

Communication in DB-Net Network

Abstract

Data transfer on serial lines using the DB-Net protocol.

Author: Zbyněk Říha
Document: ap0009_en_02.pdf

Attachment

File content: ap0009_en_01.zip

dbnet_p1_en_01.dso	A sample application – active control system
dbnet_p2_en_01.dso	A sample application – passive control system
dbnet_p3_en_01.xls	Calculating the communication time in DB-Net

Contents

	Revision history	3
	Related documents	3
1	Definitions of Terms.....	4
2	DB-Net Protocol	5
3	Control System HW Parametrization	6
4	Control system SW Parametrization	7
4.1	SW Modules for Reading/Writing Data in the DB-Net Network	7
4.1.1	Serial Communication Statuses.....	8
5	Recommended DB-Net Implementation Procedure	9
6	Sample Communication in DB-Net Network.....	10
6.1	Application for control system 4.....	10
6.2	Application for control system 6.....	10
7	APPENDIX A	12
7.1	Troubleshooting in Communication in DB-Net Network	12
8	Technical Support.....	13
9	Warning.....	14

Revision history

Version	Date	Changes
001	07. 11. 2007	New document.
002	13. 04. 2010	Chapter 4.1 modified. Chapter 4.1.4. changed to 4.1.1 and modified.

Related documents

1. Help section for DetStudio development environment
2. Application Note for AP0016 – Operating guidelines RS485
File: ap0016_cz_xx.pdf

1 Definitions of Terms

Station Number

Number (0 to 31) of a control system or PC within the DB-Net network.

Active Station

An active station is a control system or a PC in the DB-Net network that is able to actively initiate requests for communication with other control systems or PCs.

Passive Station

A passive station is a control system or PC in the DB-Net network that only responds to requests from other stations.

DetStudio

A development environment by the company AMiT serving for control systems parametrization. This environment is freely accessible at www.amitautomation.com.

WID

A numeric identifier of each variable. This number is requested upon access to remote variables and it should be unique in the entire application (therefore also in each segment of a control systems' network).

Lifelist

Lifelist is a chart in which active stations maintain an overview of current statuses of all stations in the network (i.e. of stations 0-13).

2 DB-Net Protocol

DB-Net as an asynchronous communication protocol implemented in the line RS485 (connecting multiple stations into a network) or in the line RS232 (point-to-point type connection). The topology applied stems from the definition of line RS485, therefore, it is a bus topology type. In this layout, the line RS232 is considered only a special case of network formed by two stations.

Data transfer is performed using the REQUEST-ANSWER method. Therefore, stations authorized to transmit send out a frame with a request and wait for the counter-station's response. Any station that currently has no token (see further below) may receive a request at any time that it must process and send out a response immediately.

The access of stations to the bus is controlled by the method of token transfer in a logical cycle (TOKEN-RING); the protocol is so called multimaster-multislave. The station having the token may begin to transmit a request for data transfer – it becomes a master (controlling the entire network). After processing its communication requests, the station passes the token on – it becomes a slave (only responds to requests).

Addressing in the Network

Each station has a number ranging from 0 to 31. Unambiguous numbering of stations connected must be secured in one network of stations (each station within a single network must have a unique number).

Communication speed

Communication speed is optional from 9600 bps to 115200 bps.

Each control system by the company AMiT is able to communicate immediately using the DB-Net Protocol after the operation system (NOS) has been installed in it. However, it is necessary to set the communication parameters.

DB-Net Protocol Options

The DB-Net Protocol may serve to write/read data in control systems, download operational journals of control systems or introduce application programmes into control systems.

3 Control System HW Parametrization

Communication parameters are set in most control systems by means of switches marked as SW.

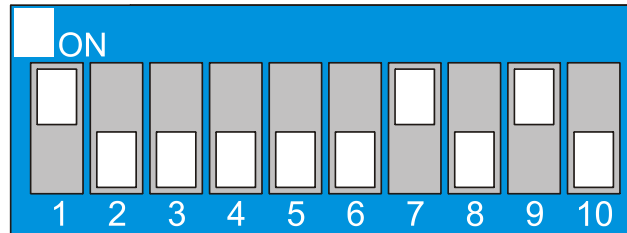


Fig. 1 – Setting communication parameters in a control system

Meaning of SW Switches

Switch	Meaning
1 to 5	Address in DB-Net Network
6 and 7	Communication speed
8	Communication interface for DB-Net (RS232/RS485)
9	Enable/disable application
10	User-configurable switch

Significance of individual address switches as well as communication speed setting options are stated in the following chart:

Switch Values

DIP1	Value 1
DIP2	Value 2
DIP3	Value 4
DIP4	Value 8
DIP5	Value 16

DIP6	DIP7	Communication speed
OFF	OFF	9600 bps
ON	OFF	19200 bps
OFF	ON	38400 bps
ON	ON	57600 / 115200 bps

Attention

If the switches No. 6 and 7 are in the ON position, control systems marked "S", or also "W" achieve communication speed up to 115200 bps. In control systems without this mark, communication speed is 57600 bps.


