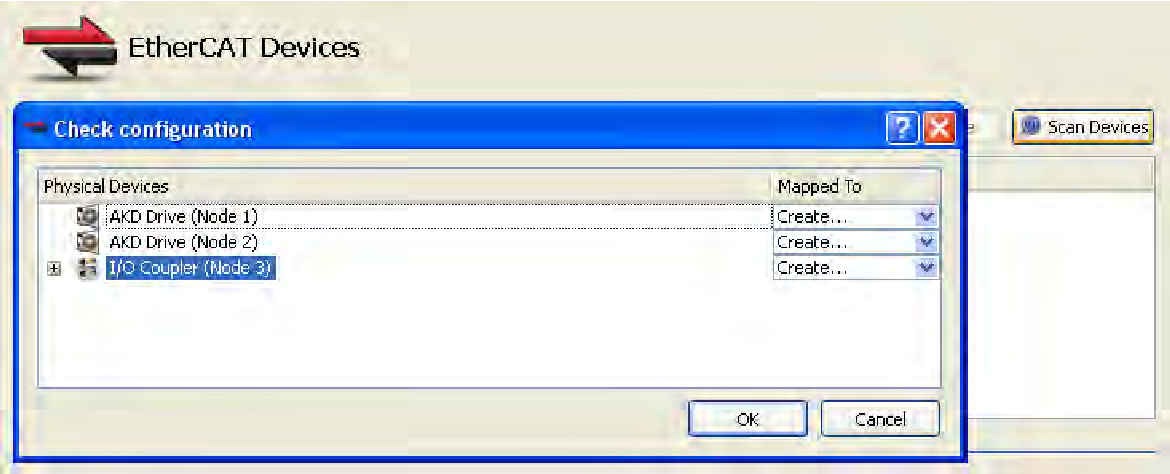
The Bus Coupler can be made to enter the normal operating state by switching it on again once the fault has been rectified.

5.2 Automatic Configuration

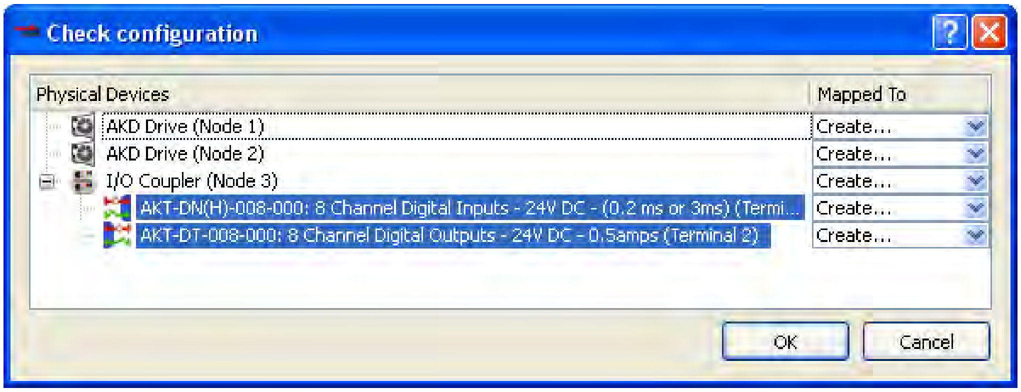
This section describes the basics of automatic configuration within the KAS Integrated Development Environment (IDE).

5.2.1 Scan Devices

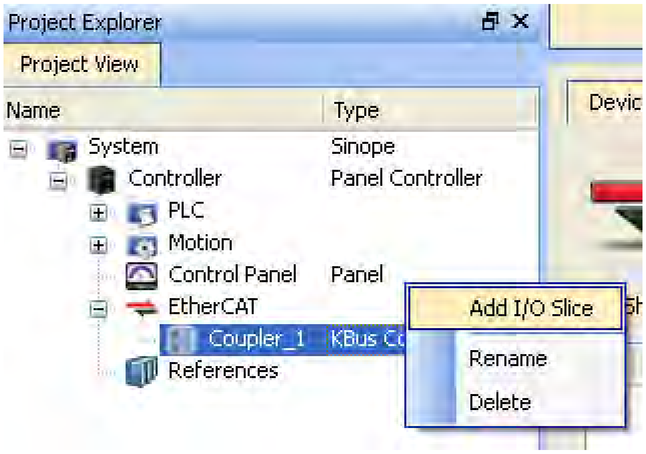
For ease-of-use the KAS IDE Scan Device feature provides automatic integration of I/O devices. This allows you to automatically locate and add I/O couplers to the application project:



Additionally, Scan Device will locate and add I/O slices to the application project:



Also, I/O slices can also be added in the project explorer of the IDE:

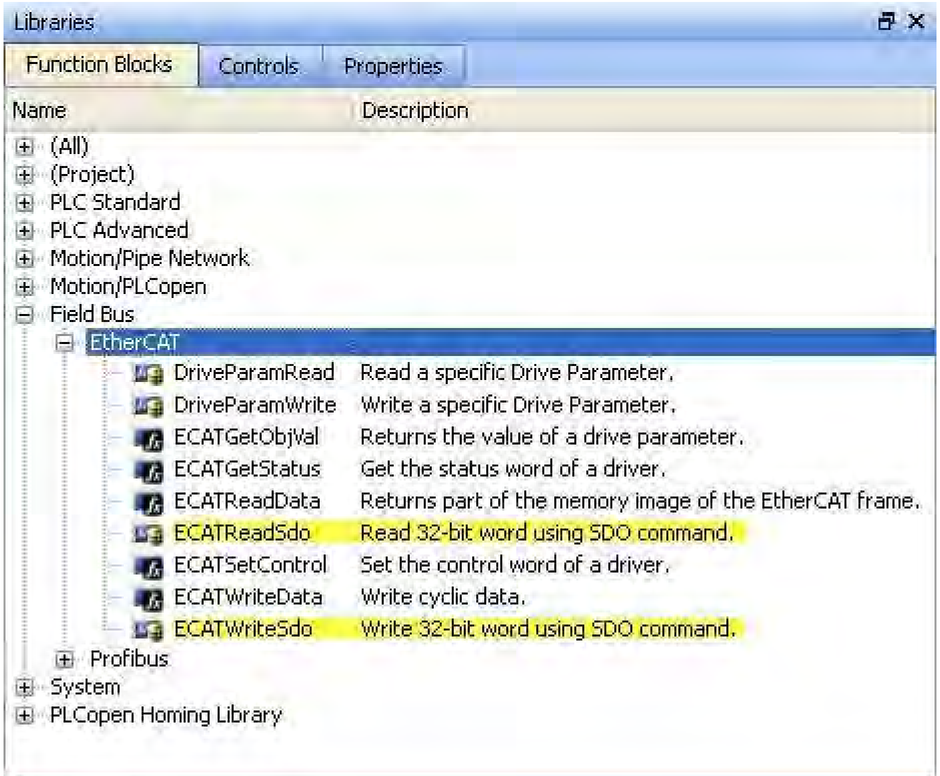


For more detailed information on these procedures refer to the section “EtherCAT Scan Device” in the KAS IDE online help.

