Holger ZELTWANGER
CAN IN AUTOMATION GMBH, NUREMBERG

Hints and kinks for CANopen system designer

Holger ZELTWANGER

Holger Zeltwanger (born 1952) finished his studies at the Fachhochschule Braunschweig/Wolfenbuettel (Germany) in 1976 with a diploma in electronic engineering. He worked at Siemens as a system programmer for two years. After that he was a technical editor for German and American magazines for more than 14 years. In 1992 he founded the international users' and manufacturers' group CAN in Automation (CiA). Since then he has worked as Managing Director for CiA.

e-mail: headquarters@can-cia.org

Abstract

Although the CANopen application layer is well specified, the system designer could make failures or misuse the services and protocols. This is not only possible for the application layer, but also for the data link and the physical layer. This paper provides some hints and kinks designing CANopen network system.

Keywords: CANopen, CAN, network design.

Porady i ostrzeżenia dla projektanta systemu CANopen

Streszczenie

Chociaż magistrala CANopen is dobrze wyspecyfikowana, to jednak projektant systemu może popełnić pomyłki lub błędnie zastosować usługi i protokoły. Może się to zdarzyć nie tylko w warstwie aplikacji, ale także w łączu danych i w warstwie fizycznej. W artykule przedstawiono niektóre porady i ostrzeżenia przydatne w projektowaniu systemu sieciowego CANopen.

Słowa kluczowe: CANopen, magistrala CAN, projektowanie systemów.

1. Physical layer – don't exceed the limits of not terminated stubs

The design of the CAN physical layer is the most critical part of the network design. You have to select carefully the bus-line cable, the connector, and all other electro-mechanical components including the termination resistors at both ends of the bus-lines. The CAN standards (ISO 11898-1 and ISO 11898-2) do not specify in detail the physical layer. For CANopen networks you find some recommendations in the CiA 303-1 specification. In particular, connector pin assignments and some general hints are given.

Tab. 1. CANopen bit-timing, network and stub length limits
Tab. 1. Prędkość transmisji bitów oraz graniczne długości magistrali i doprowadzeń końcowych magistrali CANopen

Data rate	Max bus length	Nominal sample point	Sample point tolerance	Maximal stub length	Maximal accumulated stub length
1 Mbit/s	25 m	87,5%	75% to 90%	1,5 m	7,5 m
800 kbit/s	50 m	87,5%	75% to 90%	2,5 m	12,5 m
500 kbit/s	100 m	87,5%	85% to 90%	5,5 m	27,5 m
250 kbit/s	250 m	87,5%	85% to 90%	11 m	55 m
125 kbit/s	500 m	87,5%	85% to 90%	22 m	110 m
50 kbit/s	1000 m	87,5%	85% to 90%	55 m	275 m
20 kbit/s	2500 m	87,5%	85% to 90%	137,5 m	687,5 m
10 kbit/s	5000 m	87,5%	85% to 90%	275 m	1375 m

If you select a bus-line cable with an impedance matching the termination resistor values, and you use one of the defined sample points for a given bit-rate, than you may use the recommendations regarding the maximal length of not terminated cable stubs. Table 1 shows the maximal allowed length for different bit-rates for single stubs as well as for all stubs in the network. It is recommended to consider a safety-margin. The table also provides the sample-points for the bit evaluation at different bit-rates. If a device is exceeding the given tolerance for the sample point, it may happen that the frames on the CAN network are destroyed. In general, it is wise not to go to the limits of the specified CAN physical layer parameters.

The maximum length of the terminated bus-lines is determined by the selected bit-rate due to the in bit-time detection of the bit value. This requirement is necessary because of the arbitration method used in data link layer protocols. As rule of thumb, you can say: The higher the bit-rate, the lower the network length. In order to maximize speed and length, you may use in large systems repeaters, which shorten the overall length of the network (see Figure 1). Of course, you have to consider in your network topology the repeater as a delay element. You can regard a repeater as a piece of cable: Each 5 ns represent an equivalent of a 1-m cable length.

