100BASE-T1 Media Converter and 100BASE-T1 USB Interface Communication Protocol Specification

For FW v. 1.5

MACH SYSTEMS s.r.o.

Changes

Date	Changes	Changed by
30.07.2026	Fix wrong response length and examples	PK
19.03.2023	Fix response length for WRITE_MASTER_SLAVE, WRITE_CAN_RX(/TX)ID, CAN_SEND_MESSAGE and PHY_TEST_MODE	PK
8.11.2022	Clarifications	PK
21.03.2022	Message examples	PK
30.11.2021	The possibility to use the device as a USB-CAN interface.	PK
9.7.2021	100BASE-T1 test modes added	PK
4.5.2020	Draft release	MM
16.10.2019	T1 Master/Slave overriding added	MM

Contents

1. Introduction 2
2. Communication Protocol 3
   - 2.1. USB - Virtual COM Port 3
   - 2.2. CAN bus 3
   - 2.3. Message Overview 4
   - 2.4. Error Codes 5
3. Message Specification 6
   - 3.1. Read 100BASE-T1 PHY register 6
   - 3.2. Read 100BASE-TX PHY register 6
   - 3.3. Read SQI 6
   - 3.4. Cable Test 7
   - 3.5. T1 Master/Slave Overriding 7
   - 3.6. T1 Test Mode 8
   - 3.7. CAN Configuration 9
   - 3.8. Read Status 9
   - 3.9. CAN Message Specification 10
4. Message Examples 21
   - 4.1. Device Setting and Info 21
   - 4.2. USB-CAN Interface 21
5. Contact 22

1. Introduction

Both **100BASE-T1 Media Converter** (p/n: 100BASET1-MC-ETH) and **100BASE-T1 USB Interface** (p/n: 100BASET1-USB-IF) products offer configuration and diagnostic over USB. Further, the device can be used as USB-CAN interface.

Both devices feature a micro USB connector and they act as a virtual serial port (USB Virtual COM port - VCP) when plugged in a computer. It is also possible to use the CAN bus to diagnose the device.

The following functions can be carried over the USB:

- Read status of the 100BASE-T1 port
- Carry out a cable test
- Enable test mode generation
- Access PHY registers
- Use the device as USB-CAN interface:
    - Configure CAN channel
    - Transmit and receive frames

This document describes the communication protocol that can be used to read device's status information and to control the 100BASE-T1 port. or to use the device as a USB-CAN interface.

If the device is not used as a USB-CAN interface, it is also possible to read device's status and to control the 100BASE-T1 port over the CAN bus.

2. Communication Protocol

The communication between the device and another system is based upon a binary protocol over the serial port. The same message structure is used for both directions - to and from the device.

2.1. USB - Virtual COM Port

The protocol consists of Start, Message ID, DataLength, Data bytes, Checksum, and End.
Serial port (VCP) configuration is **fixed**: 115200 Baud, 8 data bits, no parity, 1 stop bit

STX (1B)	ID (1B)	DATALEN (1B)	DATA (X B)	CHECKSUM (1B)	ETX (1B)
0x02	Message ID	Number of data bytes	Data bytes Number of bytes = DATALEN	1-byte sum of ID, DATALEN and all DATA bytes	0x03

The rest of the documentation refers to DATA part **only**. The user is then responsible for encapsulating it with the rest of the protocol fields (STX, ID, DataLen, Checksum and ETX).

2.2. CAN bus

If the device's CAN port is not used as a USB-CAN interface, it is possible to use the CAN bus for diagnostic purposes.
The device receives via CANID_RX and transmits over CANID_TX. Both CAN identifiers can be changed per device - see Message IDs 0x50 – 0x57.

Default configuration:

CANID_RX = 0x123 Std ID.
CANID_TX = 0x321 Std ID.
CAN Baud = 500 KBaud

CAN Frame

SOF	ID	RTR	IDE	r	DLC	Data Bytes 0 - 8		Chksum	DEL	ACK	DEL	EOF
SOF	ID	RTR	IDE	r	DLC	Message ID	Data	Chksum	DEL	ACK	DEL	EOF
1	11 bit	1	1	1	4	8 bit	0 – 56 bit (0 – 7 bytes)	15 bit	1	1	1	7

Data Byte 0 is always used as Message ID (just like in VCP), the rest of the data bytes carry the message content.

