miniMon

Simple CAN Monitoring Tool for Windows

Software Version 2.x

IXXAT

The expert for industrial and automotive communication
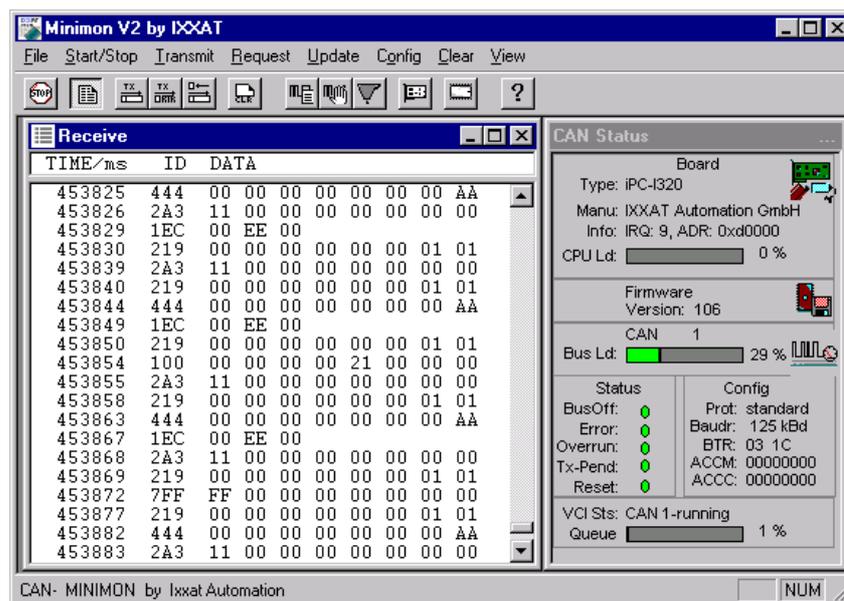
IXXAT

Headquarter
IXXAT Automation GmbH
Leibnizstr. 15
D-88250 Weingarten

Tel.: +49 (0)7 51 / 5 61 46-0
Fax: +49 (0)7 51 / 5 61 46-29
Internet: www.ixxat.de
e-Mail: info@ixxat.de

US Sales Office
IXXAT Inc.
120 Bedford Center Road
USA-Bedford, NH 03110

Phone: +1-603-471-0800
Fax: +1-603-471-0880
Internet: www.ixxat.com
e-Mail: sales@ixxat.com

Support

In case of unsolvable problems with this product or other IXXAT products please contact IXXAT in written form by:

Fax: +49 (0)7 51 / 5 61 46-29
e-Mail: support@ixxat.de

For customers from the USA/Canada
Fax: +1-603-471-0880
e-Mail: techsupport@ixxat.com

Copyright

miniMon-Manual-E-V3.2

1 Overview

The CAN miniMon is a simple and low-cost CAN bus monitor program which allows the online monitoring of bus traffic on the CAN bus as well as the transmission of single CAN objects. The CAN miniMon is delivered with VCI.
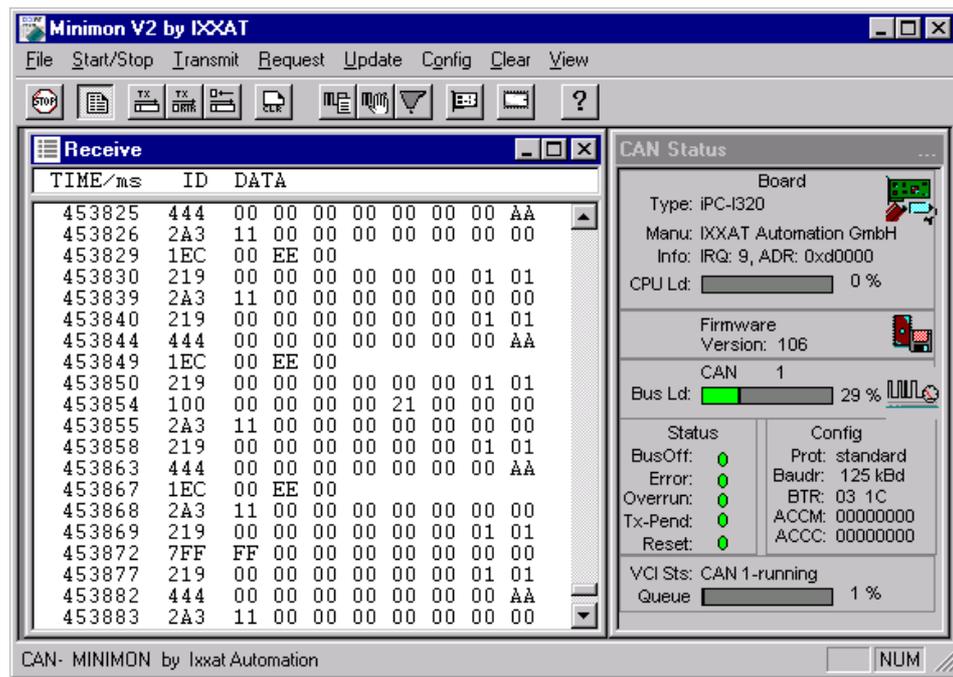
Picture 1-1: CAN miniMon

