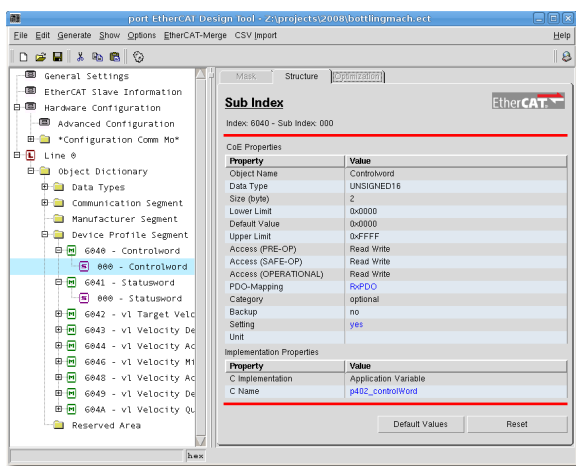


Overview

The EtherCAT Design Tool is a tool for the rapid and cost-saving development of **EtherCAT®** applications (devices). It administers device databases, from which an object dictionary and an initialization function in C-code, the EtherCAT Slave Information (ESI) file, an Electronic Data Sheet and the documentation are produced automatically. Furthermore it simplifies the configuration of the EtherCAT Library® and of the EtherCAT® Driver Packages.



Description

With the EtherCAT Design Tool a tool is available, which frees the developer from error-prone and repetitive tasks. It guarantees the consistency of the implemented functionality, device documentation and Device Description. By default the EtherCAT® communication profile is delivered with the tool.

In addition to the EtherCAT® specific communication database other databases are provided for several CANopen device profiles such as CiA 401, CiA 402, CiA 404 and CiA 406. A tree representation of all implemented parameters and data facilitates the maintenance of the device software. The EtherCAT Design Tool eases the first step into the EtherCAT® protocol and device development is accelerated.

It is a tool with which device databases can be produced and processed. These databases contain information, which describe the interface for the EtherCAT® network of the device. These parameters, control and status information that are accessible over the network are stored in the **Object dictionary**.

The major task of the EtherCAT Design Tool is to administer and create these data in an object database

and to create code for the EtherCAT Library® from it. Furthermore the EtherCAT Design Tool is used to configure the *CANopen Driver Packages*. Within one project several hardware configurations can be managed. This allows a quick change between different targets during development.

EtherCAT® Object dictionary

The generated code, which contains the object dictionary is included into the user modules. This ensures that each user variable can be accessed directly via its variable name or via its index and sub index. The produced object dictionary represents the data interface between library and user application.

EtherCAT® Library Configuration

To configure the EtherCAT Library® a header file is created which contains the hardware settings and the configuration of the used EtherCAT® services. These services are derived automatically from the object dictionary. By including this file into the project only the necessary parts of the EtherCAT Library® are compiled in order to save RAM and ROM resources.

Documentation

Based on the generated object dictionary a manufacturer specific documentation is created. Each device parameter is listed with its attributes and description. Output format is HTML and plain text. This documentation can serve as specification at the beginning of development and be brought into the user documentation, as well as into the marketing documents. Therefore writing twice is not necessary and the contents are always up to date.

Device Profiles

Creating of device databases is very simple. The first step is determining the type of the device. If a standardized profile is to be implemented, then the device data can be loaded from a profile database. Furthermore it is possible to load data from device databases or already existing ESI files (XML).

The second step is the parameterization of the objects. Pre-installed default values, limit values and access attributes have to be adapted to the application needs. For the communication parameters input masks facilitate the creation of the service objects and the configuration of partially bit-coded values like e.g. Sync-Manager settings

Generation

After an automatic check of the data for consistence, missing entries or incompatible data types the following files are created:

- the object dictionary (**ec_od.c**, **ec_objects.c**, **ec_objects.h**),
- the initialization file (**ec_init.c**),
- the configuration file (**ecat_conf.h**),
- the ESI file,
- the binary EEPROM-File,
- and documentation as HTML and plain text.

If the check fails, errors and warnings are written into the log file **generate.err**. The results can be viewed with the internal file viewer with syntax highlighting.

Profile Databases

The profile databases contain the objects of the corresponding profile. Each object is configured according to the standard and is explained in detail. For every kind of objects in a profile a prototype is included and this object can be cloned by copying or duplicating. By importing these prefabricated objects into the own project the development time can be reduced and the risks of a manual implementation are minimized. Several profile databases are already available for example:

- CiA401 - Device Profile for Generic I/O Devices
- CiA402 - Device Profile for Drives and Motion Control
- CiA404 - Device Profile for Measuring Devices and Closed-Loop Controllers
- CiA406 - Device Profile for encoders

Advantages

- acceleration of the development by avoidance of double work
- error prevention during creation of the object dictionary and ESI file
- consistent documentation to the respective device version
- reuse of the data for similar devices
- optimized configuration of the EtherCAT Library® according to the implemented EtherCAT® services
- optional CSV import PlugIn available

System Environment

The EtherCAT Design Tool runs on PC's with

Operating System: Windows™7, LINUX™
Processor: Pentium IV or better
RAM: 512 MByte
Harddisk Space: >40 MByte available

Ordering Information

0290/10	EtherCAT®-DT-W (Full Version Windows™)
0290/20	EtherCAT®-DT-L (Full Version LINUX™)
0640/11	CANopen-DT-P401 profile database for CiA 401
0640/12	CANopen-DT-P402 profile database for CiA 402
0640/13	CANopen-DT-P406 profile database for CiA 406
0640/21	CANopen-DT-P417 profile database for CiA 417
0640/23	CANopen DT-P404 profile database for CiA 404
0640/24	CANopen DT-P405 profile database for CiA 405
0640/26	CANopen DT-P410 profile database for CiA 410
0640/28	CANopen DT-P418 profile database for CiA 418
0640/29	CANopen DT-P419 profile database for CiA 419

Functional demo versions of the software tools are available for download on www.port.de/shop.

Engineering Services

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

CAN and CAN-based protocols: CANopen, J1939

Industrial Ethernet Protocols: POWERLINK, EtherNet/IP, EtherCAT PROFINET

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