5.3 Manual Configuration

Kollmorgen strongly recommends automatic configuration using the KAS IDE over manual configuration. For automatic configuration refer to section 5.2. Manual configuration is for advanced procedures only. The following sections provide information on:

1. Advanced configuration settings that can be made on Registers within this I/O module using EtherCAT Read and Write SDO function blocks in the application project code. These function blocks are located as follows:



2. Manually mapping this I/O block into a KAS project. Manual mapping requires an additional configuration tool. Contact Kollmorgen for more information.

5.3.1 Object Overview

Index		Name	Flags	Default or sample value
1000		Device type	RO	0x0000138A (5002_{dec})
1008		Device name	RO	e.g. BK1120-0000-0000
1009		Hardware version	RO	01
100A		Software version	RO	e.g. 07 (B7)
1018:0	**Subindex**	Identity	RO	>4<
	1018:01	Vendor ID	RO	0x00000002 (2_{dec})

Index		Name	Flags	Default or sample value
	1018:02	Product code	RO	0x04602C22 (73411618$_{dec}$)
	1018:03	Revision number	RO	0x00000000 (0$_{dec}$)
	1018:04	Serial number	RO	0x00000000 (0$_{dec}$)
1600 – 16FE		RxPDO Mapping Area	RW	-
16FF:0	**Subindex**	Control PDO	RW	>1<
	16FF:01	Output Mapping Area	RW	0xF200:01, 16
1701:0	**Subindex**	RxPDO Digital Align	RW	>1<
	1701:01	PDO Align	RW	e.g. 0x0000:00, 14
1A00 – 1AFE		TxPDO Mapping Area	RW	-
1AFF:0	**Subindex**	Status PDO	RW	>1<
	1AFF:01	Input Mapping Area	RW	0xF100:01, 16
1B01:0	**Subindex**	TxPDO Digital Align	RW	>1<
	1B01:01	PDO Align	RW	e.g. 0x0000:00, 6
1C00:0	**Subindex**	Sync Manager Type	RO	>4<
	1C00:01	Subindex 001	RO	0x01 (1$_{dec}$)
	1C00:02	Subindex 002	RO	0x02 (2$_{dec}$)
	1C00:03	Subindex 003	RO	0x03 (3$_{dec}$)
	1C00:04	Subindex 004	RO	0x04 (4$_{dec}$)
1C12:0**	**Subindex**	RxPDO Assign	RO	><
	1C12:01	Subindex 001	RW	0x16FF (5887$_{dec}$)
	1C12:02	Subindex 002	RW	e.g. 0x1603 (5635$_{dec}$)
	1C12:03	Subindex 003	RW	e.g. 0x1600 (5632$_{dec}$)
	…	…	RW	…
	1C12:hh	Subindex ddd	RW	0x1701(5889$_{dec}$)
1C13:0**	**Subindex**	TxPDO Assign	RW	><
	1C13:01	Subindex 001	RW	0x1AFF (6911$_{dec}$)
	1C13:02	Subindex 002	RW	e.g. 0x1A03 (6659$_{dec}$)
	1C13:03	Subindex 003	RW	e.g. 0x1A01 (6657$_{dec}$)
	…	…	…	…
	1C13:hh	Subindex ddd	RW	0x1701(5889$_{dec}$)
4000:0	**Subindex**	Coupler Table 0 [LO]	RW	>128<
	4000:01	Register 0	RW	0x0000 (0$_{dec}$)
	4000:02	Register 1	RW	0x0000 (0$_{dec}$)

Index		Name	Flags	Default or sample value
	…	…	…	…
	4000:80	Register 127	RW	0x0000 (0_{dec})
4001:0	**Subindex**	Coupler Table 0 [HI]	RW	>128<
	4001:01	Register 128	RW	0xFFFF (65535_{dec})
	4001:02	Register 129	RW	0xFFFF (65535_{dec})
	…	…	…	…
	4001:80	Register 255	RW	0xB700 (46848_{dec})
4012:0	**Subindex**	Coupler Table 9 [LO]	RO	>4<
	4002:01	Register 000	RO	0x0460 (1120_{dec})
	4002:02	Register 001	RO	e.g.: 0x8201 (33281_{dec})
	4002:03	Register 003	RO	e.g.: 0x0BBA (3002_{dec})
	4002:04	Register 004	RO	e.g.: 0x0C84 (3204_{dec})
4013:0	**Subindex**	Coupler Table 9 [HI]	RO	>0<
40B4:0	**Subindex**	Coupler Table 90 [LO]	RO	>128<
	40B4:01	Register 0	RO	0x0000 (0_{dec})
	40B4:02	Register 1	RO	0x0000 (0_{dec})
	…	…	…	…
	40B4:80	Register 127	RO	0x0000 (0_{dec})
40B5:0	**Subindex**	Coupler Table 90 [HI]	RO	>128<
	40B5:01	Register 128	RO	0x0000 (0_{dec})
	40B5:02	Register 129	RO	0x0000 (0_{dec})
	…	…	…	…
	40B5:80	Register 255	RO	0x0000 (0_{dec})
40C4:0	**Subindex**	Coupler Table 98 [LO]	RW	>5<
	40C4:01	Min. Standard-Bus Time	RW	e.g. 0x0355 (853_{dec})
	40C4:02	Max. Standard-Bus Time	RW	e.g. 0x059A (1434_{dec})
	40C4:03	Curr. Standard-Bus Time	RW	e.g. 0x036A (874_{dec})
	40C4:04	reserved	RW	0x0000 (0_{dec})
	40C4:05	reserved	RW	0x09F8 (2552_{dec})
40C5:0	**Subindex**	Coupler Table 98 [HI]	RW	>0<
40C8:0	**Subindex**	Coupler Table 100 [LO]	RW	>128<
	40C8:01	Register 0	RW	0x0000 (0_{dec})
	40C8:02	Register 1	RW	0x0000 (0_{dec})

