

MicroCANopen Classic

Part No.:
MPESA-Embedded-Communi-Classic



Fully featured, professional CANopen stack. Ready to implement CANopen slave nodes with a host of features, including dynamic PDO mapping, non-volatile memory support, LSS and more. Comes with a driver and example application.

Efficient CANopen implementation with maximum performance. MicroCANopen Classic was developed to allow for maximum task optimization, especially when used with an RTOS such as CMX-RTX. Even without an RTOS, MicroCANopen Classic allows adapting the execution priorities of critical tasks making MicroCANopen Classic one of the best performing CANopen stacks available.

MicroCANopen Classic has been developed applying the Embedded Systems Academy spirit: provide an elegant solution minimizing the resources required and maximizing the performance. After all, if an implementation uses less code it also executes faster! The developers have the expertise and hands-on experience to satisfy the most stringent real time demands that the embedded community deals with every day. With MicroCANopen Classic they created a new level of performance and quality for CANopen implementations.

Efficient Implementation. The CANopen Process Data Objects (PDO) allow for a very flexible configuration allowing a single CANbus message to be filled with a combination of variables available in the Object Dictionary (OD) of a node. This PDO mapping process is implemented so efficiently in MicroCANopen Classic, that in most cases an incoming Receive PDO (RPDO) can be processed right in the CANbus interrupt service routine.

Advanced Task Control. All major tasks performed in MicroCANopen Classic are controlled from one single module. If MicroCANopen Classic is not used with an RTOS, the execution of tasks can still be optimized towards an application. In general, tasks can be executed within the CAN interrupt service routine, a timer

interrupt service routine or in the background. This even allows processing different PDOs at different priority levels.

Flexibility. When designing multiple CANopen devices that only vary slightly (for example in some Object Dictionary (OD) entries and/or in PDO configuration) it is desirable to only develop and maintain one version of the code. With MicroCANopen Classic the entire OD and the PDO configuration can be stored in non-volatile memory. As a result, the configuration of a node can be changed drastically just by downloading a configuration file (download via CANopen supported).

Portable. MicroCANopen Classic has been designed for maximum portability and is written in 100% standard C code. As a consequence, clients can decide to port the software to processors not yet supported, including microprocessors, microcontrollers and DSPs. Designers familiar with their target processor can easily perform the port themselves.

Compact. ROM Specifications: 8K - 18K bytes (for CANopen Slaves) depending upon processor, protocols used and options selected.

An Affordable Pricing Structure. MicroCANopen Classic offers a low, one-time fee and no royalties on deployed products. And you get the entire source code for with every purchase!

Features

The following is a list of features in MicroCANopen Classic. The list is not exhaustive by any means, but does give a good overview.

- NMT State Machine
- Heartbeat
- Node Guarding
- Object Dictionary (OD)
- Expedited SDO
- Segmented SDO
- Static PDO
- Dynamic PDO
- PDO with event time
- PDO with inhibit time

- PDO with SYNC
- Emergencies
- LSS
- Configuration of OD via non-volatile memory
- Configuration of PDO via non-volatile memory
- Central task control in one module
- NMT Master
- Extremely Small ROM/RAM Requirements
- Device Setup via Non-Volatile Memory
- Supports Virtually All 8-, 16-, 32-bit Processors and DSPs
- Use of Internal CAN Interface or External
- No Proprietary Protocols
- Runs Stand Alone or with RTOS
- Central Task Control
- Economical One Time Fee
- Full Source Code Provided
- No Royalties on Deployed Products
- Excellent Documentation and Support

Suggested Application Usage

MicroCANopen is best suited for minimal CANopen slaves that are pre-configured and do not need to be re-configured during operation. CAN baud rate, the node ID and all PDO parameters are known at implementation and hard-coded into the module. This limits the usage of MicroCANopen to applications where the CANopen node is used over-and-over with the exact same configuration.