Note: Grey parts are automatically generated by a CAN controller.

2.3. Message Overview

The following table describes messages of the communication protocol. The CAN setting messages (id 0x50-0x57 and 0x60-0x68) are available over USB VCP only.

Message ID	Name	Request Data Length	Response Data Length	Comment
General				
0xFA	BOOT_UP		0	A notification that the device was powered up
0x5A	READ_SN	0	4	Read device's serial number
0x5B	READ_HW_INFO	0	6	Read device's HW info
0x5C	READ_SW_INFO	0	2	Read device's SW info
0x5D	READ_STATUS	0	4	Read status information (port link status etc.)
Ethernet				
0x01	READ_T1REG	1	2	Request data byte = register address
0x0A	READ_T100REG	1	2	Request data byte = register address *Note: Not available on 100BASE-T1 USB Interface.*
0x10	READ_SQI	0	1	T1: Read signal quality
0x11	DO_CABLE_TEST	0	1	T1: cable test
0x12	WRITE_MASTER_SLAVE	1	0B ACK	T1: Override Master/Slave
0x13	PHY_TEST_MODE	1	0B ACK	T1: test modes
CAN Bus – Configuration for Diagnostic				
0x50	READ_CAN_CONFIG	0	5	Same structure as: 0x62 CAN_READ_CONFIGURATION
0x51	WRITE_CAN_CONFIG	5	0B ACK	Configure CAN channel for conf. msg.
0x54	READ_CAN_RXID	0	4	Read CAN RX ID for conf. msg.
0x55	WRITE_CAN_RXID	4	0B ACK	Write CAN RX ID for conf. msg.
0x56	READ_CAN_TXID	0	4	Read CAN TX ID for conf. msg.
0x57	WRITE_CAN_TXID	4	0B ACK	Write CAN TX ID for conf. msg.
CAN Bus – USB-CAN Interface				
0x60	CAN_CONFIGURATION	4	0B ACK	Configure CAN channel
0x61	CAN_WRITE_CONFIG_TIM	6	0B ACK	Configure CAN channel set time quanta
0x62	CAN_READ_CONFIGURATION	1	7	Read CAN channel configuration
0x63	CAN_SAVE_CONFIGURATION	1	0B ACK	Save current CAN configuration to EEPROM
0x64	CAN_LOAD_CONFIGURATION	1	0B ACK	Load CAN configuration from EEPROM
0x65	CAN_DEFAULT_CONFIGURATION	1	0B ACK	Load CAN default configuration
0x66	CAN_ECHO_CONF	2	0B ACK	Enable/Disable Tx echo
0x67	CAN_START_CHANNEL	1	0B ACK	Start CAN channel
0x68	CAN_STOP_CHANNEL	1	0B ACK	Stop CAN channel
0x70	CAN_SEND_MESSAGE	5 to 15	0B ACK	Transmit CAN message

0x71	CAN_RECEIVED_FRAME	- (no request needed)	5 to 70	Received CAN message
System / Error				
0xFE	RESTART_BOOT	0		Restart gateway to bootloader
0xFF	GENERAL_ERROR		1	An error occurred, see 2.4

2.4. Error Codes

The following tables describes error codes

Error Code	Comment
0xA2	Unknown Message ID
0xA3	Too large or incorrect data length
0xC0	CAN error frame
0xF0	Configuration error
0xF1	Channel running, channel should be stopped before changing its configuration

2.5. General Response

Device responds with a message acknowledgment after receiving a valid message. The acknowledgement has the same ID as the original message. Acknowledgement is either without any dat. See Response Data Length in Message Overview for information about acknowledge length. If there is some problem, the response is Error Response.

Response:
No data when bus channel number is not relevant.

2.6. Error Response

Message ID = 0xFF

Device responds with an error if the command could not be processed correctly. Message ID contains ID of the requesting message.
Response:

DATA 0	DATA 1
Error code	Message ID

When Channel number is not relevant.

