LIVIC – Laboratoire de recherche sur les Intéractions Véhicule - Infrastructure - Conducteur

INRETS – Institut National de Recherche sur les Transports et leur Sécurité
LCPC – Laboratoire Central des Ponts et Chaussées
Ministères Recherche & Transport

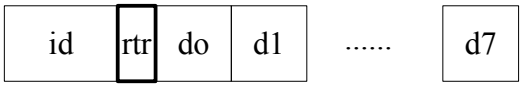
CANOpen Memento

Francis Dupin, November 2005

Version 1.2

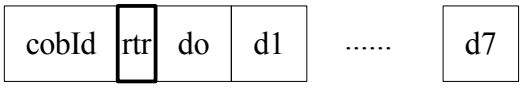
I have put on this document some of the tips to test a node or to configure a CANOpen network.
If you do not know how works CANOpen, this document will not help you at all.
Feel free to redistribute this paper ... and to report the errors and missings to francis.dupin@inrets.fr

The CAN message

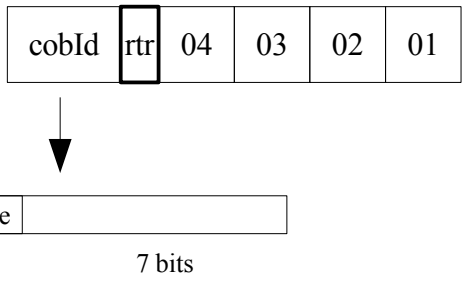


Id : the CAN identifiant of the message. Usually on 11 bits.
rtr : 0 : normal message
1 : Remote Transmit Request message. Cannot contain data
dn : data byte. A normal message can contain 0 to 8 bytes of data

The CAN message in CANOpen



The data are put in CAN frame in lsb first (ie little endian).
Example : the number 0x01020304 should be embeded like this :



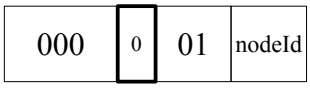
Fct Code (binary) :

EMCY	:	0001
PDO	:	0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010
SDOrx	:	1011
SDOtx	:	1100
NMT error control	:	1110
NMT	:	0000
SYNC	:	0001
TIME STAMP	:	0010

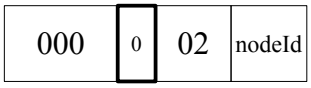
Note : In this document,
- All the numbers are in hexadecimal notation.
- If not specified, the CAN messages are « normals » (rtr = 0)

NMT protocol

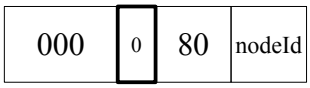
To put a node in operational mode



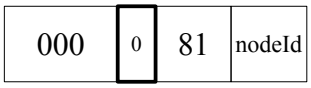
To put a node in stop mode



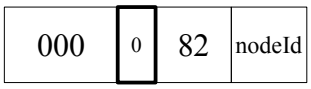
To put a node in pre-operational mode



To put a node in reset-application mode



To put a node in reset-communication mode



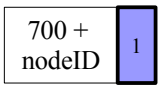
Note : To command all the nodes, use nodeId = 00

Examples

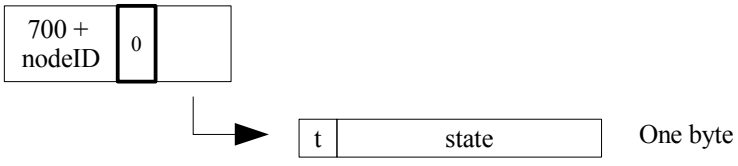
To put the node 0x6 in operational mode : 000 01 06
To put all the nodes in pre-operational mode : 000 80 00

Node guard protocol

To ask for a node its state, the master sends :



The node responds :



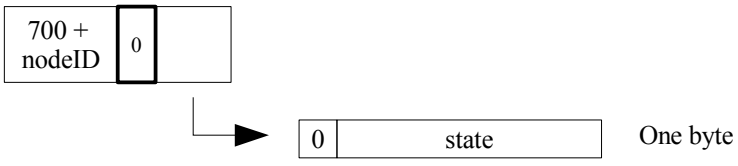
t : toggle 0,1,0,1....
state : 4 = stopped
5 = operational
7F = pre-operational