Index		Name	Flags	Default or sample value
	…	…	…	…
	40C8:80	Register 127	RW	0x0000 (0_{dec})
40C9:0	**Subindex**	Coupler Table 100 [HI]	RW	>128<
	40C9:01	Register 128	RW	0x0000 (0_{dec})
	40C9:02	Register 129	RW	0x0000 (0_{dec})
	…	…	…	…
	40C9:80	Register 255	RW	0x0000 (0_{dec})
40FE:0	**Subindex**	Coupler Table 127 [LO]	RO	>128<
	40FE:01	Register 0	RO	0x0000 (0_{dec})
	40FE:02	Register 1	RO	0x0000 (0_{dec})
	…	…	…	…
	40FE:80	Register 127	RO	0x0000 (0_{dec})
40FF:0	**Subindex**	Coupler Table 127 [HI]	RO	>128<
	40FF:01	Register 128	RO	0x0000 (0_{dec})
	40FF:02	Register 129	RO	0x0000 (0_{dec})
	…	…	…	…
	40FF:80	Register 255	RO	0x0000 (0_{dec})
427F		Terminal No	RW	0x0003 (3_{dec})
4280:0	**Subindex**	Terminal Channel 0 [LO]	RW	>64<
	4280:1	Register 0	RW	0x0000 (0_{dec})
	4280:2	Register 1	RW	0x0000 (0_{dec})
	…	…	…	…
	4280:40	Register 63	RW	0x0000 (0_{dec})
4281:0	**Subindex**	Terminal Channel 0 [HI]	RW	>0<
4282:0	**Subindex**	Terminal Channel 1 [LO]	RW	>64<
	4280:1	Register 0	RW	0x0000 (0_{dec})
	4280:2	Register 1	RW	0x0000 (0_{dec})
	…	…	…	…
	4280:40	Register 63	RW	0x0000 (0_{dec})
4283:0	**Subindex**	Terminal Channel 1 [HI]	RW	>0<
4284:0	**Subindex**	Terminal Channel 2 [LO]	RW	>64<
	4280:01	Register 0	RW	0x0000 (0_{dec})
	4280:02	Register 1	RW	0x0000 (0_{dec})

Index		Name	Flags	Default or sample value
	…	…	…	…
	4280:40	Register 63	RW	0x0000 (0_{dec})
4285:0	**Subindex**	Terminal Channel 2 [HI]	RW	>0<
4286:0	**Subindex**	Terminal Channel 3 [LO]	RW	>64<
	4280:01	Register 0	RW	0x0000 (0_{dec})
	4280:02	Register 1	RW	0x0000 (0_{dec})
	…	…	…	…
	4280:40	Register 63	RW	0x0000 (0_{dec})
4287:0	**Subindex**	Terminal Channel 3 [HI]	RW	>0<
6000 – 6FF0		Terminal Input data	RO P	-
7000 – 7FF0		Terminal Output data	RO P	-
9kk0:0 ***	**Subindex**	Slave kk Info data	RO	>10<
	9hh0:01	Position	RO	e.g. 0x0001 (1_{dec})
	9hh0:09	Module PDO Group	RO	e.g. 0x0002 (2_{dec})
	9hh0:0A	Module Indent	RO	e.g. 0x00008201 (33282_{dec})
F000:0	**Subindex**	Module Device Profile	RO	>5<
	F000:01	Module Index Distance	RO	0x0010 (16_{dec})
	F000:02	Maximum Number of Modules	RO	0x00FF (255_{dec})
	F000:03	Standard Entries in Object 0x8yy0	RO	0x00000000 (0_{dec})
	F000:04	Standard Entries in Object 0x9yy0	RO	0x00000000 (0_{dec})
	F000:05	Module PDO Group	RO	0x00 (0_{dec})
F00E:0	**Subindex**	Group Alignment PDO	RO	>3<
	F00E:01	Subindex 001	RO	0x0000 (0_{dec})
	F00E:02	Subindex 002	RO	0x0100 (256_{dec})
	F00E:03	Subindex 003	RO	0x0101 (257_{dec})
F00F:0	**Subindex**	Module Group Mapping Alignment	RO	>3<
	F00F:01	Subindex 001	RO	0x0000 (0_{dec})
	F00F:02	Subindex 002	RO	0x0002 (2_{dec})
	F00F:03	Subindex 003	RO	0x0002 (2_{dec})
F010:0 **	**Subindex**	Module List	RO	><
	F010:01	Subindex 001	RO	e.g. 0x8202 (33282_{dec})

Index		Name	Flags	Default or sample value
	…	…	…	…
	F010:hh	Subindex ddd	RO	e.g. 0x09D9 (2521$_{dec}$)
F030:0 **	**Subindex**	Configured Module List	RW	><
	F030:01	Subindex 001	RW	e.g. 0x00000000 (0$_{dec}$)
	…	…	…	…
	F030:hh	Subindex ddd	RW	e.g. 0x00000000 (0$_{dec}$)
F040:0 **	**Subindex**	Detected Address List	RO	><
	F040:01	Subindex 001	RO	0x0001 (1$_{dec}$)
	…	…	…	…
	F040:hh	Subindex ddd	RO	0x00FF (0$_{dec}$)
F050:0 **	**Subindex**	Detected Module List	RO	><
	F050:01	Subindex 001	RO	e.g. 0x00008202 (33282$_{dec}$)
	…	…	…	…
	F050:hh	Subindex ddd	RO	e.g. 0x000009D9 (2521$_{dec}$)
F100:0	**Subindex**	Status PDO	RO P	>1<
	F0100:01	CouplerState	RO P	0x0000 (0$_{dec}$)
F200:0	**Subindex**	Status PDO	RW P	>1<
	F200:01	CouplerCtrl	RW P	0x0000 (0$_{dec}$)

Key:

**Dynamic generated object:

Value range: (hh= 01..FF)$_{hex}$, (ddd= 001..255)$_{dec}$

***Dynamic generated object:

Value range: (hh= 00..FF)$_{hex}$

Flags:

RO (Read Only)

RO P (Read Only Process Data)

RW (Read/Write)

RW P (Read/Write Process Data)

5.3.2 Standard Objects

Index 1000: Device type

Device type (read only) of the EtherCAT slave.

Index 1008 Device name

Device name (read only, value: e. g. AKT-ECT-000-000) of the EtherCAT slave.

Index 1009 Hardware version

Hardware version (read only) of the EtherCAT slave.

Index 100A Software version

Version (read only, value: e. g. 07 (B7)) of the firmware (processing of the I/O signals) of the EtherCAT slave.