MicroCANopen Plus is best suited for CANopen slaves requiring minimal configurability. CAN baud rate, the node ID and PDO communication parameters are configurable. This allows using MicroCANopen Plus for nodes that typically require some setup during installation of the node.

MicroCANopen Classic is best suited for full-featured, highly flexible CANopen slave nodes and for minimal CANopen NMT (Network Management) Master applications. Using a setup file, MicroCANopen Classic can be completely re-configured. Without re-compilation Object Dictionary entries can be modified, added or removed. This

allows one MicroCANopen Classic implementation to be used for a wide variety of devices.

What's Included

MicroCANopen Classic is delivered with two examples for implementations of the CANopen Device Profile DS401 - generic I/O. Additional and customized examples are available upon request and can include Device Profile Implementations such as Joysticks, Encoders (DS406), Batteries (DSP418), Chargers (DSP419) or Elevators/Lifts (DSP417).

All our examples pass the official CANopen Conformance Test!

Currently Available Ports

The following list is the microcontrollers and compilers that MicroCANopen Classic is currently available for. For additional combinations you can either perform the port yourself or we can perform it for you.

- Freescale 56F8300 w/ Metrowerks CodeWarrior
- C166 and XC166 w/ Keil
- C166 and XC166 w/ Tasking
- CC01 w/ Keil
- LPC21xx/LPC22xx w/ GNU
- LPC21xx/LPC22xx w/ Keil CARM
- Fujitsu w/ Softune
- Freescale PPC w/ Tasking
- Freescale S12 w/ Metrowerks CodeWarrior
- NXP XAC3 w/ Raisonance
- PC Simulation (with PCANopen Magic ProS)
- Renesas R8C23 w/ HEW (High-performance Embedded Workshop)

Comparison Between CANopen Implementations

Feature	MicroCANopen	MicroCANopen Plus	MicroCANopen Classic
All CANopen baud rates supported	✓	✓	✓
Network Management state machine with autostart option	✓	✓	✓
Heartbeat producer, [1017H]	✓	✓	✓
Heartbeat consumer, [1016H]		✓	✓
Node Guarding responses		✓	✓
Setup via hard-coding in program	✓	✓	✓
Setup via CANopen Task Setup File (read/write to [1F50H])			✓
Object Dictionary support for data types of up to 4 bytes (expedited SDO transfer)	✓	✓	✓
Object Dictionary support for any data type		Segmented	Segmented and Block
PDO default configuration can be hard-coded	✓	✓	✓
Dynamic PDO Communication Parameters (write to [14xxH] and [18xxH] allowed)		✓	✓
Dynamic PDO Mapping			✓

Feature	MicroCANopen	MicroCANopen Plus	MicroCANopen Classic
Parameters (write to [16xxH] and [1AxxH] allowed)			
Store PDO parameters in non-volatile memory ([1010H], [1011H] and [1020H])		✓	✓
TPDO Trigger by Event Time	✓	✓	✓
TPDO Trigger by COS with Inhibit Time	✓	✓	✓
TPDO Trigger by SYNC		✓	✓
TPDO Trigger by RTR			✓
Emergency Producer, [1014H] and [1015H]		✓	✓
Emergency Consumer, [1028H]			✓
Layer-Setting Services		Regular and FastScan	Regular
SDO Client		With optional Manager Add-On	✓
Implements NMT Master		With optional Manager Add-On	✓
Common CAN driver interface		✓	✓
Flexible integration into RTOS		✓	✓
Maximum number of PDOs	8	1024	1024

Feature	MicroCANopen	MicroCANopen Plus	MicroCANopen Classic
Maximum size of process image storing all data that can be mapped to PDOs (in bytes)	254	65,534	65,534
Smallest timer tick resolution available	1ms	1ms	100us
Minimal SDO Manager			✓
DS447 Car Add On Devices Support		With optional DS447 Add-On	
Error and Emergency History, [1003H]		✓	



標高電子股份有限公司
 Microport Computer Electronics Inc.
 Tel/06-2753783 Fax/06-2375031
<http://www.microporttaiwan.com>