

CANopen Bootloader

Modern device designs need enormous flexibility in hard- and software. Today hardware is equipped with plenty of resources in order to allow software changes later on. Software itself has to provide means to enable an update/upgrade of the current firmware. At this point in-system-programming offers the greatest possibilities, and thus a change of the software too.

The desire for this feature also exists for CANopen devices and it is easy to implement. With the SDO transfer CANopen provides all means to transmit large chunks of data that occurs with a software update.

This CANopen binary bootloader implements a minimal node in a CANopen network as specified in the CiA Draft Standard Proposal 302 "Framework for Programmable CANopen Devices". It allows to use regular CANopen master software or configuration tools to download new firmware into the user flash code memory.

First implementations for the TMS320F28x DSPs of Texas Instruments, Fujitsu MB90xx and the XC164CS family of Infineon, are available.

The free version, available as binary FLASH image, has only few limitations:

- use of predefined bit rates. Configuration is done through writing the BTR values to FLASH. (When purchasing the source code the software can be adopted to manufacturer specific methods to determine bitrates)
- The bootloader has to be linked on a fixed address within the user code flash memory.
- Some object directory entries are linked statically and cannot be changed.

The full version without restrictions can be obtained from *port-GmbH* via licensing agreement and doesn't have the restrictions mentioned above. For details please contact service@port.de.

Hardware Requirements

Through the use of the hardware independent *CANopen Sourcecode Library* of *port* the bootloader can be used on all supported target platforms. Solely the flash routines have to be adopted. Please inquire for the availability of the targets.

As a client a PC interface hardware (USB-CAN CPC-USB or USB-XS, PCI, ISA) or a gateway according to the CiA 309-3 (EtherCAN) is needed.

Software Requirements

A CANopen software able to do SDO domain transfers and supported by a CAN interface board is needed. A download program runnable under LINUX™ or Windows™ as well as some other tools,

e.g. to generate a CRC checksum, can be downloaded from *port's* web-site.

Installation and Setup

In the upper area of the bootloader the configuration areas are located. Adjust the values of the configuration according your requirements with a hex-editor.

```
struct {
/* 0xC0'3FE0 */  UNSIGNED32 vendor_id;
/* 0xC0'3FE4 */  UNSIGNED32 product_code;
/* 0xC0'3FE8 */  UNSIGNED32 revision_number;
/* 0xC0'3FEC */  UNSIGNED32 serial_number;
/* 0xC0'3FF0 */  UNSIGNED8  nodeid;
/* 0xC0'3FF1 */  UNSIGNED8  btr0;
/* 0xC0'3FF2 */  UNSIGNED8  btr1;
} USER_CONFIGURATION_T;
```

After filling in the configuration data you should now be able to talk to the node using your CANopen configuration tool.

This is the list of implemented Object Dictionary entries:

Index	Sub	Datatype	Acc	Function
0x1000	0x00	u32	co	Device Type
0x1001	0x00	u8	ro	Error Register
				Error Field
0x1003	0x00	u8	ro	Number of Entries
0x1003	0x01	u32	ro	Standard Error Field
0x1008	0x00	vs	co	Device Name
0x1009	0x00	vs	co	Hardware Version
0x100a	0x00	vs	co	Software Version
0x1014	0x00	u32	ro	COB-ID EMCY
0x1017	0x00	u32	rw	Producer HB time
				Identity Object
0x1018	0x00	u8	co	Number of Entries
0x1018	0x01	u32	co	Vendor ID
0x1018	0x02	u32	co	Product Code
0x1018	0x03	u32	co	Revision Number
0x1018	0x04	u32	co	Serial Number
0x1F50	0x00	u8	ro	Number of Entries
0x1F50	0x01	domain	wo	Download Program Area
0x1F51	0x00	u8	ro	Number of Entries
0x1F51	0x01	u8	wo	Program Control

Note: The memory optimized Bootloader does not contain all optional objects and does not implement services not mandatory.

Device Type

Contains 0xFFFF as Bootloader profile id in the lower part of the "Device Profile" area. In the upper part there is the identification of the download capabilities, e.g. possible data formats.

Error Register

Always 0. An error during programming is not indicated here but rather through an "SDO Abort" message or an "EMCY" message.