Index 1018 Identity object

Index	Name	Meaning	Flags	Default
1018:0	Identify object	Length of this object	RO	4
1018:01	Vendor id	Vendor ID of the EtherCAT slave	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	RO	0x04602C22 (73411618_{dec})
1018:03	Revision number	Revision number of the EtherCAT slave	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave	RO	0x00000000 (0_{dec})

Index 16kk RxPDO Mapping Terminal

Value range:

$(kk = 00..FE)_{hex}$,

$[(hh = 01..FF)_{hex}, (ddd = 001..255)]_{dec}$

Index	Name	Meaning	Flags	Default
16kk:0	RxPDO Mapping Terminal	Length of this object	RW	-
16kk:01		First Output variable of this terminal (Subindex 001)	RW	e.g. 0x7kk0:01, 1
…	…	…	…	…
16kk:hh		Last Output variable of this terminal (Subindex ddd)	RW	e.g. 0x7kk0:hh, 1

Index 16FF Control PDO

Index	Name	Meaning	Flags	Default
16FF:0	Control PDO	Length of this object	RW	-
16FF:01	Output Mapping Area	Control word	RW	0x0000:00, 00

Index 1701 RxPDO Digital Align

Index	Name	Meaning	Flags	Default
1701:0	RxPDO Digital Align	Length of this object	RW	-
1701:01	PDO Align	Word Alignment Dummy for digital RxPDOs	RW	0x0000:00, 00

Index 1Akk TxPDO Mapping Terminal

Value range:

$(kk = 00..FE)_{hex}$,

$[(hh = 01..FF)_{hex}$,$(ddd = 001..255)_{dec}]$

Index	Name	Meaning	Flags	Default
1Akk:0	TxPDO Mapping Terminal	Length of this object	RW	-
1Akk:01	Input Mapping Area 001	First input variable of this terminal (Subindex 001)	RW	e.g. 0x6kk0:01, 1
…	…	…	…	…
1Akk:hh	Input Mapping Area ddd	Last input variable of this terminal (Subindex 001)	RW	e.g. 0x6kk0:hh, 1

Index 1AFF Status PDO

Index	Name	Meaning	Flags	Default
16FF:0	Status PDO	Length of this object	RW	-
16FF:01	Input Mapping Area	Status word	RW	0xF100:01, 16

Index 1B01 TxPDO Digital Align

Index	Name	Meaning	Flags	Default
1B01:0	TxPDO Digital Align	Length of this object	RW	-
1B01:01	PDO Align	Word Alignment Dummy for digital TxPDOs	RW	0x0000:00, 00

Index 1C12 RxPDO Assign

Value range:

[(hh = 01..FF)$_{hex}$,(ddd = 001..255)$_{dec}$]

Index	Name	Meaning	Flags	Default
1C12:0	RxPDO Assign	Length of this object	RW	-
1C12:01	Subindex 001	Assignment of Control PDO	RW	0x16FF(5887$_{dec}$)
1C12:02	Subindex 002	Assignment 1. RxPDO analog terminals	RW	e.g. 0x1603 (5635$_{dec}$)
…	…	…	…	…
1C12:nn	Subindex nnn	Assignment n. RxPDO analog terminals	RW	e.g. 0x1605 (5637$_{dec}$)
1C12:(nn+1)	Subindex (nnn+1)	Assignment (n+1). RxPDO digital terminals	RW	e.g. 0x1600 (5632$_{dec}$)
…	…	…	…	…
1C12:hh	Subindex ddd	Assignment Word Alignment Dummy for Digital RxPDOs	RW	0x1701(5889$_{dec}$)

Index 1C13 TxPDO Assign

Value range:

[(hh = 01..FF)$_{hex}$,(ddd = 001..255)$_{dec}$]

Index	Name	Meaning	Flags	Default
1C13:0	TxPDO Assign	Length of this object	RW	-
1C13:01	Subindex 001	Assignment of Control PDO	RW	0x1AFF(6911$_{dec}$)
1C13:02	Subindex 002	Assignment 1. TxPDO analog terminals	RW	e.g. 0x1A03 (6659$_{dec}$)
…	…	…	…	…
1C13:nn	Subindex nnn	Assignment n. TxPDO analog terminals	RW	e.g. 0x1A05 (6661$_{dec}$)
1C13:(nn+1)	Subindex (nnn+1)	Assignment (n+1). TxPDO digital terminals	RW	e.g. 0x1A01 (6657$_{dec}$)
…	…	…	…	…
1C13:hh	Subindex ddd	Assignment Word Alignment Dummy for Digital TxPDOs	RW	0x1B01(6913$_{dec}$)

Index 1C00 SM type

Index	Name	Meaning	Flags	Default
1C00:0	SM type	Length of this object	RO	4
1C00:01	Subindex 001	Sync Manager, parameter 1	RO	0x01 (1$_{dec}$)
1C00:02	Subindex 002	Sync Manager, parameter 2	RO	0x02 (2$_{dec}$)
1C00:03	Subindex 003	Sync Manager, parameter 3	RO	0x03 (3$_{dec}$)
1C00:04	Subindex 004	Sync Manager, parameter 4	RO	0x04 (4$_{dec}$)

Index 4000 Coupler Table 0 [LO]: General configuration of Bus Coupler

Index	Name	Meaning	Flags	Default
4000:0	Coupler Table 0 [LO]	Length of this object	RW	128
4000:01	Subindex 001	Register 0	RW	
4000:02	Subindex 002	Register 1	RW	
…	…	…	…	…
4000:80	Subindex 128	Register 127	RW	

5.3.3 Device Specific Objects

Index 4001 Coupler Table 0 [HI]: General configuration of Bus Coupler (continuance)

Index	Name	Meaning	Flags	Default
4000:0	Coupler Table 0 [HI]	Length of this object	RW	128
4000:01	Subindex 001	Register 0	RW	
4000:02	Subindex 002	Register 1	RW	
…	…	…	…	…
4000:80	Subindex 128	Register 127	RW	

Index 4012 Coupler Table 9 [LO]: Structure of the terminal block

Index	Name	Meaning	Flags	Default
4012:0	Coupler Table 0 [HI]	Length of this object	RO	Max. 128
4012:01	Subindex 001	Register 0: identification of the Bus Coupler	RO	0x0460 (1120_{dec}[1])
4012:02	Subindex 002	Register 1: identification of the 1st terminal	RO	e.g.: 0x8201 (33281_{dec}[2])
4012:03	Subindex 003	Register 2: identification of the 2nd terminal	RO	e.g.: 0x0BBA (3002_{dec}[1])
4012:04	Subindex 004	Register 3: identification of the 3rd terminal	RO	e.g.: 0x0C84 (3204_{dec}[1])
…	…	…	…	…
40012:80	Subindex 128	Register 127: identification of the 127th terminal	RO	

1. For Bus Couplers and intelligent (e.g. analog) terminals the terminal identification is shown as plain text;

2. For non-intelligent (digital) terminals, the following coding is essential :

Bit	Value	Meaning
15	0_{bin}	Intelligent (e.g. analog) terminal
	1_{bin}	Non-intelligent (digital) terminal
14…8	$xxxxxx_{bin}$	Size of terminal in bit (range: 0 to 63 bit)
7	1_{bin}	Shift register extension to control panels
6…3	0_{bin}	reserved
2	1_{bin}	**Not supported**
1	1_{bin}	Output terminal
0	1_{bin}	Input terminal