3. Message Specification

3.1. Read 100BASE-T1 PHY register

Message ID=0x01

Request: Register Address (1 byte)
Response: Register Value (2 bytes)

Register	Start Bit	Bit Length	Name	Description
1	2	1	Link Status	1 = link is up
18	15	1	Master/Slave	1 = PHY is Master
25	6	1	Polarity Detect	1 = polarity inversion detected on T1 port
26	0	8	Local Receiver Link-fail Counter	The counter is incremented when local receiver is NOT_OK; when the counter overflows, the value of FFh is retained. The counter is reset when the register is read.
26	8	8	Remote Receiver Link-fail Counter	The counter is incremented when remote receiver is NOT_OK; when the counter overflows, the value FFh is retained. The counter is reset when the register is read.

3.2. Read 100BASE-TX PHY register

Message ID=0x0A

Request: Register Address (1 byte)
Response: Register Value (2 bytes)

Register	Start Bit	Bit Length	Name	Description
1	2	1	Link Status	1 = link is up
15	0	16	Rx Error Counter for Symbol Error frames	
31	12	1	Energy Detect	1 = Presence of signal on RX+/- analog wire pair 0 = No signal detected on RX+/-

3.3. Read SQI

Message ID=0x10

DATA 0:

bit 7							bit 0
-	-	-	-	-	SQI2	SQI1	SQI0

SQI 0..2 Signal quality indicator
000 – worse than class A SQI (unstable link)
001 – class A SQI (unstable link)
010 – class B SQI (unstable link)
011 – class C SQI (good link)
100 – class D SQI (good link; bit error rate < 1e-10)
101 – class E SQI (good link)

110 – class F SQI (very good link)
111 – class G SQI (very good link)

3.4. Cable Test

Message ID=0x11

Carries out a cable test on the 100BASE-T1 channel.

DATA 0:

bit 7							bit 0
-	-	-	-	-	-	SHORT	OPEN

OPEN	Open link detected
SHORT	Short link detected

3.5. T1 Master/Slave Overriding

Message ID=0x12

DATA 0:

bit 7							bit 0
-	-	-	-	-	-	MS1	MS0

MS1	**MS0**	**Mode**
0	0	Set by a dip switch
0	1	T1 Slave
1	0	T1 Master
1	1	Reserved

3.6. T1 Test Mode

Message ID=0x13

Can be used to generate Test symbols on the 100BASE-T1 channel.

DATA 0:

bit 7							bit 0
-	-	-	-	-	MODE2	MODE1	MODE0

MODE 0..2 Test mode selection:
000 – No test mode
001 – 100BASE-T1 test mode 1
010 – 100BASE-T1 test mode 2
011 – Test mode 3
100 – 100BASE-T1 test mode 4
101 – 100BASE-T1 test mode 5
110 – Scrambler and descrambler bypassed
111 – Reserved

Mode	Description
Test mode 1	Test mode 1 is used to test transmitter droop. In Test mode 1, the PHY transmits '+1' symbols for 600 ns followed by '-1 ' symbols for a further 600 ns. This sequence is repeated continuously.
Test mode 2	Test mode 2 is used to test transmitter timing jitter in Master mode. In Test mode 2, the PHY transmits the data symbol sequence {+1, −1} repeatedly. The transmission of the symbols is synchronized with the local external oscillator
Test mode 3	Same as test mode 2 only use Slave mode.
Test mode 4	Test mode 4 is used to test transmitter distortion. In Test mode 4, the PHY transmits the sequence of symbols generated by the scrambler polynomial gs1 = 1 + x9 + x11.
Test mode 5	Test mode 5 is used to test the transmit PSD mask. In Test mode 5, the PHY transmits a random sequence of PAM-3 symbols.

3.7. CAN Configuration

Message ID=0x51

When the CAN channel is configured with this command the Sync-Jump Width is always set to 2.
Important remark: This configures the CAN bus baud rate used for reading 100BASE-T1 status over CAN bus. Hence, do not use this message if you want to use the device as a USB-CAN interface (see 3.9).

Note: Available over USB VCP only.

