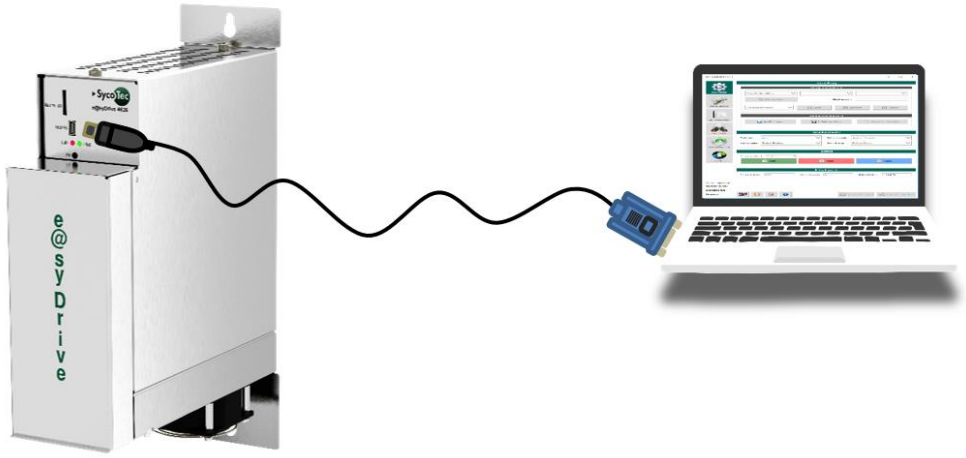
Interface Description RS232 EN
of HF Inverters e@syDrive 4624, 4625, 4626



INDUSTRIAL DRIVES

SycoTec

CAUTION

This RS232 interface description is only valid in connection with the operating manual HF Inverters e@syDrive 4624, 4625, 4626 (material no. 2.002.1912)!

- *The safety information must be observed before commissioning!*

SycoTec GmbH & Co. KG
Wangener Strasse 78
88299 Leutkirch
Germany

Phone +49 7561 86-0
Fax +49 7561 86-371
info@sycotec.eu
www.sycotec.eu

Table of Contents

1 User Information

1.1 Symbols Used

Operating Manual / Unit

CAUTION	Indicates a hazardous situation that can cause damage to property or mild to moderate injuries.
NOTICE	Important information for operator and engineer.

1.2 Important Information

Target group: This document is intended for machine manufacturers and persons responsible for putting into service and operating the frequency inverter e@syDrive 4624, 4625, 4626.

CAUTION
The operating manual must be read by the user/operator before starting up the unit for the first time in order to avoid incorrect operation and other damage. Duplication and distribution of the operating manual require SycoTec's prior consent.

All specifications, information and properties of the product described in the operating manual correspond to the status on going to press.
Modifications and improvements to the product as a result of new technical developments are possible.
This does not imply any right to retrofitting of existing units.

SycoTec assumes no responsibility for damage arising through:

- use of incorrect information
- improper use

NOTICE
The safety and application instructions in the hardware description and in the software description of the frequency inverter must be observed!

1.3 Intended Use

This document describes the serial (RS232) interface of the e@syDrive 4624, 4625, 4626 frequency inverters. The serial (RS232) interface can be used to control the frequency inverters e@syDrive 4624, 4625, 4626 (start/stop, speed, direction of rotation, reset) and to set the control parameters. Error messages and display values (identification and status) of the frequency inverters can be read out.