Index 4013 Coupler Table 9 [HI]: Structure of the terminal block (continuance if more than 127 terminals)

Index	Name	Meaning	Flags	Default
4013:0	Coupler Table 9 [HI]	Length of this object	RO	Max. 128 $_{dec}$
4013:01	Subindex 001	Register 128: identification of the 128th terminal	RO	
4013:02	Subindex 002	Register 129: identification of the 129th terminal	RO	
…	…	…	…	
4013:04	Subindex 128	Register 255: identification of the 255th terminal	RO	

Index 40B4 Coupler Table 90 [LO]: Internal diagnostic information of the Bus Coupler

Index	Name	Meaning	Flags	Default
40B4:0	Coupler Table 90 [LO]	Length of this object	RO	128 $_{dec}$
40B4:01	Subindex 001	Register 0: reserved	RO	
40B4:02	Subindex 002	Register 1: reserved	RO	
…	…	…	…	
40B4:4	Subindex 128	Register 127: reserved	RO	

Index 40B5 Coupler Table 90 [HI]: Internal diagnostic information of the Bus Coupler

Index	Name	Meaning	Flags	Default
40B5:0	Coupler Table 90 [HI]	Length of this object	RO	128 dec
40B5:01	Subindex 001	Register 128: reserved	RO	
40B5:02	Subindex 002	Register 129: reserved	RO	
…	…	…	…	
40B5:4	Subindex 128	Register 255: reserved	RO	

Index 40C4 Coupler Table 98 [LO]: Pd Update time (µs)

Index	Name	Meaning	Flags	Default
40C4:0	Coupler Table 98 Pd Update time (µs)	Length of this object	RW	5
40C4:01	Min time	Min Standard-Bus update time	RW	0x0355 (853dec)
40C4:02	Max time	Max Standard-Bus update time	RW	0x059A (1434dec)
40C4:03	Curr time	Current Standard-Bus update time	RW	0x036A (874dec)
40C4:04	-	reserved	RW	0x0000 (0dec)
40C4:50	-	reserved	RW	0x09F8 (2552dec)

Index 40C5 Coupler Table 98 [HI]: Pd Update time (µs) (continuance)

Index	Name	Meaning	Flags	Default
40C5:0	Coupler Table 98 Pd Update time (µs)	Length of this object	RW	0

Index 40C8 Coupler Table 100 [LO]: Reserved for internal fieldbus specific adjustments

Index	Name	Meaning	Flags	Default
40C8:0	Coupler Table 100 [LO]	Length of this object	RO	128 dec
40C8:01	Subindex 001	Register 0: reserved	RO	
40C8:02	Subindex 002	Register 1: reserved	RO	
…	…	…	…	…
40C8:04	Subindex 128	Register 127: reserved	RO	

Index 40C9 Coupler Table 100 [HI]: Reserved for internal fieldbus specific adjustments (continuance)

Index	Name	Meaning	Flags	Default
40C9:0	Coupler Table 100 [HI]	Length of this object	RO	128 $_{dec}$
40C9:01	Subindex 001	Register 128: reserved	RO	
40C9:02	Subindex 002	Register 129: reserved	RO	
…	…	…	…	…
40C9:04	Subindex 128	Register 255: reserved	RO	

Index 40FE Coupler Table 127 [LO]: Reserved for internal adjustments

Index	Name	Meaning	Flags	Default
40FE:0	Coupler Table 100 [LO]	Length of this object	RO	128 $_{dec}$
40FE:01	Subindex 001	Register 0: reserved	RO	
40FE:02	Subindex 002	Register 1: reserved	RO	
…	…	…	…	…
40FE:04	Subindex 128	Register 127: reserved	RO	

Index 40FF Coupler Table 127 [HI]: Reserved for internal adjustments (continuance)

Index	Name	Meaning	Flags	Default
40FF:0	Coupler Table 100 [HI]	Length of this object	RO	128 $_{dec}$
40FF:01	Subindex 001	Register 128: reserved	RO	
40FF:02	Subindex 002	Register 129: reserved	RO	
…	…	…	…	…
40FF:04	Subindex 128	Register 255: reserved	RO	

Index 427F Terminal No.

Index	Name	Meaning	Flags	Default
427F:0	Terminal No	Number of the terminal in the terminal block, which can be parameterized with the 4 following tables (object 4280 - 4287) Example: the Bus Coupler denomination in the terminal block is No. "0" . The first terminal connected to the Bus Coupler has the No. "1", etc. Only intelligent terminals can be parameterized!	RO	3

Index 4280 Terminal Table 0 [LO]: Parameterization of the terminal specified in object 427F

Index	Name	Meaning	Flags	Default
4280:0	Terminal Table 0	Length of this object	RW	64
4280:01	Subindex 001	Register 0	RW	0x0000 (0_{dec})
4280:02	Subindex 002	Register 1	RW	0x0000 (0_{dec})
…	…	…	…	…
4280:40	Subindex 064	Register 63	RW	0x0000 (0_{dec})

Index 4281 Terminal Table 0 [HI]: Parameterization of the terminal specified in object 427F (continuance)

Index	Name	Meaning	Flags	Default
4081:0	Terminal Table 0	Length of this object	RW	0

Index 4282 Terminal Table 1 [LO]: Parameterization of the terminal specified in object 427F

Index	Name	Meaning	Flags	Default
4282:0	Terminal Table 1	Length of this object	RW	64
4282:01	Subindex 001	Register 0	RW	0x0000 (0_{dec})
4282:02	Subindex 002	Register 1	RW	0x0000 (0_{dec})
…	…	…	…	…
4282:40	Subindex 064	Register 63	RW	0x0000 (0_{dec})

Index 4283 Terminal Table 1 [HI]: Parameterization of the terminal specified in object 427F (continuance)

Index	Name	Meaning	Flags	Default
4083:0	Terminal Table 1	Length of this object	RW	0

Index 4284 Terminal Table 2 [LO]: Parameterization of the terminal specified in object 427F

Index	Name	Meaning	Flags	Default
4284:0	Terminal Table 2	Length of this object	RW	64
4284:01	Subindex 001	Register 0	RW	0x0000 (0_{dec})
4284:02	Subindex 002	Register 1	RW	0x0000 (0_{dec})
…	…	…	…	…
4284:40	Subindex 064	Register 63	RW	0x0000 (0_{dec})

Index 4285 Terminal Table 2 [HI]: Parameterization of the terminal specified in object 427F (continuance)

Index	Name	Meaning	Flags	Default
4085:0	Terminal Table 2	Length of this object	RW	0

Index 4286 Terminal Table 3 [LO]: Parameterization of the terminal specified in object 427F

Index	Name	Meaning	Flags	Default
4286:0	Terminal Table 3	Length of this object	RW	64
4286:01	Subindex 001	Register 0	RW	0x0000 (0_{dec})
4286:02	Subindex 002	Register 1	RW	0x0000 (0_{dec})
…	…	…	…	…
4286:40	Subindex 064	Register 63	RW	0x0000 (0_{dec})

Index 4287 Terminal Table 3 [HI]: Parameterization of the terminal specified in object 427F (continuance)

Index	Name	Meaning	Flags	Default
4087:0	Terminal Table 3	Length of this object	RW	0

Index 6kk0 Terminal Input Data

The value range of the variables "kk", "hh", und "ddd" of the input- and output indices are illustrated below.