Examples

To ask for the node 5 its state, the master sends a CAN request (rtr = 1) : 705
The node 5 which is in stopped mode responds : 705 05
The master sends a NMT to put the node in pre-operational mode : 000 80 05
The master sends a new CAN request : 705
The node 5 which is now in pre-operational mode responds : 705 FF because the toggle = 1

Heartbeat protocol

The node transmit cyclically its state, without the need of a request frame :



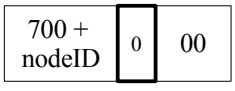
state : 4 = stopped
5 = operational
7F = pre-operational

Examples

The node 5, in pre-operational mode sends cyclically : 705 7F
Beware : There is no toggle bit.

Bootup protocol

When a node enters in pre-operational mode after being in initializing mode, it sends :



SDO protocol

SDO are used to read or write to the object dictionary of a node.
The node which request a read or a write is the client node.
The node in which the data is read or written is the server node.
Read = upload protocol
Write = download protocol
To read or write a data of 4 bytes or less, the simplest way is to use the
SDO upload/download expedited protocol.
All the SDO have the same CAN frame length : 8 bytes of data and rtr = 0.

SDO Download expedited protocol

To write the data 0xd0d1... in the server node object dictionary, the client node sends :

The client request :

Data length = 1 byte

600 + Serv NodeId	0	2F	Index	Sub index	d0	x	x	x

X : undefined. Put 0

The client request :

Data length = 2 bytes

600 + Serv NodeId	0	2B	Index	Sub index	d1	d0	x	x

X : undefined. Put 0

The client request :

Data length = 3 bytes

600 + Serv NodeId	0	27	Index	Sub index	d2	d1	d0	x

X : undefined. Put 0

The client request :

Data length = 4 bytes

600 + Serv NodeId	0	23	Index	Sub index	d3	d2	d1	d0

The server responds (if success) :

580 + Serv NodeId	0	60	Index	Sub index	00	00	00	00

The server responds (if failure) :

580 + Serv NodeId	0	80	Index	Sub index	SDO abort code error

Examples

To write the 1 byte data : 0xFD in the object dictionary of node 5, at index 0x1400, subindex 2, sends :
605 2F 00 14 02 FD 00 00 00
If success, the node 5 responds :
585 60 00 14 02 00 00 00 00

To write the 4 bytes data : 0x60120208 in the object dictionary of node 5, at index 0x1603, subindex 1, sends :
605 23 03 16 01 08 02 12 60
If success, the node 5 responds :
585 60 03 16 01 00 00 00 00

SDO Upload expedited protocol

To read the data 0xd0d1... in the server node object dictionary, the client node sends :

The client request :

600 + Serv NodeId	0	40	Index	Sub index	00	00	00	00

The server responds (if success) :

Data length = 1 byte

580 + Serv NodeId	0	4F	Index	Sub index	d1	x	x	x

X : undefined. Should be 0

The server responds (if success) :

Data length = 2 bytes

580 + Serv NodeId	0	4B	Index	Sub index	d1	d0	x	x

X : undefined. Should be 0

The server responds (if success) :

Data length = 3 bytes

580 + Serv NodeId	0	47	Index	Sub index	d2	d1	d0	x

X : undefined. Sould be 0

The server responds (if success) :

Data length = 4 bytes

580 + Serv NodeId	0	43	Index	Sub index	d3	d2	d1	d0

The server responds (if failure) :

580 + Serv NodeId	0	80	Index	Sub index	SDO abort code error

Examples

To read the 1 byte data : 0xFD in the object dictionary of node 5, at index 0x1400, subindex 2, sends :
605 40 00 14 02 00 00 00 00
If success, the node 5 responds :
585 4F 00 14 02 FD 00 00 00

To read the 4 bytes data : 0x60120208 in the object dictionary of node 5, at index 0x1603, subindex 1, sends :
605 40 03 16 01 00 00 00 00
If success, the node 5 responds :
585 43 03 16 01 08 02 12 60

