

# Source Code

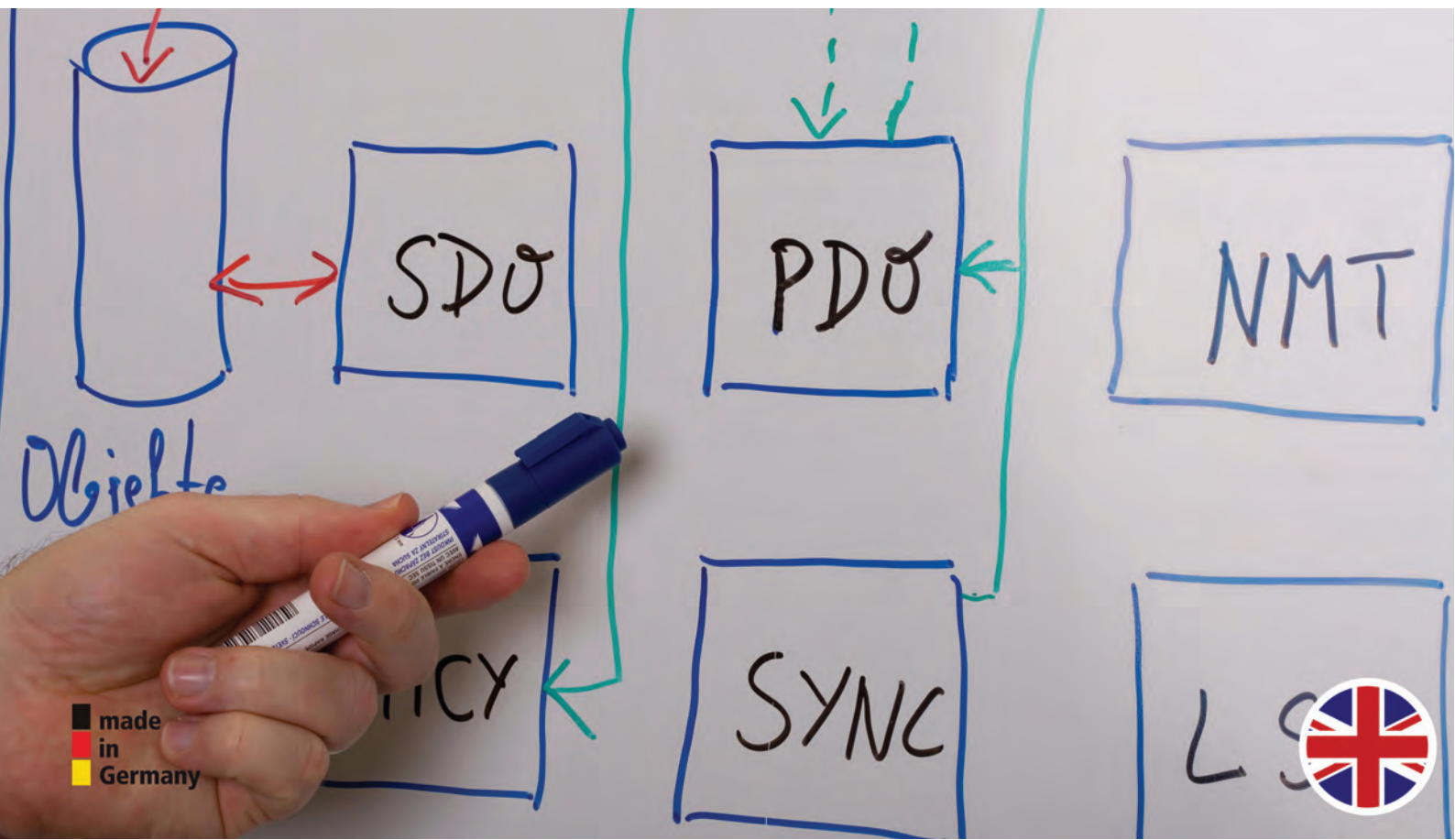
**CANopen**

SAE J1939



**EtherCAT**

**MicroControl**  
Systemhaus für Automatisierung





## Source Code

# CANopen®

Numerous companies worldwide benefit from our CANopen protocol stacks. The CANopen protocol stacks have been developed to combine both low storage requirements and an optimized performance range.

We offer the solution for various needs: our **CANopen Master** for complex control systems, **CANopen Slave** for the development of intelligent sensors and actuators and **CANopen Bootloader** for programming via the CAN interface.

CANopen protocol stacks are delivered as ANSI-C source code and can be implemented on any number of target platforms without acquisition of runtime licences.

# SAE J1939

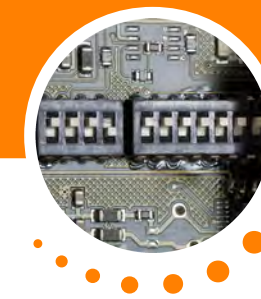
The **SAE J1939** source code comprises the complete functionality for the integration of the SAE J1939 standards into your own devices. Easy setting options facilitate convenient adjustment to a target product.