The dynamically generated input and output objects depends on the order of the terminals in the bus terminal block.

Sample configuration of bus terminal block

AKT-ECT-000-000 – AKT-DN-008-000

Mapping of the assigned PDOs (from Index 6000):

Index	Name	Meaning		
6000:0	AKT-DN-008-000	AKT-DN-008-000, max. Subindex (8)	Position 1 after coupler AKT-DN-008-000	Terminal input data
6000:01	Channel 1	1. Channel, AKT-DN-008-000		
6000:02	Channel 2	2. Channel, AKT-DN-008-000		
6000:03	Channel 3	3. Channel, AKT-DN-008-000		
6000:04	Channel 4	4. Channel, AKT-DN-008-000		
6000:05	Channel 5	5. Channel, AKT-DN-008-000		
6000:06	Channel 6	6. Channel, AKT-DN-008-000		
6000:07	Channel 7	7. Channel, AKT-DN-008-000		
6000:08	Channel 8	8. Channel, AKT-DN-008-000		
6010:0 - 7030:06		**Not supported**		

Digital Terminals:

Value range:

$(kk = 00..FF)_{hex}$,

$[(hh = 01..FF)_{hex}$,$(ddd = 001..255)_{dec}]$

Index	Name	Meaning	Flags	Default
6kk0:0	Terminal Input Data	[Terminal description]	RO P	-
6kk0:01	Channel 001	Channel 001	RO P	-
…	…			…
6kk0:hh	Channel ddd	Channel ddd	RO P	-

Non Digital Terminals (for hh ≥ 03; ddd ≥ 003):

Value range:

$(kk = 00..FF)_{hex}$,

$[(hh = 01..FF)_{hex}, (ddd = 001..255)_{dec}]$ values for hh resp. ddd must be divisible by 3

Index	Name	Meaning	Flags	Default
6kk0:0	Terminal Input Data	[Terminal description]	RO P	-
6kk0:01	Channel 001 Status	Channel 001 Status	RO P	-
6kk0:02	Channel 001 Word-Alignment	Channel 001 Word-Alignment	RO P	-
6kk0:03	Channel 001 Data	Channel 001 Data	RO P	-
…	…	…	…	…
6kk0:hh-2	Channel ddd/3	Channel ddd/3: Status	RO P	-
6kk0:hh-1	Channel ddd/3	Channel ddd/3: Word-Alignment	RO P	-
6kk0:hh	Channel ddd/3	Channel ddd/3: Data	RO P	-

Index 7kk0 Terminal Output Data

Digital Terminals:

Value range:

$(kk = 00..FF)_{hex}$,

$[(hh = 01..FF)_{hex} ,(ddd = 001..255)_{dec}]$

Index	Name	Meaning	Flags	Default
7kk0:0	Terminal Input Data	[Terminal description]	RO P	-
7kk0:01	Channel 001	Channel 001	RO P	-
…	…			…
7kk0:hh	Channel ddd	Channel ddd	RO P	-

Non Digital Terminals (for hh ≥ 03; ddd ≥ 003):

Value range:

(kk = 00..FF)$_{hex}$,

[(hh = 01..FF)$_{hex}$, (ddd = 001..255)$_{dec}$] values for hh resp. ddd must be divisible by 3

Index	Name	Meaning	Flags	Default
7kk0:0	Terminal Input Data	[Terminal description]	RO P	-
7kk0:01	Channel 001 Status	Channel 001 Status	RO P	-
7kk0:02	Channel 001 Word-Alignment	Channel 001 Word-Alignment	RO P	-
7kk0:03	Channel 001 Data	Channel 001 Data	RO P	-
…	…	…	…	…
7kk0:hh-2	Channel ddd/3	Channel ddd/3: Status	RO P	-
7kk0:hh-1	Channel ddd/3	Channel ddd/3: Word-Alignment	RO P	-
7kk0:hh	Channel ddd/3	Channel ddd/3: Data	RO P	-

Index 9kk0 Slave Info Data

Value range:

(kk = 00..FF)$_{hex}$

Index	Name	Meaning	Flags	Default
9kk0:0	Slave Info Data	[Terminal description]	RO	-
9kk0:01	Position	Module position in the terminal block	RO	-
9kk0:09	Module PDO Group	Module PDO Group • Non Digital Terminals: Group 1 • Digital Terminals: Group 2	RO	-
9kk0:0A	Module Indent	Terminal Description	RO	-

Index F000 Modular Device Profile

Index	Name	Meaning	Flags	Default
F000:0	Modular Device Profile	Length of this object	RO	5
F000:01	Module Index Distance	Module Index Distance	RO	0x0010 (16_{dec})
F000:02	Maximum Number of Modules	Maximum Number of Modules	RO	0x00FF (255_{dec})
F000:03	Standard Entries in Object 0x8yy0	Available entries in objects 0x8kk0	RO	0x0000000 (0_{dec})
F000:04	Standard Entries in Object 0x9yy0	Available entries in Object 0x9kk0	RO	0x0000000 (0_{dec})
F000:04	Module PDO Group	Module PDO Group of the Device = 0, because the Control and Status Data will assigned before all terminals in the process data	RO	0x00 (0_{dec})

Index F00E Group Alignment PDO

Index	Name	Meaning	Flags	Default
F00E:0	Group Alignment PDO	Length of this object (3 Module PDO Groups)	RO	3
F00E:01	Subindex 001	PDO Group 0: No Align is needed after Module, Status or Control Data of the device	RO	0x0000 (0_{dec})
F00E:02	Subindex 002	PDO Group 1: PDO 257 is used for the align after Module PDO Group 1 (non-digital terminals)	RO	0x0100 (256_{dec})
F00E:03	Subindex 003	PDO Group 2: PDO 258 is used for the align after Module PDO Group 2 (digital terminals)	RO	0x0101 (257_{dec})