SDO abort protocol

Abort code (hexa)	
0503 0000	Toggle bit not alternated
0504 0000	SDO protocol timed out
0504 0001	Client/server command specifier not valid or unknown
0504 0002	Invalid block size (block mode only)
0504 0003	Invalid sequence number (block mode only)
0504 0004	CRC error (block mode only)
0504 0005	Out of memory
0601 0000	Unsupported access to an object
0601 0001	Attempt to read a write only object
0601 0002	Attempt to write a read only object
0602 0000	Object does not exist in the object dictionary
0604 0041	Object cannot be mapped to the PDO
0604 0042	The number and length of the objects to be mapped whould exeed PDO length
0604 0043	General parameter incompatibility reason
0604 0047	General internal incompatibility in the device
0606 0000	Access failed due to a hardware error
0607 0010	Data type does not match, length of service parameter does not match
0607 0012	Data type does not match, length of service parameter too hight
0607 0013	Data type does not match, length of service parameter too low
0609 0011	Sub-index does not exist.
0609 0030	Value range of parameter exceeded (only for write access)
0609 0031	Value of parameter written too hight
0609 0032	Value of parameter written too low
0609 0036	Maximum value is less than minimum value
0800 0000	General error
0800 0020	Data cannot be transferred or stored to the application
0800 0021	Data cannot be transferred or stored to the application because of local control
0800 0022	Data cannot be transferred or stored to the application because ofthe present device state
0800 0023	Object dictionary dynamic generation fails or no object dictionary is present.

How to configure a PDO Transmit ?

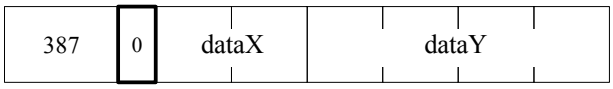
Example :

- Configuring the PDO 0x1800 + n
- Its cobId must be 0x387 (why not ?)
- The PDO is transmitted on synchro
- It must contains the data : data X (2 bytes) and data Y (4 bytes) in this order
- dataX is defined at index 0x6000 subindex 03
- dataY is defined at index 0x2010 subindex 21

1 – Index 1800 + n, subindex 01 : write the cobId (4 bytes)
2 – subindex 02 : write the transmission type « t » (1 byte)
t = 1 to 0xF0 : The PDO is transmitted every reception of « t » SYNC
t = FD : Transmission after reception of a request PDO (rtr = 1)
t = FF : Transmission on event. The nodes sends the PDO spontaneously
3 – Index 1A00 + n : define the mapping of the nth data.
Subindex 0 : write the number of data embeded in the PDO (1 byte). For this example, write « 2 »
Subindex 1 : define where to find the first data embeded and the size. (8 bytes)
The format is : index (2 bytes) – subindex (1 byte) – size in bits (1 byte)
For this example, write « 60000308 »
Subindex 2 : define where to find the second data embeded an the size (8 bytes)
For this example, write « 20102120

To configure the PDO 1802 of the node 5 to be transmitted every 3 synchro,
the SDO(s) to send should be (I have not tested) :
605 23 02 18 01 00 00 07 38
605 2F 02 18 02 03 00 00 00
605 2F 02 1A 00 02 00 00 00
605 23 02 1A 01 08 03 00 60
605 23 02 1A 02 20 21 10 10

The PDO :



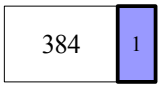
Note : a PDO may contains less than 8 bytes of data.

What is a PDO Transmited « on request » ?

It is a PDO which must be transmited when the node is receiving a remote transmit request (rtr) with the same COBID than the PDO.

Example :

PDO request :



If the PDO with the COBID 384 is « on request », it must be transmited.

How to configure a PDO Receive ?

Example :
- Configuring the PDO 0x1400 + n
- Its cobId must be 0x183 (why not ?)
- The PDO is transmitted on synchro
- It must contains the data : data X (2 bytes) and data Y (4 bytes) in this order
- dataX is defined at index 0x6000 subindex 03
- dataY is defined at index 0x2010 subindex 21

1 – Index 1400 + n, subindex 01 : write the cobId (4 bytes)
2 – Index 1400 + n, subindex 02 : write the transmission type « t » (1 byte)
t = 1 to 0xF0 : The PDO is transmitted every reception of « t » SYNC
t = FD : Transmission after reception of a request PDO (rtr = 1)
t = FF : Transmission on event. The nodes sends the PDO spontaneously
3 – Index 1600 + n : define the mapping.
Subindex 0 : write the number of data embeded in the PDO (1 byte). For this example, write « 2 »
Subindex 1 : define where to find the first data embeded and the size. (8 bytes)
The format is : index (2 bytes) – subindex (1 byte) – size in bits (1 byte)
For this example, write « 60000308 »
Subindex 2 : define where to find the second data embeded an the size (8 bytes)
For this example, write « 20102120

