see and get more…

GFCAN - Driver of CANBUS Protocol for CanCard
User's Manual

Doc. No. ENP4025
Version: 29-08-2005

1. GFCAN - Driver of CANBUS Protocol for CanCard

1.1. Driver Use

The GFCAN driver is used for data exchange between devices with the CAN network interface and an **asix** system computer provided with a CAN network communication processor card manufactured by Garz & Fricke Industrieautomation GmbH and provided with the Garz & Fricke‘s CAN driver for Windows NT" version 1.0.

1.2. Declaration of Transmission Channel

The full syntax of declaration of transmission channel operating according to the GFCAN protocol is given below:

logical_name=GFCAN

The GFCAN driver is loaded as a DLL automatically.

1.3. Addressing the Process Variables

Values of process variables are transferred in telegrams sent by controllers connected to the CAN network. Each telegram consists maximally of 8 bytes, which may be identified as:

bytes with indexes 1 – 8	(type BY),
16-bit numbers with indexes 1 – 4	(type WD),
32-bit numbers with indexes 1 – 2	(type DW),
32-bit floating-point numbers with indexes 1 – 2	(type FP).

The GFCAN driver differs the following access types to process variables:

read only	(type R_),
write only	(type W_),
write/read	(type RW_).

The addressing the process variables consists in the indication of:

- access type (R_, W_ or RW_);
- variable type (BY, WD, DW, FP);
- telegram no. (for variables with RW_ access type it is the telegram number that is used to read the variable);
- index within the telegram (for variables with RW_ access type it is the index in the telegram that is used to read the variable);
- for variables with RW_ access type it is necessary to declare additionally:
  - a/ telegram number that is used to read the variable,
  - b/ index in the telegram that is used to write the variable.

The syntax of symbolic address which is used for variables belonging to the GFCAN driver channel is as follows:

<access_type><variable_type><tel>.<index>[.<tel>.<index>]

where:

access type - access type to a process variable:
- R_ - only reading,
- W_ - only writing,
- RW_ - reading and writing,

variable_type - process variable type:
- BY - variable of the byte type,
- WB - variable of the 16-bit number type,
- DW - variable of the 32-bit number type,
- FP - variable of the 32-bit floating-point number type.

tel - telegram number,

index - index within the telegram.

EXAMPLE

X1, bytes 1-4 of telegram 31,	R_FP31.2,	NONE, 1, 1, NOTHING_FP
X2, word no. 3 of telegram 31,	R_WD31.3,	NONE, 1, 1, NOTHING
X3, state of burners,	RW_BY31.1.35.3,	NONE, 1, 1, NOTHING_BYTE
X4, valve setting,	RW_WD32.1.34.1,	NONE, 1, 1, NOTHING

The variable X1 is a variable of the 32-bit floating-point number type transferred to the **asix** system in bytes 1,2,3 and 4 of the telegram no. 31.

The variable X2 is a variable of the 16-bit number type, the value of which is transferred to the **asix** system by bytes 5 and 6 (third word) of telegram no.31. The value of the variable can't be modified by the application (variable only to read).

Value of the variable X3 is transferred to the **asix** system by means of the byte no. 1 of the telegram no. 31. The value exchange of the variable X3 consists in sending from the **asix** system the telegram no. 35, the byte no. 3 of which includes the required value of the variable X3.

1.4. Driver Configuration

The driver of GFCAN protocol may be configured by using the **[GFCAN]** section, placed in the application INI file. Individual parameters are transferred in separate items of the section. Each item has the following syntax:

item_name=[number [,number]] [YES] [NO]

☑ ***TRANSMISSION_SPEED=baud_id***

Meaning - the item is used to declare a transmission speed in the CAN network.

Default value - by default, the item is set to 500 kB.

Parameter:

baud_id - identifier of transmission speed in the CAN network:
- 1000 - 1 MB
- 800 - 800 kB
- 500 - 500 kB

250	-	250 kB
125	-	125 kB
100	-	100 kB
50	-	50 kB
20	-	20 kB
10	-	10 kB

EXAMPLE

An exemplary declaration of the transmission speed of 125 kB:

TRANSMISSION_SPEED=125

NETWORK_CONTROL=number

Meaning - the item allows to test the reception of telegrams from the CAN network. It defines the maximal time between receptions of successive telegrams with the same number. In case of exceeding this time, the process variables bound with such telegram will be provided with an error status. If additionally in the same time any telegram was not received from the CAN network, a message about a lack of telegrams in network is generated in '*Control Panel*'.

Default value - by default, the GFCAN driver does not check reception of telegrams.

Parameter:

number - maximal number of seconds, which may pass between successive telegrams with the same number.

EXAMPLE

An exemplary declaration of checking reception of telegrams every 5 seconds:

NETWORK_CONTROL=5

TELEGRAM_TRACE=YES|NO

Meaning - the item controls transferring to the operator panel the messages about telegrams that have been received from the CAN network. A message includes the number of telegram, number of bytes and content of individual telegrams in hexadecimal form.

Default value - by default, the contents of telegrams are not displayed.

EXAMPLE

An exemplary declaration of tracing the content of received telegrams:

TELEGRAM_TRACE=YES

CONTROL_TRACE=YES|NO

Meaning - the item controls transferring to the operator panel the messages about control telegrams that have been sent from the **asix** system computer to controllers. A message includes the number of control telegram, number of bytes and telegram contents in hexadecimal form.

Default value - by default, the contents of telegrams are not displayed.

EXAMPLE

An example of tracing the control telegrams:

CONTROL_TRACE=YES

LOG_FILE=file_name

Meaning - the item allows to define a file, to which all diagnostic messages of GFCAN driver and information about content of telegrams received and sent by the GFCAN driver will be written. If the item does not define a full path, the log file will be created in the current directory. The log file should be used only while the **asix** system start-up.

Default value - by default, the contents of telegrams are not displayed.

MAX_MOTOROLA_TEL=number

Meaning - the item allows to specify a maximal number of telegram, the content of which will be converted according to MOTOROLA format. All the telegrams with numbers, which are bigger than declared by means of the item MAX_MOTOROLA_TEL, will be converted according to INTEL format.

Default value - by default it is assumed that all telegrams are converted according to INTEL format.

EXAMPLE

An example of declaration, as a result of which the telegrams with numbers up to 150 inclusive are converted according to MOTOROLA format:

MAX_MOTOROLA_TEL=150