The picture above shows the set station number 13, communication speed 38400 bps, and the control system communicates by means of the DB-Net Protocol on the line RS485.

Note:

All changes in individual switch positions manifest only after the control system restart (unplugging and replugging the power supply).

Note:

The DB-Net communication protocol allows the control system to communicate always on one serial interface only, i.e. either by means of the line RS232 or the line RS485. The DB-Net communication protocol does not allow the control system to communicate on multiple serial interfaces at once.

In control systems not including SW switches, communication parameters are set using the display. The configuration menu is usually accessible after the control system is unplugged from the power supply, the button  on the control system pressed and held while the control system is replugged into the power supply.

4 Control system SW Parametrization

In communication within the DB-Net network, we distinguish two basic types of stations connected to the DB-Net network; they are *active station* and *passive station*.

Passive Station

Passive station only listens to the operations on the network. If it recognizes a request framework for itself, it responds immediately.

Active Station

In case the control system is an active station, it will act as a Master in the network and maintain a so called Lifelist of other control systems connected into the DB-Net network. If a PC is also connected into the DB-Net network, it acts as an active station in most cases (actively requesting variable values from other control systems).

The setting whether the control system acts as an active or passive station is performed already during the actual SW design in the appropriate parametrization environment (DetStudio) and this choice cannot be activated or deactivated once the SW has been installed into the control system; it would require reload of the SW in the control system.

The option may be activated or deactivated in the menu *Project/Settings* of the DetStudio parametrization environment. After you open the window *Project Settings* you may select Active/Passive station in the item *Miscellaneous*.

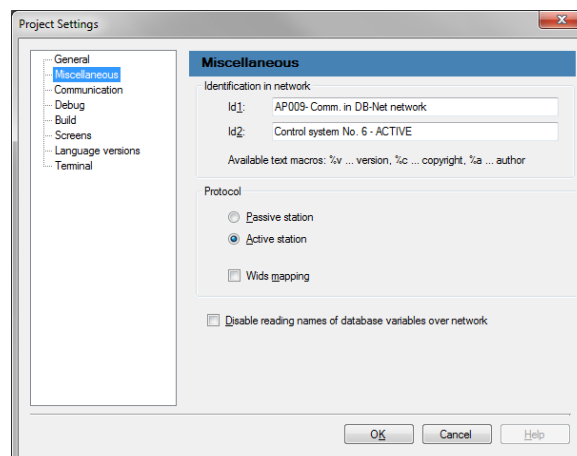


Fig. 2 – Selecting active/passive station

If the station itself does not need to write or read data from/to another station, it is recommended it is a passive station. Active stations, even when they are not transmitting data, generally reduce the width of the band for data transfer.

4.1 SW Modules for Reading/Writing Data in the DB-Net Network

Should control systems communicate with each other without any PC interference, it is necessary to program this communication using the DetStudio development environment. Reading/writing data using the Db-Net Protocol is performed by means of SW modules *ReqXXX*. Individual module descriptions are available in the Help section of DetStudio development environment.

- ◆ *ReqDb* – request for database variable transfer by means of the DB-Net Protocol.
- ◆ *ReqDbDr* – request for database variable transfer by means of the DB-Net Protocol. It differs from the *ReqDb* module by the fact that its parameters can be entered using variables.
- ◆ *ReqTime* – time transfer on the network.

Note:

In case the control system does not actively initiate communication in the DB-Net network (but only responds to potential queries in the network) it is not necessary to program the control system any further. In such a control system, it suffices to set the switches marked SW properly, or set communication parameters in the service menu on the control system terminal properly.

4.1.1 Serial Communication Statuses

Variables substituted for parameters Vložení and Stav of DB-Net communication modules obtain various values depending on the current communication status. We recommend using these parameters especially when adjusting the communication, when you are able to get information on possible communication error on the basis of their values.