Index F00F Module Group Mapping Alignment

Index	Name	Meaning	Flags	Default
F00F:0	Group Alignment PDO	Length of this object	RO	3
F00F:01	Subindex 001	Group 0: No Align is needed after Module PDO Group 0 (Status or Control Data of the device) The size of the Status Data and Control Data is fixed and already aligned	RO	0x0000 (0_{dec})
F00F:02	Subindex 002	Group 1: A Word Align is needed after Module PDO Group 1 (Non Digital Terminals) The process data of the digital terminals is always located on an even byte address	RO	0x0002 (2_{dec})
F00F:03	Subindex 003	Group 2: A Word Align is needed after Module PDO Group 2 (digital terminals) The process data size is always even	RO	0x0002 (2_{dec})

Index F010 Module List

Value range:

$(hh = 01..FF)_{hex}$,$(ddd = 001..255)_{dec}$

Index	Name	Meaning	Flags	Default
F010:0	Module List	Length of this object	RO	-
F010:01	Subindex 001	Terminal description Module 001	RO	-
…	…	…	…	…
F010:hh	Subindex ddd	Terminal description Module ddd	RO	-

Index F030 Configured Module List

Value range:

(hh = 01..FF)$_{hex}$,(ddd = 001..255)$_{dec}$

Index	Name	Meaning	Flags	Default
F030:0	Module List	Length of this object	RW	-
F030:01	Subindex 001	ModuleIdent of the module configured on position 1 (same value as 0x8kk0:0A)	RW	-
…	…	…	…	…
F030:hh	Subindex ddd	ModuleIdent of the module configured on last position (same value as 0x8kk0:0A)	RW	-

Index F040 Detected Address List

Value range:

(hh = 01..FF)$_{hex}$,(ddd = 001..255)$_{dec}$

Index	Name	Meaning	Flags	Default
F040:0	Module List	Length of this object	RO	-
F040:01	Subindex 001	Address of the module detected on position 1 (same value as 0x9kk0:01)	RO	-
…	…	…	…	…
F040:hh	Subindex ddd	Address of the module detected on last position (same value as 0x9kk0:01)	RO	-

Index F050 Detected Module List

Value range:

(hh = 01..FF)$_{hex}$,(ddd = 001..255)$_{dec}$

Index	Name	Meaning	Flags	Default
F050:0	Module List	Length of this object	RO	-
F050:01	Subindex 001	ModuleIdent of the module detected on position 1 (same value as 9kk0:0A)	RO	-
…	…	…	…	…
F050:hh	Subindex ddd	ModuleIdent of the module detected on last position (same value as 9kk0:0A)	RO	-

Index F100 Status PDO

Index	Name	Meaning	Flags	Default
F100:0	Status PDO	Length of this object	RO	1
F100:01	Coupler/State	Coupler Status Word	RO P	0x0000 (0_{dec})

Index F200 Control PDO

Index	Name	Meaning	Flags	Default
F200:0	Control PDO	Length of this object	RO	1
F200:01	Coupler/Ctrl	Coupler Control Word	RO P	0x0000 (0_{dec})

6 ERROR HANDLING AND DIAGNOSIS

This section provides information about the diagnostic LEDs for the ETHERCAT Coupler.

6.1 Diagnostic LEDs

After switching on, the ETHERCAT Bus Coupler immediately checks the connected configuration. Error-free start-up is indicated when the red I/O ERR LED goes out. If the I/O ERR LED blinks, an error in the area of the terminals is indicated. The error code can be determined from the frequency and number of blinks. See below for more information.

The ETHERCAT Bus Coupler has respectively a green and yellow LED at the RJ45 plug sockets, which indicate the state of the fieldbus (Figure 4.1). The RUN and ERROR LEDs (upper middle) indicate the state of the EtherCAT State Machine.

On the upper right hand side of the Bus Couplers are two more green LEDs that indicate the supply voltage. The left hand LED indicates the presence of the 24 V supply for the Bus Coupler. The right hand LED indicates the presence of the supply to the power contacts.

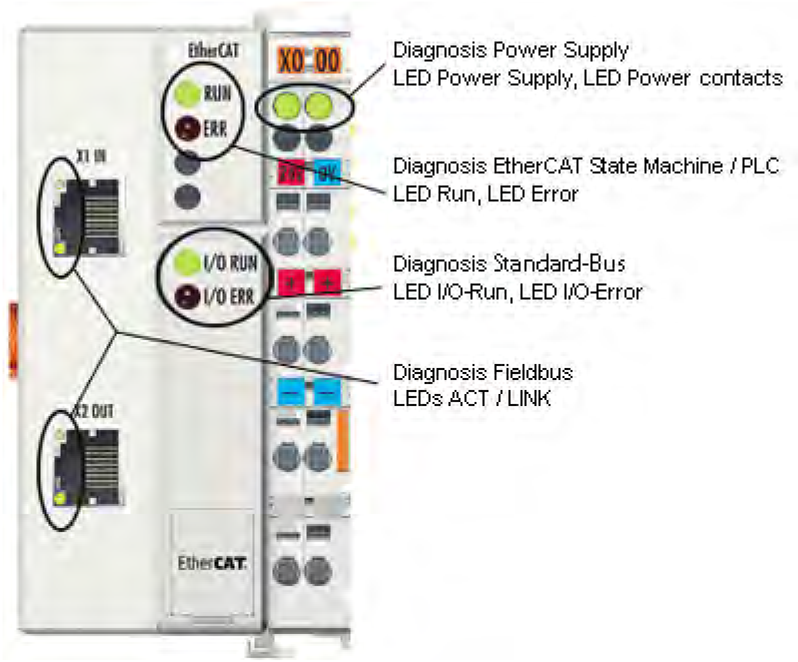


Figure 5.1: ETHERCAT Coupler Diagnostic LEDs

6.2 LED Power Supply Diagnosis

LED		Display	Description
Power Supply	Green	Off	No operating voltage connected
		On	24 V_{DC} operating voltage connected
Power Contacts	Green	Off	No 24V_{DC} power connected to the power contacts
		On	24V_{DC} power connected to the power contacts

6.3 LED Off Power Supply Diagnosis

LEDs	Description
Left green LED off	Bus coupler has no power
Right green LED off	No 24V_{DC} power connected to the power contacts

6.4 LEDs for EtherCAT State Machine/PLC Diagnosis

LEDs		Display	Status	Description
RUN	Green	Off	Init	State of the EtherCAT State Machine: INIT = *Initialization*
		Blinking	Pre-Operational	State of the EtherCAT State Machine: *PREOP = Pre-Operational*
		Single flash	Safe-Operational	State of the EtherCAT State Machine: *SAFEOP = Safe-Operational*
		On	Operational	State of the EtherCAT State Machine: *OP = Operational*
		Flashes	Bootstrap	State of the EtherCAT State Machine: *BOOT = Bootstrap* (Update of the coupler firmware)
ERROR	Red	Off	-	No errors
		Blinking	Err-Operational No Communication	PLC error / Lost frames