To configure the PDO 1402 of the node 5 to be received every 3 syncchro, the SDO(s) to send should be (I have not tested) :
605 23 02 14 01 00 00 07 38
605 2F 02 14 02 03 00 00 00
605 2F 02 16 00 02 00 00 00
605 23 02 16 01 08 03 00 60
605 23 02 16 02 20 21 10 10

The PDO :

Question : Why do we need to configure a « transmission type » for a PDO receive ?
Answer : I don't know

Important note concerning the PDO receive and transmit :
- A PDO may contains less than 8 bytes of data.
- You are not obliged to make the cobId with the node id. Put the value you prefer with this restriction : the 4 msb (bit 10 to 7) – the function code - which identifies the messages as a PDO must be choosen in this list (binary) :
0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010. You have a free choice for the bits 6 to 0, the number 0 excluded
-> **The CobId allowed for a PDO are (hexa) : 181 to 57F excluding 200, 280, 300, 380, 400, 480, 500**

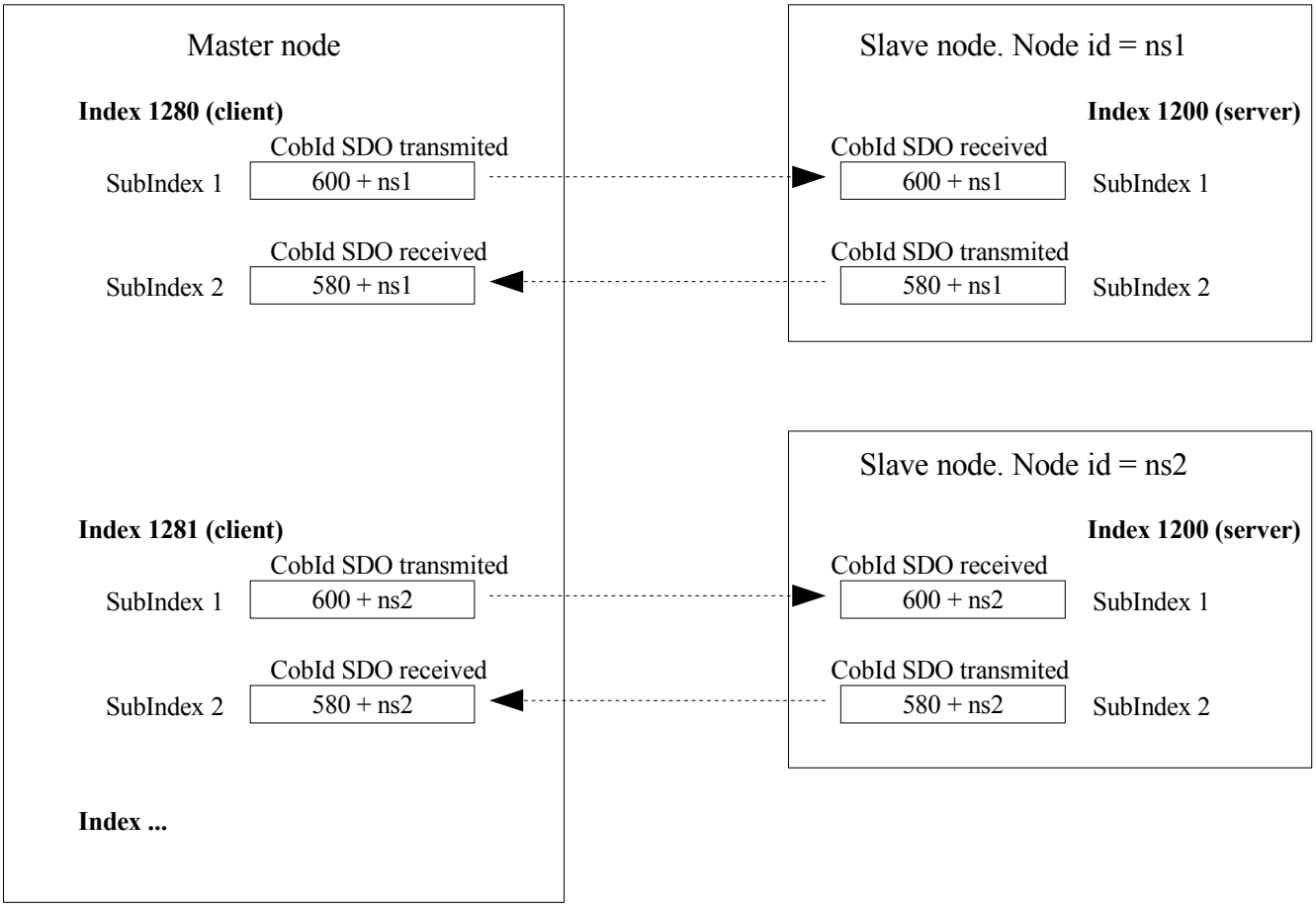
How to configure a SDO client ?