Entering a request for serial transfer

Value	Meaning
1	Request issued – no error.
2	Request not issued because there was an identical request earlier that is still pending.
4	Request not issued because the request buffer is full.
8	Request not issued because: 1) this station is the owner of the database variable requested or 2) the transfer of database matrix VÝŘEZ is requested that exceeds the dimensions of the matrix or 3) the local variable has a different data type than the one defined by the communication module (only for modules for direct transfer of variables).

Request status

Bit	Meaning
0	Has value 1 if communication is taking place.
1	Has value 1 if last communication terminated was finished successfully.
2	Has value 1 if last communication terminated was finished with an error.
3	Has value 1 if the request was split into multiple frames and transfer of one of the frames ended in an error and other frames are still being communicated.
12 to 15	If the communication ended in an error (bit 0 has value 0 and bit 2 has value 1), these bits contain the communication error codes according to the following chapter. Values of these bits are not otherwise specified.

Error codes in bits 12 to 15

Value	Meaning
1	Transfer error (check sum, invalid frame control character,...).
2	Wrong WID, type or dimension of the remote station database variable. Error in database parametrization, variable definitions differ in this and remote station.
3	The remote station does not support the function requested.
4	Unknown error.
5	The remote station has not responded.
7	The remote station responded with an incorrect frame type.

Attention

The status variables serve to provide information on the communication status and result. The application should not write anything into variables substituted for parameters *RqState* and *State*. At the same time, it is necessary that each module *EthReqDb/EthRqDbDr* uses its own status variables. The variable substituted for the parameter *State* must not be shared by multiple modules.

5 Recommended DB-Net Implementation Procedure

In order for the entire network to work properly, it is necessary to set all required parameters correctly and connect the whole network properly.

Communication speed

The communication speed must be the same in all control systems. The speed value cannot be determined beforehand, but the general rule is the higher the communication speed the lower the potential connection length of RS485 (typically 1200 m at 19200 bps) and requirements for connection cables quality also increase.

If the control system communicates both by means of the DB-Net network and Ethernet (DB-Net/IP) at the same time, the communication speed for DB-Net should be set to 19200 bps or 38400 bps.

Communication Time

Calculation of the minimum period necessary for writing/reading the requested variables is available in attachment ap0009_en_01.zip. It is a file named dbnet_p3_en_01.xls created in MS Excel.

If this calculated period gets shortened, there is no guarantee for flawless communication.

Control System Addresses

Each control system must have a unique address set. The address, as well as the communication speed, is set by means of a SW switch (or a keyboard) in each control system.

Note:

In version NOS 3.50, when using multiple active stations in a single DB-Net network, it is necessary that numbers of these stations do not go in a row (1, 2, 3, ...); they need to have at least 1 point interval, i.e. 1, 3, 5, ...

Communication Lines

In order for the network to function properly, it is necessary to connect the individual control systems (stations) properly. The higher quality cabling and cables, the higher transfer reliability in serial lines. In an environment with heavy interference (typically when using frequency converters), the cabling quality is decisive for reliable functioning of the entire communication network. In an environment where interference may occur, we recommend you use overvoltage protection. In communication using the line RS485, we recommend you connect this line according to the Application Note AP0016 – Operating Guidelines RS485.

6 Sample Communication in DB-Net Network

Communication is laid out between two control systems numbered 4 and 6. The control system 6 reads/writes variables from/to the control system 4. Control system 6 is therefore active and must have project parameters set according to Chapter *SW parametrizace řídicího systému*.

6.1 Application for control system 4

In control system 4, there are variables defined that we would like to read as well as variables into which we would like to write. The control system itself will not initiate communication, therefore it is passive.

Variables defined in control system 4

Name	WID	Type	Station	Comment
Read	4000	L	4	Variable read by the active CS
Write	4001	F	4	Variable into which the active CS writes
ReadDir	4002	MF[2,3]	4	Variable read by the active CS – Dir
WriteDir	4003	I	4	Variable into which the active CS writes – Dir

The application with defined variables is included in the Attachment ap0009_en_01.zip. It is a sample project called dbnet_p2_en_01.dso created in the DetStudio development environment. This project is created for the control system **StartKit**. However, it can be changed for any control system equipped with a serial communication line, by means of a DetStudio menu “Tools/Change station”.

6.2 Application for control system 6

Before we actually enter the request for reading/writing, the program checks (based on bit no. 0 variables substituted for parameter Stav) whether an identical request has been entered previously and whether it is being communicated at the moment. If not, the request for reading/writing is entered.