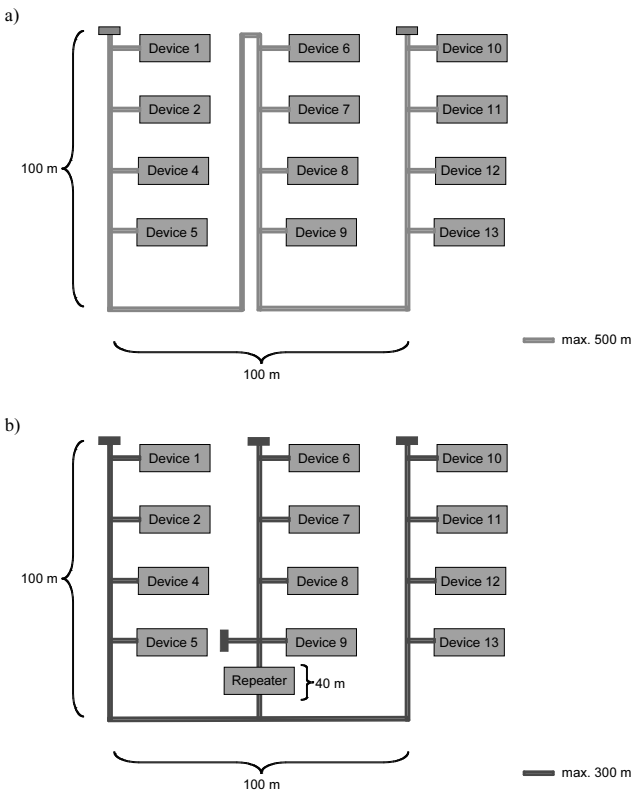

Fig. 1. Network topology without repeater (a) and with repeater (b)
Rys. 1. Topologia sieci bez wzmacniaka (a) i ze wzmacniakiem (b)

Another possibility is to divide the entire network into different segments interconnected by means of bridge/router devices. The CiA 302-7 specification describes Remote SDOs (service data object) and Remote Emergency messages as well as NMT and Heartbeat request services by means of SDO. Additionally the specification defines the forwarding of PDOs (process data object) from one segment to another one. Because of timing constraints, it is not recommended to use heavily the forwarding of PDOs.

2. Data link layer – don't use remote frames

The CAN data link layer, so to say the CAN protocol, is well defined and stable for a very long time – more than 15 years. However, there is one protocol, which makes a lot of headache: The remote frame protocol. The remote and the requested data frame use the very same CAN identifier. Remote frames have the same structure as data frames with two exceptions: The remote frame as never a data field (it consists always of zero byte), and the DLC (data length code) has to match the DLC of the requested data frame. However, some device designers implement a DLC of 0 (because the data field has a length of 0). If two devices send at the very same moment the same remote frame, and one implements the correct DLC corresponding to the requested data frame and the other use a DLC of 0, than one device is caused to send an error flag. Both devices will retransmit the remote frame automatically. Of course, the one device will detect again a bit-failure and sends another error flag. Each error flag transmission causes and increase of the transmit error counter, which leads at the end to a bus-off condition.

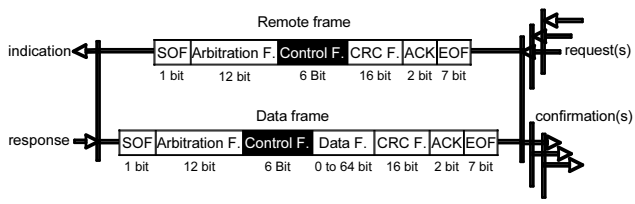

Fig. 2. The DLC (data length code) provided in the Control Field of a Remote Frame must match the DLC of the requested Data Frame
Rys. 2. Kod długości danych DLC podany w polu kontrolnym ramki urządzenia odległego musi być dopasowany do kodu DLC w polu kontrolnym urządzenia zapytywanego

There is another problem with remote frames: They are differently implemented regarding the response. Some CAN controllers with receive buffer or receive FIFOs answers remote frames only under CPU control. This means, they forward the request to the application. There is no determined time, when the request will be served. Therefore, other chipmakers have developed CAN controller, which serves remote frames without CPU control. They send whatever is in the corresponding message buffer. It could be historically, in the case that the CPU was not able to update the message buffer. More badly some CAN implementations provide advanced message storing capability and answer remote frames either automatically or optionally under CPU control. Due to the limited quality of device documentation, in many cases you will not find a detailed description of remote frame behavior. Therefore CiA does not recommend requesting PDOs remotely. In addition, CiA recommends to use the Heartbeat services and instead of the old-fashioned node/life-guarding services.