DATA 0:

bit 7							bit 0
BAUD7	BAUD6	BAUD5	BAUD4	BAUD3	BAUD2	BAUD1	BAUD0

DATA 1:

bit 7							bit 0
BAUD15	BAUD14	BAUD13	BAUD12	BAUD11	BAUD10	BAUD9	BAUD8

DATA 2:

bit 7							bit 0
BAUD23	BAUD22	BAUD21	BAUD20	BAUD19	BAUD18	BAUD17	BAUD16

DATA 3:

bit 7							bit 0
BAUD31	BAUD30	BAUD29	BAUD28	BAUD27	BAUD26	BAUD25	BAUD24

DATA 4:

bit 7							bit 0
SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0

BAUD 0..31 **CAN baud rate**
SP 0..7 **CAN sample point [%]**

3.8. Read Status

Message ID=0x5D

This message reads the status of

Returns 4 bytes:

STATUS_BYTE	-	-	-

STATUS_BYTE:

bit 7							bit 0
-	-	-	-	T1_POLARITY	T1_ACTIVITY	T1_LINK	T100_LINK

T100_LINK 100BASE-TX link is up *(relevant for 100BASET1-MC-ETH only)*
T1_LINK 100BASE-T1 link is up *(relevant for 100BASET1-MC-ETH only)*
T1_ACTIVITY 100BASE-T1 activity
T1_POLARITY 100BASE-T1 polarity inversion (the same as “Polarity Detect” in T1 register 25 - see Read 100BASE-T1 PHY register)

3.9. CAN Message Specification

The following messages are available in order to use the device as a USB-CAN interface.
Note: Messages in this chapter are available over USB VCP only.

3.9.1.Channel Configuration

Message ID=0x60

This message configures a CAN channel. The time quanta for CAN controller are chosen automatically based on the sample point and baud rate. Sample point cannot always be exactly set to the desired value, and the closest value is used instead. The actual time quanta setting can be obtained by Read Configuration command. The 100BASE-T1 Media Converter CAN controller clock is 40 MHz and the 100BASE-T1 USB Interface CAN controller clock is 48 MHz.

Request:
Data section of a packet:

DATA 0	DATA 1	DATA 2	DATA 3
Channel	Configuration Register 1	Configuration Register 2	Configuration Register 3

Bit 0..1 **Channel**
00 – CAN 1
01 – Reserved
10 – Reserved
11 – Reserved

Configuration CHANNEL N Register 1:

bit 7							bit 0
PROTOCOL1	PROTOCOL0	AUTOSTART	ACK	ASP3	ASP2	ASP1	ASP0

Bit 6..7 **Protocol**
00 – CAN 2.0B
01 – Reserved
10 – Reserved
11 – Reserved

Bit 5 **Auto start**
0 – CAN channel is NOT automatically started on power-up
1 – CAN channel is automatically started on power-up

Bit 4 **Controller mode**
0 – Normal mode
1 – Silent mode

Bit 0..3 **Sample point**
0000 – 60%
0001 – 62,5%
0010 – 65%
0011 – 67,50%
0100 – 70%
0101 – 72,50%
0110 – 75%
0111 – 77,50%

1000 – 80%
1001 – 82,50%
1010 – 85%
1011 – 87,50%
1100 – 90%
1101 – Reserved
1110 – Reserved
1111 – Reserved

Configuration CHANNEL N Register 2:

bit 7							bit 0
Reserved	Reserved	Reserved	Reserved	Reserved	ABAUD2	ABAUD1	ABAUD0

Bit 3..7 **Reserved**

Bit 0..2 **Baud rate**
000 – 125 kBd
001 – 250 kBd
010 – 500 kBd
011 – 1 MBd
100..111 – Reserved

Configuration CHANNEL N Register 3:

bit 7							bit 0
Reserved	ASJW6	ASJW5	ASJW4	ASJW3	ASJW2	ASJW1	ASJW0

Bit 7 **Reserved**

Bit 0..6 **Synchronization jump width**
0000000 – 1
0000001 – 2
0000010 – 3
0000011 – 4
…
1111111 – 128

Response:
1) CAN channel reconfigured
   No data = success
2) CAN channel cannot be reconfigured
   General Error Message = error

   Possible reasons:
   - Wrong synchronization jump width.