Use of ReqDb

If the command `ReqDb` is used for communication, it is necessary to first define the copy of a remote variable with the same WID, station number and date type as the variable we write/read data to/from located in control system 4 in the control system that actively reads/writes data. Therefore, first we define two copies of remote variables in the control System 6:

Name	WID	Type	Station	Comment
Read	4000	L	4	Variable read by the active CS
Write	4001	F	4	Variable into which the active CS writes

Now, all we have to do is enter the following code into a periodic process with a suitable period:

```
//Read
If Read_Stat.0
//If the reading is in process, I do not initiate communication
Else
    ReqDb 0x0000, Read[0,0], 1, 1, Read_Put, Read_Stat
EndIf

//write
If Write_Stat.0
//If the writing is in process, I do not initiate communication
Else
    ReqDb 0x0001, Write[0,0], 1, 1, Write_Put, Write_Stat
EndIf
```

User of ReqDbDir

When using the command `ReqDbDir` it is not necessary to have a copy of the remote variable in the active control system. It is fully sufficient to define any variables of corresponding data types in the active control system. We will read the Float matrix using the command `ReqDbDir` and write a variable type Integer. We define the two following variables:

Name	WID	Type	Station	Comment
Dir_Read	6000	MF[2,3]	6	Variable read by the active CS – Dir
Dir_Write	6001	I	6	Variable into which the active CS writes – Dir

Now, all we have to do is enter the following code into a periodic process with a suitable period:

```
//Read
If Read_StatDir.0
//If the reading is in process, I do not initiate communication
Else
    ReqDbDir 0x0000, 4, NONE, 4002, NONE, 5, NONE, 0, NONE, 0, NONE, 2, NONE,
    3, NONE, Dir_Read, 0, NONE, 0, NONE, Read_PutDir, Read_StatDir
EndIf
//Write
If Writ_StatDir.0
//If the reading is in process, I do not initiate communication
Else
    ReqDbDir 0x0001, 4, NONE, 4003, NONE, 0, NONE, 0, NONE, 0, NONE, 1, NONE,
    1, NONE, Dir_Write, 0, NONE, 0, NONE, Writ_PutDir, Writ_StatDir
EndIf
```

The application using `ReqDb i ReqDbDir` is included in Attachment `ap0009_en_01.zip`. It is a sample project called `dbnet_p1_en_01.dso` created in the DetStudio development environment. This project is created for the control system **StartKit**. However, it can be changed for any control system equipped with a serial communication line, by means of a DetStudio menu “Tools/Change station”.

7 APPENDIX A

7.1 Troubleshooting in Communication in DB-Net Network

If the communication in DB-Net Network does not function, perform check-up according to the following procedure:

- ◆ Check whether all cables are functional and connected properly.
- ◆ Check the speed setting in control systems between which you want to communicate. SW switches No. 6 and 7 must be set identically in all control systems.
- ◆ Check the selected communication interface RS232/RS485 is set correctly. If you communicate between two control systems using the interface RS232, you need to have the SW switch No. 8 in the OFF position in both control systems. If you communicate using the line RS485, the SW switch No. 8 needs to be in the ON position in all control systems.
- ◆ If you communicate by means of the **DM-232TO485** converter connected to the control system, both switches on the converter must be in the OFF position (RTS). If the converter is connected to PC, the switch No. 2 has to be in the ON position (Automatic) on the converter. Check whether the converter is connected to the control system or to PC correctly.
- ◆ Check whether control systems with modules `ReqDb/ReqDbDir` are active.
- ◆ Define two variables that you put into parameters `RqState` and `State` in module `ReqDB/ReqDbDir`. These two variables contain the code of request placement for communication and communication status on the basis of which you will be able to remove the error.

8 Technical Support

All information on communication by means of the Db-Net Protocol will be provided by the technical support department of the company AMiT. Do not hesitate to contact the technical support via e-mail using the following address: support@amit.cz.

9 Warning

The company AMiT, spol. s r. o. does not provide any guarantee concerning the contents of this publication and reserves the right to change the document with no obligation to inform any person or authority.

This document can be copied and redistributed under following conditions:

1. The whole text (all pages) must be copied without making any modifications.
2. All copies must include the AMiT, spol. s r. o. copyright and any other notices stated in the document.
3. This document must not be distributed for profit.

The names of products and companies used herein may be trademarks or registered trademarks of their respective owners.