2 Functions and Handling

2.1 Starting the program

The CAN miniMon can be started in the Start menu folder of the VCI or by directly executing the file MINMON32.EXE.

Upon start the dialog box for selecting the type of CAN interface is opened automatically. In this dialog box you have to select a PC/CAN interface which is installed on your PC (see chapter 2.2.1).

2.2 Configuration

By selecting the item ´Config´ in the main menu, the sub-menu for the configuration of the CAN miniMon is opened.

2.2.1 Board Type (Hardware Configuration

Since CAN miniMon is able to run with different types of PC/CAN interfaces, the user has to specify which CAN interface the CAN miniMon has to use.

Picture 2-2: Board type configuration in CAN miniMon

In this dialog the parameter "Board" shows that CAN interface which was used in your last session. If you want to select a different CAN interface, press the button next to the parameter window. After that a window opens which shows all PC/CAN interfaces which are installed on the PC (corresponding to the applet "IXXAT Interfaces"). Here you can select the CAN interface the CAN miniMon shall work with and confirm your selection by pressing OK. It is not necessary to specifiy hardware specific parameters like interrupt number or memory address.

Furthermore it is necessary to specify which CAN controller of the PC/CAN interface (if the interface has two CAN controllers) and which CAN protocol shall be used.

Chip: The CAN interface can be provided with one or two CAN controllers (e.g. one SJA1000 and one 82527). Therefore the user has to specify which CAN controller should be used. The CAN controllers are referenced as CAN 1 and CAN 2. The information about the hardware configuration of the used interface can be taken

- from the ´IXXAT Interfaces´ applet in the Control Panel by executing the hardware test.

- from the ´CAN-Type´ dialog in the CAN miniMon menu ´View´

- from the manual of the interface.

- from your order.

Protocol: Selection which CAN protocol shall be used: standard protocol (11 bit identifier) or extended protocol (29 bit identifier).

For the selection of the CAN controller to be used the following has to be considered:

- The extended CAN protocol (29 bit identifier) and the detection of error frames is not supported by the 82C200 CAN controller.

- The reception of remote frames in the receive window is only possible using the SJA1000 CAN controller.

- When the 82527 CAN controller is used an exact display of the actual bus load is not possible.

2.2.2 Baudrate

The baud rate can either be selected from a list which provides a number of standard baud rates for selection (acc. to CiA) or can be set directly by entering the values for the bus timing registers.

- **Baudrate (list of values)**

The baud rate is selected and confirmed by pressing 'OK'.
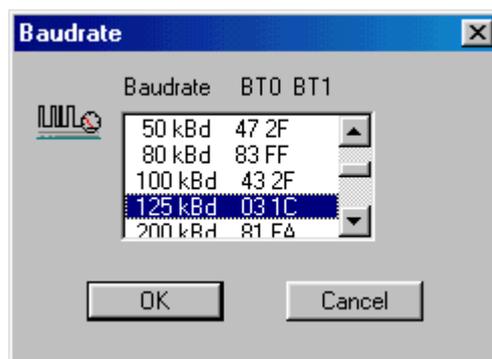
Picture 2-1: Baudrate configuration with a list of values

- **Baudrate (manual)**

The values for the bit timing registers of the CAN controller are entered directly and confirmed by pressing 'OK'. The values are stored when the CAN miniMon program is terminated and reloaded when the program is restarted.