Default configuration

Channel 1

- Normal mode
- Speed 500 kBd
- SJW 2
- Sample Point 80%
- Autostart disable

3.9.2.Channel Configuration Time Quanta Timing

Message ID=0x61

Request:
Data section of a packet:

DATA 0	DATA 1	DATA 2	DATA 3	DATA 4
Channel	Configuration Register 1	Configuration Register 2	Configuration Register 3	Configuration Register 4

DATA 5
Configuration Register 5

Bit 0..1 **Channel**
00 – CAN 1
01 – Reserved
10 – Reserved
11 – Reserved

Configuration CHANNEL N Register 1:

bit 7							bit 0
PROTOCOL1	PROTOCOL0	AUTOSTART	ACK	Reserved	Reserved	Reserved	Reserved

Bit 6..7 **Protocol**
00 – CAN 2.0B
01 – Reserved
10 – Reserved
11 – Reserved

Bit 5 **Auto start**
0 – CAN channel is NOT automatically started on power-up
1 – CAN channel is automatically started on power-up

Bit 4 **Controller mode**
0 – Normal mode
1 – Silent mode

Bit 0..3 **Reserved**

Configuration CHANNEL N Register 2:

bit 7							bit 0
ATSEG1_7	ATSEG1_6	ATSEG1_5	ATSEG1_4	ATSEG1_3	ATSEG1_2	ATSEG1_1	ATSEG1_0

Bit 0..7 **Time segment 1**
0000 0000 – 1
0000 0001 – 2
0000 0010 – 2
0000 0011 – 3
…
1111 1111 – 256

Configuration CHANNEL N Register 3:

bit 7							bit 0
Reserved	ATSEG2_6	ATSEG2_5	ATSEG2_4	ATSEG2_3	ATSEG2_2	ATSEG2_1	ATSEG2_0

Bit 7 **Reserved**

Bit 0..6 **Time segment 2**
000 0000 – 1
000 0001 – 2
000 0010 – 3
000 0011 – 4
…
111 1111 – 128

Configuration CHANNEL N Register 4:

bit 7							bit 0
APRESC_7	APRESC_6	APRESC_5	APRESC_4	APRESC_3	APRESC_2	APRESC_1	APRESC_0

Bit 0..7 **Prescaler**
0000 0000 – 1
0000 0001 – 2
0000 0010 – 2
0000 0011 – 3
…
1111 1111 – 256

Configuration CHANNEL N Register 5:

bit 7							bit 0
Reserved	ASJW6	ASJW5	ASJW4	ASJW3	ASJW2	ASJW1	ASJW0

Bit 7 **Reserved**

Bit 0..6 **Synchronization jump width**
000 0000 – 1
000 0001 – 2
000 0010 – 3
000 0011 – 4
…
111 1111 – 128

Response:
1) CAN channel reconfigured
   No data = success
2) CAN channel cannot be reconfigured
   General Error Message = error

   Possible reasons:
   - Wrong arbitration or data jump width.

3.9.3.Read Configuration

This command reads CAN interface settings. If configuration is set by precise timing message, 0xF values are set instead of Sample point and Baud rate values.

Message ID=0x62

Request:

DATA 0
Channel

Bit 0..1 **Channel**
00 – CAN 1
01 – Reserved
10 – Reserved
11 – Reserved

Response:

DATA 0	**DATA 1**	**DATA 2**	**DATA 3**	**DATA 4**	**DATA 5**
Configuration Register 1	Configuration Register 2	Configuration Register 3	Configuration Register 4	Configuration Register 5	Configuration Register 6

DATA 6
Echo configu-ration register 7

Configuration CHANNEL N Register 1:

bit 7							bit 0
PROTOCOL1	PROTOCOL0	AUTOSTART	ACK	ASP3	ASP2	ASP1	ASP0

Bit 6..7 **Reserved**

Bit 5 **Auto start**
0 – CAN channel is NOT automatically started on power-up
1 – CAN channel is automatically started on power-up

Bit 4 **Controller mode**
0 – Normal mode
1 – Silent mode

Bit 0..3 **Sample point**
0000 – 60%
0001 – 62,5%
0010 – 65%
0011 – 67,50%
0100 – 70%
0101 – 72,50%
0110 – 75%
0111 – 77,50%
1000 – 80%
1001 – 82,50%
1010 – 85%
1011 – 87,50%
1100 – 90%
1101 – Reserved

1110 – Reserved
1111 – Reserved

Configuration CHANNEL N Register 2:

bit 7							bit 0
Reserved	Reserved	Reserved	Reserved	Reserved	ABAUD2	ABAUD1	ABAUD0

