

Approach for node-ID negotiation in CANopen

The increasing complexity and size of CANopen (FD) networks create new challenges on how to decide which node-ID should be used. This article discusses a theoretical approach how devices in a CANopen (FD) network could negotiate their node-ID by themselves.

Bit 28 to 13	Bit 12 to 8	Bit 7 to 0
reserved	Data code	Data field

Table 1: 29-bit CAN-ID usage

The [complete article](#) is published in the [September issue](#) of the CAN Newsletter magazine 2021. This is just an excerpt.

The assignment of a node-ID to a CANopen device is an essential requirement. Without a node-ID a CANopen device is unable to communicate over services defined in CiA 301 and CiA 1301. Without a valid node-ID in the range of 1 to 127, a CANopen device would never leave the NMT (net- work management) sub-state reset communication. Since the node-ID has to be unique in the network, there has to be a way to configure the node-ID value. Neither CiA 301 nor CiA 1301 specify a way to configure this value. There are various solutions on the market how this can be achieved.

The assignment of a node-ID to a CANopen device is an essential requirement. Without a node-ID a CANopen device is unable to

Some manufactures use hardware solutions on their devices, such as dip- or rotary-switches, or encode plugs. Others, use some kind of proprietary software configuration. All of these solutions need more or less knowledge about the system where the devices are integrated. Another way to configure the node-ID is specified in the document [CiA 305, CANopen layer setting services \(LSS\)](#). This specification describes services and protocols for identifying CANopen devices and assigning node-IDs, which can be used by a so called LSS manager (formerly named LSS master) or some configuration tool.

The CANopen application profile for building door control, CiA 416, also specifies a procedure for claiming node-IDs by the devices themselves. But besides that, this procedure is slow, it is also patented. The following theoretical approach for negotiation of node-IDs introduced by the engineers of Emotas Embedded Communication should be a starting point how to overcome most of the disadvantages of the other solutions.

Systems without a superordinated manager

Some systems are designed very flexible. That means that a system could be built with different combinations of devices, exactly meeting the needed requirements. Sometimes, it could be enough that only two devices are needed, sometimes much more, and sometimes there are multiple of the same device type in a network. An example of such kind of systems are heating systems based on CANopen. They can include multiple sensors and controllers depending on location and size where the system is installed.

Data code	1018 _n data
1000 _b	Sub-index 1, bit 0 to 7
1000 _{1b}	Sub-index 1, bit 8 to 15
1001 _b	Sub-index 1, bit 16 to 23
1001 _{1b}	Sub-index 1, bit 24 to 31
1010 _b	Sub-index 2, bit 0 to 7
1010 _{1b}	Sub-index 2, bit 8 to 15
1011 _b	Sub-index 2, bit 16 to 23
1011 _{1b}	Sub-index 2, bit 24 to 31
1100 _b	Sub-index 3, bit 0 to 7
1100 _{1b}	Sub-index 3, bit 8 to 15
1101 _b	Sub-index 3, bit 16 to 23
1101 _{1b}	Sub-index 3, bit 24 to 31
1110 _b	Sub-index 4, bit 0 to 7
1110 _{1b}	Sub-index 4, bit 8 to 15
1111 _b	Sub-index 4, bit 16 to 23
1111 _{1b}	Sub-index 4, bit 24 to 31

Table 2: Data code correlation with 1018_n

When creating the devices for such a system, it is not clear which kind of device could be unique in the system, so it can operate as a CANopen manager. The CANopen manager is the device with NMT manager functionality plus additional functionality, for example, the LSS manager. One way to assign node-IDs without an LSS manager is when integrating the system. But this leads to a static system, without a real plug-and-play possibility. And, in case some parts are added, or have to be replaced, an unused node-ID has to be assigned to the replaced part, which might be impossible for a service technician.

Another example are battery clusters, which are capable of plug and play. The only devices in this network are the batteries themselves. They are physically all of the same device type and have the same software. In such a system, the node-ID is irrelevant to the system, but is required for CANopen communication.

Systems with a superordinated manager

In generic CANopen systems with a superordinated manager, there are also possibilities that the actual distribution of the node-IDs is not relevant for the functionality of the system. Examples are the CANopen application profile for energy management systems, CiA 454, or the CANopen application profile for special-purpose car add-on devices, CiA 447. These specifications use the LSS Fastscan procedure, defined in CiA 305, to detect devices and to assign node-IDs to them. With LSS Fastscan devices can be only detected one after another, and in the worst case of completely unknown devices it will take 128 messages to verify every single bit of the CANopen LSS address. With a response-timeout of 10 ms, it would take 1,28 s to detect one single device.

In generic CANopen systems with a superordinated manager, there are also possibilities that the actual distribution of the node-IDs is not relevant for the functionality of the system. Examples are the CANopen application profile for energy management

New approach

The idea behind this approach is that, in systems which do not depend on the node-ID distribution, the devices negotiate their node-ID themselves. The idea is not new, for example in the document J1939-81 of the SAE (Society of Automotive Engineers), there is a description of a so-called address claim procedure. In this procedure, the devices in a CAN-based network negotiate the addresses (node-IDs), depending on values of the Name field. The problem here is that this procedure uses the 8-byte data field of the CAN

frame. This led to the fact, that, in case two or more devices are claiming the same address, it could lead to collisions on the CAN net- work.

In our approach, we are trying to avoid this by not using the data field: All the information exchanged between devices are encoded in the identifier field. The main requirements for this procedure are that the software of the devices is able to send and receive CAN data frames in classical extended frame format (CEFF). Because most modern CAN FD controllers support sending and receiving this kind of data frames, this approach works in CANopen FD as well. It is also important that the software can distinguish between extended and basic frame formats.

If you would like to read the full article from Emotas, you can [download](#) it free of charge or you [download the entire magazine](#).

[CW](#)