- BTR0: CAN baudrate register 0
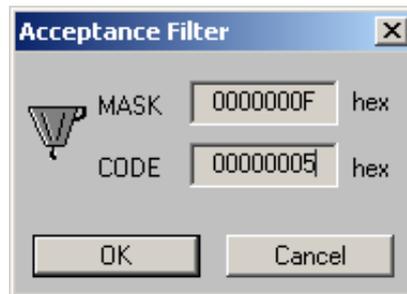
- BTR1: CAN baudrate register 1

Picture 2-2: Manual baudrate configuration

The values have to be calculated for a CAN controller SJA1000 or 82527, clocked with 16 MHz

2.2.3 Filtering of received objects (acceptance)

By selecting the item ´Config´|´Acceptance´ in the main menu a filter for the reception of CAN objects can be configured.

Picture 2-3: Acceptance mask

In the dialog box 'Acceptance' the values for the acceptance code and acceptance mask are set. The input is hexadecimal. The acceptance mask specifies the bits in the identifier that are relevant for the filter (1 = relevant; 0 = not relevant). The acceptance code specifies the value which the relevant bits must have to be allowed to pass the filter. The acceptance filter works independent of the used type of CAN controller. The input is hexadecimal. For standard CAN identifiers (11 bit identifier) only the last three hexadecimal digits are used.

Example:

You are only interested in CAN objects with value 5 in the lower 4 identifier bits.

Therefore only the lower 4 identifier bits are relevant for filtering:

    MASK = 000 0000 1111 = 0x00**F**

These relevant bits must have the value 5:

    CODE = 000 0000 1001 = 0x00**5**

With these settings CAN miniMon will only receive CAN objects with value 5 in the relevant identifier bits:

    0x00**5**, 0x01**5**, 0x02**5**, 0x03**5**, 0x04**5** ...

Filter algorithm:

A bitwise AND operation is performed between the identifier of a received CAN object and the adjusted acceptance mask. The result of the AND operation is compared with the adjusted acceptance code. Only CAN objects for which the result of the AND operation is equal to the acceptance code are allowed to pass the acceptance filter. All other CAN objects are not received.

If 0 is set for the acceptance mask there are no relevant bits and therefore the filter is completly open. All CAN objects will be received.

2.3 Start/Stop

The item ´Start/Stop´ in the main menu is a switch for starting and stopping all functions. The actual status is displayed in the status window in the window CAN status („RUN" or „RESET", see chapter 0).

When the operation is stopped, all queues , buffers and status information are deleted. The receive window contents are kept and the CAN controller is in reset.

2.4 Transmission of objects (Transmit)

By selecting the item ´Transmit´ a dialog box is opened which allows the input of a CAN object. The CAN object is specified by the identifier and the data which have to be transmitted. The input of the identifier and data is hexadecimal with spaces between identifier and each data byte. By pressing 'OK' the object is transmitted.

The following dialog show a CAN object with 11-bit identifier 0x123 and the data bytes 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0xEE, 0xFF.
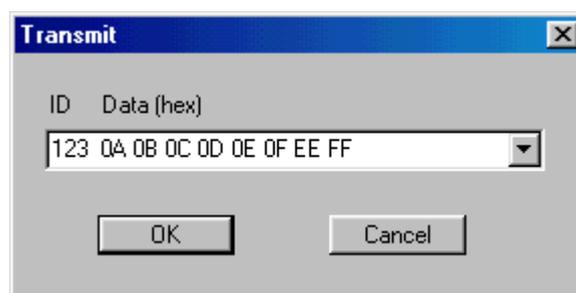
Picture 2-4: Message input by using 11 bit identifier

The following dialog show a CAN object with 29-bit identifier 0x12345678 and the data bytes 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x01, 0x02.
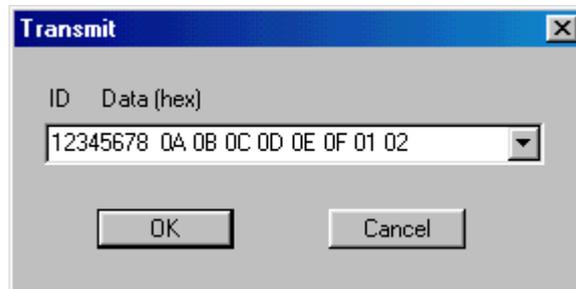
Picture 2-5: Message input by using 29 bit identifier

2.5 Request of objects (Request)

By selecting the item ´Request´ a dialog box is opened which allows the input of remote frame. A remote frame requests a CAN object with the same identifier from another bus node. The input of the identifier which shall be requested is hexadecimal. By pressing 'OK' the object is requested by the transmission of the remote frame.