3. Application layer – don't double-use CAN-IDs

The CANopen communication objects (COB) have to be assigned uniquely with CAN identifiers (CAN-ID). In order to simply this assignment, the CANopen specification defines a set of pre-defined CAN-IDs. The resulting CAN-IDs are deriving from function codes and node-IDs. The system designer is responsible to assign the node-IDs uniquely, so that no CAN-ID is used twice. Table 2 shows the pre-defined CAN-IDs. Nevertheless, the system designer may reconfigure all CAN-IDs except those, which are defined as fixed. Fixed CAN-IDs are those for the NMT message, the Heartbeats, and the Default SDO servers. However, if reconfiguring the CAN-IDs the system designer has to ensure that no CAN-ID is transmitted by to devices; otherwise there could be an unsolvable bus arbitration conflict. This would make one device going into bus-off mode.

Tab. 2. Pre-defined CAN-ID set for CANopen communication objects (COB)
Tab. 2. Predefiniowany zestaw identyfikatorów dla obiektów komunikacyjnych (COB) magistrali CANopen

COB	Function code	Node-ID	Resulting CAN-Ids
NMT	0000_b	00_h	000_h
Sync	0001_b	00_h	080_h
Time	0010_b	00_h	100_h
1st TPDO	0010_b	01_h *to* $7F_h$	181_h *to* $1FF_h$
1st RPDO	0100_b	01_h *to* $7F_h$	201_h *to* $27F_h$
2nd TPDO	0101_b	01_h *to* $7F_h$	281_h *to* $2FF_h$
2nd RPDO	0110_b	01_h *to* $7F_h$	301_h *to* $37F_h$
3rd TPDO	0111_b	01_h *to* $7F_h$	381_h *to* $3FF_h$
3rd RPDO	1000_b	01_h *to* $7F_h$	401_h *to* $47F_h$
4th TPDO	1001_b	01_h *to* $7F_h$	481_h *to* $4FF_h$
4th RPDO	1010_b	01_h *to* $7F_h$	501_h *to* $57F_h$
SDO (tx)	1011_b	01_h *to* $7F_h$	581_h *to* $5FF_h$
SDO (rx)	1100_b	01_h *to* $7F_h$	601_h *to* $67F_h$
Heartbeat	1110_b	01_h *to* $7F_h$	701_h *to* $77F_h$

Tab. 3. Pre-defined CAN-ID set for CANopen application profiles
Tab. 3. Predefiniowany zestaw identyfikatorów dla profili aplikacyjnych magistrali CANopen

COB	Function code	Node-ID	Resulting CAN-Ids
NMT	0000_b	00_h	000_h
Sync	0001_b	00_h	080_h
Time	0010_b	00_h	100_h
1st TPDO *to* 512th TPDO	-	-	Profile-specific
1st RPDO *to* 520th RPDO	-	-	Profile-specific
SDO (tx)	1011_b	01_h *to* $7F_h$	581_h *to* $5FF_h$
SDO (rx)	1100_b	01_h *to* $7F_h$	601_h *to* $67F_h$
Heartbeat	1110_b	01_h *to* $7F_h$	701_h *to* $77F_h$

In some CANopen application profiles, the CAN-IDs of PDOs are not deriving from the node-IDs. In these specifications up to 512 PDOs are pre-defined regarding the CAN-ID (see Table 3). The other COBs use the CAN-IDs as specified in the pre-defined connection set. An application profile is a system and not a single device interface specification. The system designer has to ensure that each PDO is provided only by one single device.

