CiA Draft Standard Proposal 417-1

CANopen

Application Profile for Lift Control Systems

Part 1: General definitions and physical layer specifications

This is a draft standard proposal and may be changed wihtout notification.

Version 1.0.1

July 15, 2003

© CAN in Automation (CiA) e. V.

HISTORY

Date	Changes
2003-04-15	deleted network overview as it will not be standardized
2003-04-19	final preperation
2003-07-15	change version numbering

CONTENTS

1   Scope

The CANopen application profile for lift-control systems consists of several parts:

>   Part 1 describes general definitions and physical layer specifications

>   Part 2 defines virtual devices and assignment of application objects

>   Part 3 defines pre-definitions of communication objects and PDOs

>   Part 4 describes application objects in detail

These documents represent the CANopen application profile for lift-control systems. Part 2 to part 4 describes virtual devices for a lift-control system. All physical devices compliant to this standard use communication techniques, which conforms to those described in the CANopen application layer and communication profile /CiA301/. Programmable devices may use communication techniques, which conform to those described in the framework for CANopen managers and programmable CANopen devices /CiA302/. If the device transmits or receives safety-relevant data, it shall be compliant to the CANopen framework for safety-relevant communication /CiA304/. These specifications should be consulted in parallel to those device profile specifications.

2   Normative references

/CiA301/   CiA DS 301 V4.02, CANopen application layer and communication profile, June 2002.

/CiA302/   CiA DSP 302 V3.2.1, Framework for CANopen managers programmable CANopen devices, April 2003.

/CiA304/   CiA DSP 304 V1.0, Framework for safety-relevant communication, January 2001.

/CiA305/   CiA DSP 305 V1.1, Layer setting services (LSS), February 2002.

/CiA3031/   CiA DR 303-1 V1.1 Cabling and connector pin assignment, April 2001.

/CiA3032/   CiA DR 303-2 V1.1, Representation of SI units and prefixes, January 2000.

3   Definitions, acronyms and abbreviations

CAN   Controller Area Network: Data link layer protocol for serial communication as specified in ISO 11898-1 (2002)

COB   Communication Object: Is made of one or more CAN frames; any information transmitted via CANopen has to be mapped into COBs

COB-ID   COB-Identifier: Identifies a COB uniquely in a CAN network; the identifier determines the priority of that COB in the data link layer, too

RPDO   Receive Process Data Object: Communication object of a device, which contains output data

SDO   Service Data Object: Peer-to-peer communication with access to the Object Dictionary of a CANopen device

TPDO   Transmit Process Data Object: Communication object of a device, which contains input data

4   Hardware preferences

4.1   Physical layer

Physical devices shall implement ISO 11898-2 compliant transceiver chips capable to drive in minimum 64 nodes.

4.2   Bit rate

Auto-baud detection shall be supported. No default bit rate is specified. All physical devices shall be able to support 20 kbit/s, 50 kbit/s, 125 kbit/s, and 250 kbit/s.

4.3 Bus topology

No specific bus topology is defined. The network may be segmented and/or cascaded.

4.4 Bus cable

No specific bus cable is defined. It is recommended to consider the cabling recommendations as given in /CiA3031/. For network segments connecting only panel and display devices patch cables of category 5 are recommended.

4.5 Bus connector

No specific bus connector is defined. However it is recommended to use DSUB9, RJ45, and open style connector. If other connectors are used, it is recommended to apply the pin assignments as given in /CiA3031/.

5 Node-ID assignment

5.1 Introduction

There is no standardized node-ID assignment defined. The node-ID may be hardwired, settable by switches, or by manufacturer-specific methods.

In order to achieve an off-the-shelf plug-and-play capability it is recommended to implement in each physical CANopen slave device LSS slave services as defined in /4/. Using LSS, it is recommended storing the assigned node-ID in non-volatile memory.

6 Error handling

6.1 Principle

Emergency Messages shall be triggered by internal errors in the physical device and they are assigned to high priority to ensure that they get access to the bus without latency. By default, the Emergency Messages shall contain the error field with pre-defined error numbers and additional information.

6.2 Error behavior

If a serious device failure is detected the physical device shall enter by default autonomously the pre-operational state. If object 1029_h is implemented, the physical device can be configured to enter alternatively the stopped state or remain in the current state in case of a device failure. Device failures shall include the following communication errors:

    Bus-off conditions of the CANopen interface

    Life guarding event with the state 'occurred'

    Heartbeat event with state 'occurred'

Severe device errors also can be caused by device internal failures.

6.3 Additional error code meanings

Devices compliant to these gateway profile specifications may use the following error codes:

Error Code	Description
$FF01_h$	Light barrier defect
$FF02_h$	Finger protector defect
$FF03_h$	Motion detection defect