Bit 3..7 **Reserved**

Bit 0..2 **Baud rate**
000 – 125 kBd
001 – 250 kBd
010 – 500 kBd
011 – 1 MBd
100..111 – Reserved

Configuration CHANNEL N Register 3:

bit 7							bit 0
Reserved	ASJW6	ASJW5	ASJW4	ASJW3	ASJW2	ASJW1	ASJW0

Bit 7 **Reserved**

Bit 0..6 **Synchronization jump width**
0000000 – 1
0000001 – 2
0000010 – 3
0000011 – 4
…
1111111 – 128

Configuration CHANNEL N Register 4:

bit 7							bit 0
ATSEG1_7	ATSEG1_6	ATSEG1_5	ATSEG1_4	ATSEG1_3	ATSEG1_2	ATSEG1_1	ATSEG1_0

Bit 0..7 **Time segment 1**
0000 0000 – 1
0000 0001 – 2
0000 0010 – 2
0000 0011 – 3
…
1111 1111 – 256

Configuration CHANNEL N Register 5:

bit 7							bit 0
Reserved	ATSEG2_6	ATSEG2_5	ATSEG2_4	ATSEG2_3	ATSEG2_2	ATSEG2_1	ATSEG2_0

Bit 7 **Reserved**

Bit 0..6 **Time segment 2**
000 0000 – 1
000 0001 – 2
000 0010 – 3
000 0011 – 4
…
111 1111 – 128

Configuration CHANNEL N Register 6:

bit 7							bit 0
APRESC_7	APRESC_6	APRESC_5	APRESC_4	APRESC_3	APRESC_2	APRESC_1	APRESC_0

Bit 0..7 **Prescaler**
0000 0000 – 1
0000 0001 – 2
0000 0010 – 2
0000 0011 – 3
…
1111 1111 – 256

Echo configuration CHANNEL N register 7:

bit 7						bit 0	
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	TXECHO	RXECHO

Bit 2..7 **Reserved**

Bit 1 **TX Echo on/off**

0 – Echo Off (default)

1 – Echo On

Bit 0 **RX Echo on/off**

0 – Echo Off

1 – Echo On (default)

3.9.4.Save Configuration

Message ID=0x63

This command saves to EEPROM the configuration registers set by commands **Channel Configuration** and **Channel Configuration Time Quanta Timing.** Also, the **Frame Echo Configuration** is saved.

Request 0x63:

DATA 0
Channel

Bit 0..1	**Channel**
	00 – CAN 1
	01 – Reserved
	10 – Reserved
	11 – Reserved

1. CAN channel reconfigured
   No data = success
2. CAN channel can't be reconfigured
   General Error Message = error

3.9.5.Load Configuration

Message ID=0x64

Request 0x64:

DATA 0
Channel

Bit 0..1	**Channel**
	00 – CAN 1
	01 – Reserved
	10 – Reserved
	11 – Reserved

Type of response:
Configuration was load
Message with ID 0x64 and no data = success

3.9.6.Default Configuration

Message ID=0x65

Loads a default CAN bus configuration.

Request 0x65:

DATA 0
Channel

Bit 0..1	**Channel**
	00 – CAN 1
	01 – Reserved
	10 – Reserved
	11 – Reserved

Type of response:
Default configuration was saved

No data = success

General Error Message = error

Possible reasons:
- CAN channel is already running.

3.9.7.Frame Echo Configuration

Message ID=0x66

Request 0x66:

DATA 0	DATA 1
Channel	Echo configuration

bit 7							bit 0
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	TXECHO	RXECHO

Bit 2..7 **Reserved**

Bit 1 **TX Echo on/off**

0 – Echo Off (default)

1 – Echo On

Bit 0 **RX Echo on/off**

0 – Echo Off

1 – Echo On (default)

Type of response:
Echo configuration was changed

No data = success

General Error Message = error

Possible reasons:
- CAN channel is already running.

3.9.8.Start Channel

Message ID=0x67

Starts a CAN channel.

Request 0x67:

DATA 0
Channel

Bit 0..1 **Channel**
00 – CAN 1
01 – Reserved
10 – Reserved
11 – Reserved

Type of response:
CAN channel was successfully started:
No data = success

CAN channel error:
General Error Message = error

Possible reasons:
- CAN channel is already running.