In most of the CANOpen networks, each slave node implements only one SDO server (index 1200), to be able to receive SDO from a client node. By default, the slave node SDO functionality is well configured.
Usually, a slave node does not implement any SDO client, because it usually not need to send SDO to other nodes.

The case of the Master node is different if it must configure the slaves nodes.
To send SDO to slaves nodes, it should have several SDO clients (one for each slave).
This entries must be configured at index 1280, 1281, ...

Configuring the SDO client defined at index 0x1280 + n to communicate with the node nsl (slave node)

1 – Index 1280 + n, subindex 01 : write the cobId transmit (4 bytes) : 600 + nsl
2 – subindex 02 : write the cobId receive (4 bytes) : 580 + nsl
3 – subindex 03 : write the slave node id (1 byte) : nsl (Optional)



How to configure a node to send the SYNC ?

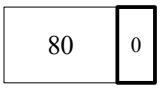
One node can send cyclically the SYNC signal. This may be done by the master or by a slave node.

1 – Index 1006, subindex 00 : write the period in microseconds. (4 bytes) :

Example :
 to send a SYNC every 10(dec) ms (1000 micro), write the value : 0x 2710
The SDO should be if it is the node 9: 609 23 06 10 00 10 27 00 00
To stop, write 0.
To start, write 0x40000080 at index 0x1005, subindex 0

Note that the node must be in operational mode to send the SYNC

What is the SYNC message ?



The SYNC is used only for the PDO.

How to configure a node to send its heartbeat ?

To send its heartbeat every n milliseconds :
1 – Index 1017 – subindex 00 : write « n ». (2 bytes) :

Example :
to send a heartbeat every 100 ms, write the value : 0x 64
The SDO should be, if it is the node 9 : 609 2B 17 10 00 64 00 00 00
To stop, write 0.

How to configure a node to monitor the heartbeats of several nodes ?

Example :
The node is expecting 2 hearteats from the nodes :
« nodeid1 », at least every n1 milliseconds
« nodeid2 », atleast every n2 milliseconds

1 – Index 1016, subindex 00 : Number of heartbeats monitored. (4 bytes). For this ex, the value is « 2 ».
2 - subindex 01 (4 bytes) : values for node 1 : 00 – nodeid1 (1 byte) – n1 (2 bytes)
3 - subindex 02 (4 bytes) : values for node 2 : 00 – nodeid2 (1 byte) – n2 (2 bytes)

Example :
To configure the node 9 to expect heartbeats from
node 2 : at least every 100(dec) ms = 0x64
node 3 : at least every 400(dec) ms = 0x190
node 7 : at leats every 4000(dec) ms = 0xFA0
The SDO should be:
609 2B 16 10 00 03 00 00 00 // 3 entries
609 2B 16 10 01 64 00 02 00 // The data to write is 0x00020064
609 2B 16 10 02 90 01 03 00 // The data to write is 0x00030190
609 2B 16 10 03 A0 0F 07 00 // The data to write is 0x00070FA0

To stop, write 0. at subindex 01 , ...,

What is this message ?

CobId (hex)	Protocol
0	NMT
80	SYNC
81 - FF	EMERGENCY
100	TIME STAMP
181 – 57F	PDO*
581 – 5FF	SDO response (server -> client)
601 – 67F	SDO request (client->server)
701 – 77F	NMT error control

* excluded : 200, 280, 300, 380, 400, 480, 500