Standard Error Field

Contains the error code of the lately sent EMCY

Manufacturer Device Name**Manufacturer Hardware Version****Manufacturer Software Version****Heart Beat Producer Time**

Period for sending heartbeat messages. This index is configurable via CANopen.

Identity Object

Vendor ID port . Reads back 0x0000'0034, *port's* official vendor ID assigned by the CiA.

Product Code

At time of delivery the object contains a number of *port*-GmbH . This value can be configured in the configuration area.

Download Program Area

Use this entry to download your Intel hex file with the programming data. Please note that the download area is located on a fixed address. Download file format is binary, with an application header prepended.

Program Control

Writing 0x01 into this entry will start the loaded application. Writing 0x04 into this entry will "ERASE" the application FLASH area.

Implementation Details and Application Requirements - XC164CS

The bootloader is linked at address 0xC0'0000. The user code flash area begins at 0xC0'4000. The first 128 bytes, 0xC0'4000 to 0xC0'407F, are used for administration. It contains information about length of the application and a cyclic redundancy checksum.

```
struct {
    UNSIGNED32 length;
    UNSIGNED16 crc;
    UNSIGNED16 applicationType; /* reserved */
    UNSIGNED32 addr;           /* reserved */
} APPLICATION_HEADER_T;
```

This area is filled with 0x00. The application has to be linked to the flash area 0xC0'4080 to 0xC0'1FFF. The interrupt vector table should start at 0xC1'0000. The bootloader jumps at the start address 0xC1'0000.

EDS File

The file [Bootloader.eds](http://www.port.de/engl/canprod/sw_bootloader.html) (http://www.port.de/engl/canprod/sw_bootloader.html) is a standard CANopen **Electronic Data Sheet** that can be used with a CANopen configuration tool, like the [CANopen Device Monitor](http://www.port.de/engl/canprod/sw_monitor.html) (http://www.port.de/engl/canprod/sw_monitor.html), to allow immediate access to the object dictionary entries.

Download

The binary and Intel-hex file of the bootloader and some tools can be downloaded from our Web Site.

cksum

Checksum program LINUX executable

cksum.exe

Checksum program Windows executable

downloader

LINUX command line Downloader; transfers a file using Domaintransfer

downloader.exe

Windows command line Downloader for EMS interface boards; transfers a file using Domaintransfer

Tools

There are two additional programmes to assist with the program download.

cksum produces a complete "ready to download file" with prepended application header from the binary image. This header contains information about length and checksum, which are stored in flash memory

Use cksum:

```
C:> cksum -C -O download.bin application.bin
```

Further information are given using the option **-h**.

downloader.exe executes the SDO domaintransfer as a client. This program takes the file as argument and transmits it. The bootloader writes the received programme into the flash memory.

Ordering Information

0261/01	Bootloader Binary Infineon XC164
0261/02	Bootloader Binary Fujitsu MB90F5xx
0261/03	Bootloader Binary TMS320F2812
0261/04	Bootloader Binary LPC2129
0261/05	Bootloader Binary TMS320F2808
0261/06	Bootloader Binary Atmel AT91SAM7X256
0261/50	Generic Bootloader Source Code
0261/51	Bootloader Source Code Infineon XC164 (for Tasking Compiler)
0261/52	Bootloader Source Code Fujitsu MB90F5xx
0261/53	Bootloader Source Code TMS320F2812
0261/55	Bootloader Source Code Infineon XC164 (for Keil Compiler)
0261/56	Bootloader Source Code Philips LPC2129 (for Keil Compiler)
0261/57	Bootloader Source Code TMS320F2808
0261/58	Bootloader Source Code Atmel AT91SAM7X256 (for Crossworks)
0261/60	Downloader Source Code

Engineering Services

port is providing engineering services and trainings for our business activities:

- CAN and CAN-based protocols: CANopen, J1939, DeviceNet
- Industrial Ethernet Protocols: POWERLINK, EtherNet/IP, EtherCAT
- Implementation of devices according to CANopen device profiles
- VHDL based solutions for industrial applications
- application specific implementations or enhancements
- embedded LINUX projects

Notice

Brands and product names are trademarks or registered trademarks of their respective companies. The product will be continuously improved. *port* therefore reserves the right to change technical properties at any time without appointment.



port GmbH
Regensburger Straße 7b
D-06132 Halle/Saale
+49 345 777 55 0
service@port.de