3.9.9.Stop Channel

Message ID=0x68

Stops a CAN channel.

Request:

DATA 0
Channel

Bit 0..1 **Channel**
00 – CAN 1
01 – Reserved
10 – Reserved
11 – Reserved

Type of response:
CAN channel was successfully stopped:
No data = success

CAN channel error:
General Error Message = error

3.9.10. Frame Transmission

Message ID=0x70

Transmits a CAN frame on the CAN bus. The structure of the message differs for Standard and Extended CAN Identifiers. The message length for Standard ID is 5 bytes long, whilst for Extended ID it is 7 bytes long. The format of CAN ID is LSB (Intel).

Request:

Standard ID (EXT ID==0)

DATA 0	DATA 1	DATA 2	DATA 3	DATA 4	DATA n
Channel	MESSAGE_INFO	ID0	ID1	DLC	DATA bytes

Extended ID (EXT ID==1)

DATA 0	DATA 1	DATA 2	DATA 3	DATA 4	DATA 5	DATA 6	DATA n
Channel	MESSAGE_INFO	ID0	ID1	ID2	ID3	DLC	DATA bytes

Channel select:

Bit 0..1 **Channel**

00 – CAN 1
01 – Reserved
10 – Reserved
11 – Reserved

MESSAGE_INFO:

bit 7							bit 0
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RTR	EXT ID

Bit 2..7 **Reserved**

Bit 1 **RTR**

0 – Data frame
1 – Remote frame

Bit 0 **EXT ID**

0 – Standard ID
1 – Extended ID

Response:

CAN Frame was successfully sent:

No data = success

CAN channel error:

General Error Message = error

Possible reasons:

- Wrong bit configuration.

3.9.11. Frame Reception

Message ID=0x71

Message response has the same structure as Frame. The message is sent by the device once a CAN frame is received from the bus. Error frames are handled like **CAN bus errors** (0xC0 – see 2.4).

4. Message Examples

The following depicts some communication examples over USB VCP.

4.1. Device Setting and Info

Command	Bytes [hex]
Read serial number	02 5A 00 5A 03 Gateway response: 02 5A 04 40 01 07 05 6F 03
Read status 100BASE-TX link is up, 100BASE-T1 link is down, 100BASE-T1 no activity, 100BASE-T1 polarity no inversion	02 5D 00 5D 03 Gateway response: 02 5D 04 01 00 00 00 62 03
Read 100BASE-T1 PHY register Link down	02 01 01 01 03 03 Gateway response: 02 01 02 E1 01 E5 03
Read 100BASE-TX PHY register Link up	02 0A 01 01 0C 03 Gateway response: 02 0A 02 6D 78 F1 03
Read SQI class G SQI (very good link)	02 10 00 10 03 Gateway response: 02 10 01 07 18 03

4.2. USB-CAN Interface

Command	Bytes [hex]
Configure CAN channel Channel 0, CAN, AutoStart, Normal mode, Arbitration SP = 80%, Arbitration Baud = 500 kBd, Arbitration SJW = 2	02 60 04 00 28 02 01 8F 03 Gateway response: 02 60 00 60 03
Configure CAN channel timing Channel 0, CAN, AutoStart, Normal mode, Arbitration T_seg1 = 15, Arbitration T_seg2 = 4, Arbitration Prescaler = 4, Arbitration SJW = 2	02 61 06 00 20 0E 03 03 01 9C 03 Gateway response: 02 61 00 61 03
Start CAN channel Channel 0	02 67 01 00 68 03 Gateway response: 02 67 00 67 03
Transmit CAN Frame Channel 0, format = CAN, ID = 0x01FF, DLC = 7, Data (hex) = 05 04 50 06 06 08 14	02 70 0C 00 00 FF 01 07 05 04 50 06 06 08 14 04 03 Gateway response: 02 70 00 70 03
Stop CAN channel Channel 0	02 68 01 00 69 03 Gateway response: 02 68 00 68 03

5. Contact

MACH SYSTEMS s.r.o.
www.machsystems.cz
info@machsystems.cz
Czech Republic

Company Registration: 29413893
VAT no.: CZ29413893