2.6 Clear all windows

The item ´Clear´ deletes all contents of the receive queue and all contents of the receive window. The error frame counter are reset.

2.7 Reception of CAN objects

Received CAN objects are displayed in the receive window. Each object is displayed with time stamp (decimal, resolution 1 millisecond), identifier and data (hexadecimal).

This window shows a received data frame with identifier 0x100 and the data bytes 0xFF, 0x05, 0x03, 0x08. It was received 33730 msec after opening the board.
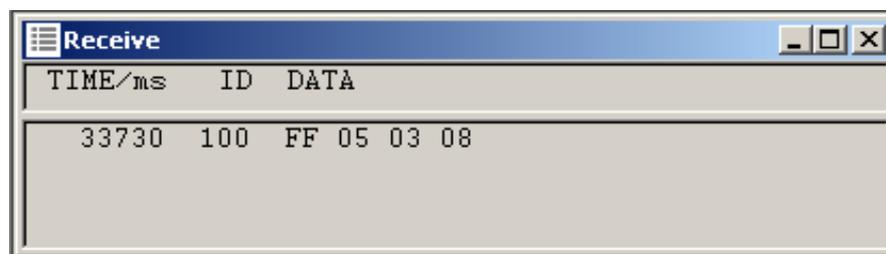

Bild 2-6: A received data frame

This window shows a received remote frame with identifier 0x32F and data length code (DLC) 1. Remote frames do not contain any data bytes, just a data length code. It was received 177962 msec after opening the board.
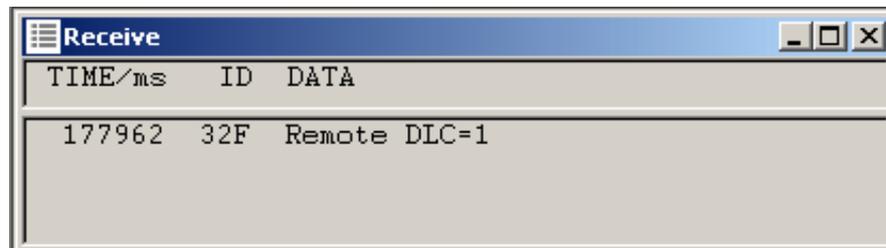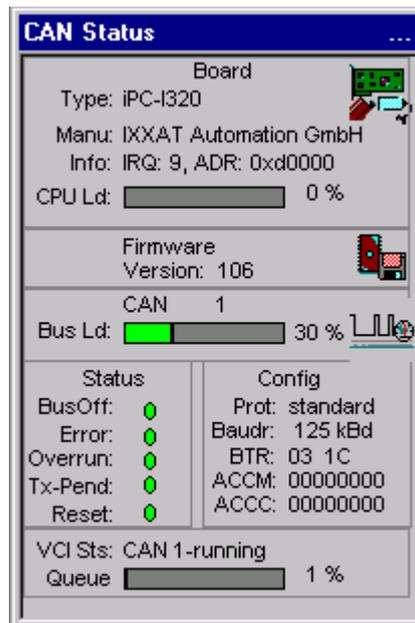

Bild 2-7: A received remote frame

2.8 Status Display

The windows on the right side displays the actual settings of the CAN controller and the most important information about the interface

Picture 2-8: Status display

Board:	Information about the current hardware settings
CPU Ld:	Current load of the CPU on the PC/CAN interface board
Bus Ld:	Current bus load in % and as a graphic beam
Section Config:	
Prot:	Selected protocol type: standard or extended
Baudr:	Selected baudrate
BTR:	Current values of Bus Time Registers 0 und 1
ACCM:	Current value of acceptance mask
ACCC:	Current value of acceptance code

Section Status:

CAN Status: Information about the status bits in the CAN controller (red = set, green = reset):

    - BusOFF: bus off status

    - Error: error status

    - Overrun: receive overrun

    - Tx-Pend: transmit pending

    - RUN / RESET: CAN is in initialization mode (RESET) or active (RUN)

VCI Sts: Current status of the software interface and the CAN interface.

Queueload: Information about the current load of the receive queue.