Optional add-ons allow creation of ECUs for **ISOBUS** or **NMEA2000** applications. The Address Claiming protocol is supported completely.

The J1939 protocol stack is delivered as ANSI-C source code and can be implemented on any number of target platforms without acquisition of runtime licences. CAN is connected to the target hardware via our standard CANpie driver.

	CANopen Slave	CANopen Master	CANopen Bootloader
Application range	Intelligent sensors and actuators with CANopen	Embedded control systems with CANopen	Firmware Update via CANopen
Progr. language	ANSI-C (C90)	ANSI-C (C90)	ANSI-C (C90)
Standards	CiA 301, CiA 305	CiA 301, CiA 302, CiA 305	CiA 301, CiA 302, CiA 305
Services	NMT, SDO, PDO, SYNC, EMERGENCY, TIME, LSS	NMT, SDO, PDO, SYNC, EMERGENCY, TIME, LSS	NMT, SDO, EMERGENCY, LSS
CAN Driver	CANpie	CANpie	CANpie
Functional range	Pre-defined Connection Set PDO-Mapping, LSS	Pre-defined Connection Set Boot Manager, LSS	Pre-defined Connection Set LSS
Options	Device profiles Mini-Master	Command Line API (CLI)	Flash-API

## Source Code



The **DeviceNet** Slave protocol stack speeds up the development process of devices which have to comply with the ODVA CIP standard. Support of I/O Messages (Poll / Bit-Strobe / COS / Cyclic) can be suited to the individual application. Functionality can be enhanced by adding your own classes.

The DeviceNet protocol stack is delivered as ANSI-C source code and can be implemented on any number of target platforms without acquisition of runtime licences.

Functionality of the protocol stack is verified through the Conformance test tool of ODVA. The CAN interface is connected via our standard CANpie driver.



Our **EtherCAT** slave equipped with full-range CoE functionality has been developed on the code base of our well-established CANopen slave protocol stack. Via a defined interface the EtherCAT slave controller (ESC) can be connected to any microcontroller device.

The CoE object directory can be enhanced by device profiles. The CoE implementation supports SDO information protocol as well as variable PDO assignment and mapping. Functionality of the stack has been verified via the EtherCAT Conformance test tool of ETG.

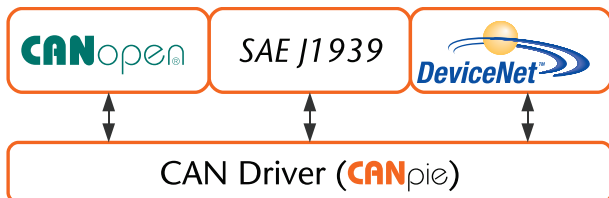
The EtherCAT stack is delivered as ANSI-C source code and can be implemented on any number of target platforms without acquisition of runtime licences.

SAE J1939	DeviceNet Slave	EtherCAT Slave
ECUs with J1939 / ISOBUS /NMEA2000 Interface	Sensors and actuators with DeviceNet interface	Application range Sensors and actuators with EtherCAT interface
ANSI-C (C90)	ANSI-C (C90)	Prog. language ANSI-C (C90)
SAE J1939 Transport (TP/BAM), Address Claiming	CIP Volume 1 and 3 Explicit & I/O Messages, Heartbeat	EtherCAT services Mailboxes, FMMUs Distributed Clocks
CANpie	CANpie	Driver API for ESC
Cyclic transmission Time monitoring	Identity, Message Router, DeviceNet, Assembly, Connection, Acknowledge	CoE functions variable PDO assignment variable PDO mapping online SDO information
PGN-Lists	Device profiles	Options CoE Device profiles



# MicroControl

Systemhaus für Automatisierung



The **CAN Driver CANpie** is an open interface for the development of CAN applications. It provides a basis for higher-layer protocols such as CANopen, Device-Net or SAE J1939 and it is available for more than 40 CAN controllers.

### Development Support

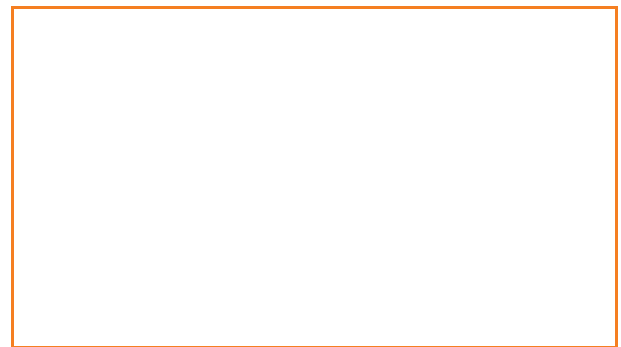
As an enhancement of our source code packages we offer additional services such as seminars, software-tools and technical support aimed at the development of your devices.

For further information please contact us!

Lindlaustraße 2 c  
53842 Troisdorf  
Germany

Tel.: +49 (0) 2241 256 59 - 0  
Fax: +49 (0) 2241 256 59 - 11

[info@microcontrol.net](mailto:info@microcontrol.net)  
[www.microcontrol.net](http://www.microcontrol.net)



sales partner