6.5 LEDs for Fieldbus Diagnosis

LEDs		Display	Status	Description
LINK (X1 IN)	Yellow	Off	-	No connection with the previous EtherCAT client
		On	Linked	Previous EtherCAT-client connected
ACT (X1 IN)	Green	Blinking	Active	Communication with the previous EtherCAT client
		Off	-	No connection with the previous EtherCAT client
		On	-	No communication with the previous EtherCAT client
LINK (X2 OUT)	Yellow	Off	Linked	Next EtherCAT client connected
		On	Active	Next EtherCAT client connected
ACT(X2 OUT)	Green	Blinking	Active	Communication with the next EtherCAT client
		Off	-	No connection with the next EtherCAT client
		On	-	No communication with next previous EtherCAT client

6.6 LEDs for Standard-Bus Diagnosis

LEDs		Display	Status	Description
I/O-RUN	Green	Off	-	Standard-Bus inactive
		On	-	Standard-Bus active

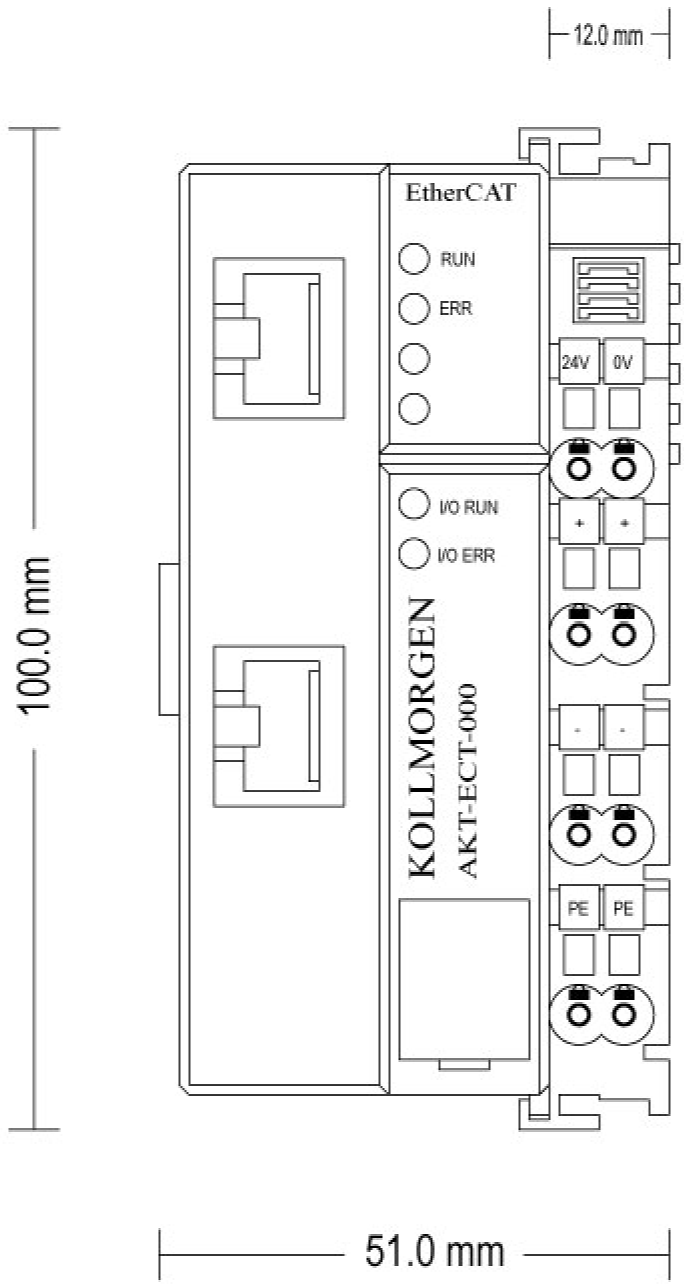
LED Red; I/O Error	Error Code Argument	Description	Remedy
Persistent, continuous blinking		EMC problems	– Check power supply for overvoltage or under voltage peaks – Implement EMC measures – If a Standard-Bus error is present, it can be localized by a restart of the coupler (by switching it off and then on again)
1 pulse	0	EEPROM checksum error	Set manufacturer's setting with the configuration software
	1	Code buffer overflow	Insert fewer Bus Terminals. The programmed configuration has too many entries in the table Software update required for the Bus Coupler
	2	Unknown data type	Software update required for the Bus Coupler
2 pulses	0	Programmed configuration has an incorrect table entry	Check programmed configuration for correctness
	n (n > 0)	Table comparison (Bus Terminal n)	Incorrect table entry
3 pulses	0	Standard-Bus command error	– No Bus Terminal inserted – One of the Bus Terminals is defective; halve the number of Bus Terminals attached and check whether the error is still present with the remaining Bus Terminals. Repeat until the defective Bus Terminal is located.
4 pulses	0	Standard-Bus data error, break behind the Bus Coupler	Check whether the n+1 Bus Terminal is correctly connected; replace if necessary.
	n	Break behind Bus Terminal n	Check whether the Bus End Terminal is connected.
5 pulses	n	Standard-Bus error in register communication with Bus Terminal n	Exchange the nth Bus Terminal

LED Red; I/O Error	Error Code Argument	Description	Remedy
14 pulses	n	nth Bus Terminal has the wrong format	Start the Bus Coupler again, and if the error occurs again then exchange the Bus Terminal
15 pulses	n	Number of Bus Terminals is no longer correct	Start the Bus Coupler again. If the error occurs again, restore the manufacturers setting using the configuration software
16 pulses	n	Length of the Standard-Bus data is no longer correct	Start the Bus Coupler again. If the error occurs again, restore the manufacturers setting using the configuration software

APPENDIX A

This section provides the mechanical drawing of the I/O terminal.

A.1 AKT-ECT-000-000 Mechanical Drawing



About Kollmorgen

Kollmorgen is a leading provider of motion systems and components for machine builders. Through world-class knowledge in motion, industry-leading quality and deep expertise in linking and integrating standard and custom products, Kollmorgen delivers breakthrough solutions that are unmatched in performance, reliability and ease-of-use, giving machine builders an irrefutable marketplace advantage.

For assistance with your application needs, contact us at: 540-633-3545, contactus@kollmorgen.com or visit www.kollmorgen.com

North America

Kollmorgen

203A West Rock Road
Radford, VA 24141 USA
Phone: 1-540-633-3545
Fax: 1-540-639-4162

Europe

Kollmorgen

Wacholderstraße 40 – 42
40489 Düsseldorf Germany
Phone: + 49 (0) 203-9979-235
Fax: + 49 (0) 203-9979-3314