How the node-IDs are assigned is not specified. The device designer have different options: Dip switches, separate serial interfaces, local HMI (human machine interfaces), etc. There are also two options to assign the node-ID via the CANopen interface. The Layer Setting Services (LSS) are specified in CiA 305. They require addressing each CANopen device uniquely in the world. For this purpose the CANopen application layer provides the mandatory identity parameter record (index 0x1018). It comprises the mandatory sub-index 0x01 containing the uniquely assigned vendor-ID as well as the product-code (sub-index 0x02), revision-number (sub-index 0x03), and serial-number (0x04). The identity parameter record is also necessary to claim a node-ID as specified in CiA 416. This node-ID claiming procedure uses also the CANopen interface.

4. PDO – don't transmit relative data

PDOs are used to transmit in real-time within one data frame measured data or commands. This is an unconfirmed

communication service. Therefore it could happen that one PDO is accidentally transmitted twice. This is because of the seldom scenario that the last bit of end of frame is detected as dominant. The CAN data link layer protocol specifies that the last bit of End of Frame (EOF) shall be transmitted recessive. If it is disturbed – meaning a dominant bit is detected – the receivers and the transmitter interpret this differently. The transmitting node regards the dominant bit in the last EOF bit as an error condition, and retransmits automatically the last data frame. All receiving nodes regard the dominant 7th EOF bit as an overload condition, because they have already accepted the data frame as correctly received after the 6th EOF bit (if recessive).

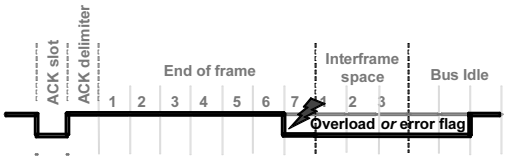

Fig. 3. If the last bit of EOF is disturbed dominantly, the transmitting node automatically retransmits the corrupted data frame, while the receiving nodes have already accepted the frame as correct

Rys. 3. Jeżeli ostatni bit ramki EOF jest istotnie zaburzony, to węzeł transmitujący automatycznie retransmituje zaburzoną ramkę danych, podczas gdy węzły odbiorcze już zaakceptowały ramkę jako poprawną

If the receiving nodes have accepted the data/remote frame already, and the transmitting node sends it automatically again, they receive the frame twice. In case of relative data this causes an unexpected behavior. For example: If the command is to increase the temperature by 20 °C and the heating system receives this twice, the temperature is increased by 40 °C. Therefore, you should only send absolute information. In this example: Commanding to increase the temperature to +25 °C will not lead to an undesired behavior, because the heating system will increase the temperature to +25 °C. This means don't send delta counts or any other relative data.

The same is with the digital commands. If a motor receives twice the command to switch-on ("1") or to switch-off ("0"), there is no problem. But if the first "1" will switch-on and the second "1" switches-off (so-called toggle-commands), you can't see motor running, because the automatic retransmission is just delayed by just 23 bit-times in worst-case for error active nodes. In error passive nodes it is in worst-case 31 bit-times.

The 7th EOF bit is necessary that receiving nodes can indicate a failure (if dominant) in the 6th EOF bit. This means to shorten the EOF doesn't help to solve the problem. Even it may happen very seldom, that the last EOF bit is dominantly disturbed, you should consider this possibility. Therefore, do not transmit relative data in CAN-based networks!

5. Conclusion

The CANopen physical layer gives the system designer some freedom. In particular, the network topology has to be carefully designed. It is highly recommended to assign the node-ID and the CAN-ID uniquely to the CANopen devices. Any double assignment may cause severe problems and misbehavior. The use of remote frames should be avoided, especially if you buy off-the-shelf products from different vendors. The system designer also should not transmit in any relative data. If you consider these design hints and kinks, you avoid a lot of trouble during network operation.

6. References

[1] ISO 11898-1 standard, published by ISO, Geneve 2003
[2] CiA 301 application layer and communication profile, published by CiA, Nuremberg 2007
[3] Zeltwanger H., CANopen das standardisierte eingebettete Netzwerk (2nd edition), published by VED Verlag, Berlin 2008

Artykuł recenzowany