1.4 Abbreviations

A	Ampere
ACK	Acknowledge – Response without data content
ASCII	American Standard Code for Information Interchange
char.	ASCII-Character
CHK	Checksum – Checksum of a message
CMD	Command – Request that starts a defined command
CRC	Cyclic Redundancy Check
DATA	Data content
DEC	Decimal
ETX	End of Text – End detection of a message
h	Hour
HEX	Hexadecimal
HF	High frequency
Hz	Hertz
ID	Identifikation
LEN	Length of Data – Length of data content
Msg.	Message
MSGID	Message ID – Message identification
n	Number of bytes of a message
PC	Personal Computer
PTC	Positive Temperature Coefficient
REQ	Request – Request without data content
RES	Response – Response with data content
rpm	Revolutions per minute
RS232	Recommended Standard 232
SET	Setting – Request with data content
STX	Start of Text – Start detection of a message
VER	Version of a message
V	Volt
W	Watt

2 Scope of Supply

Operation manual interface description RS232
of HF inverters e@syDrive 4624, 4625, 4626 Material No. 2.002.9501

NOTICE
Check that all parts are present.

3 RS232 connection

The serial (RS232) communication interface connects the service PC and the frequency inverters e@syDrive 4624, 4625, 4626.

NOTICE
The protocol of the serial (RS232) communication interface is implemented from firmware version 4.3.3 onwards.

3.1 Pin assignment

The settings of the serial interface are:

Baud rate	38400
Data bits	8
Stop bits	1
Parity	None
Flow control	None

3.2 Connection

A standard mini-USB cable (2.002.2675) is required to connect to a computer (included in the scope of delivery of the frequency inverters e@syDrive 4624, 4625, 4626).

4 Message protocol

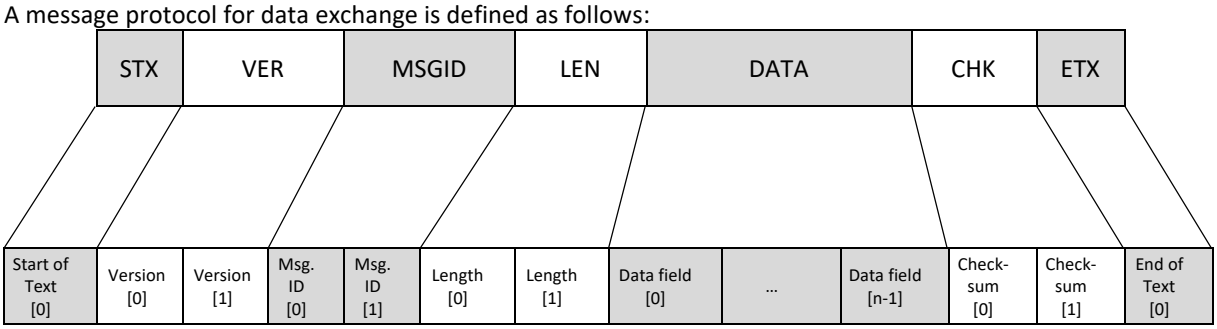
This chapter describes the transfer protocol for the data exchange between client and the e@syDrive device. The transfer protocol is described as transport oriented and application oriented.

Essentially, it should be noted that the initiative for data transfer always originates from the client. This applies both to commanding and to configuration and to data logging.

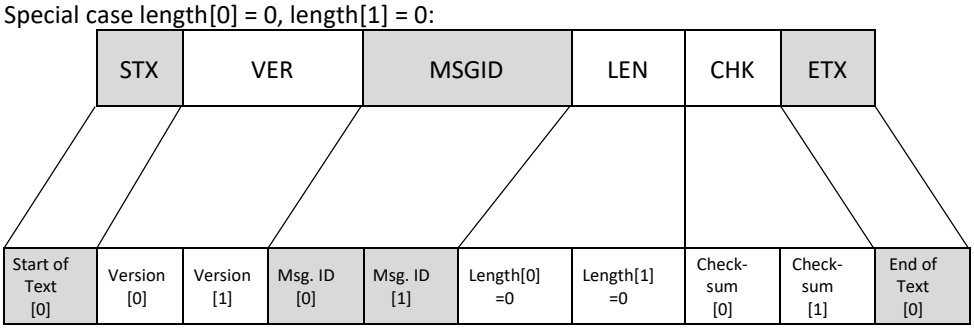
4.1 Transport oriented

4.1.1 Protocol structure

A message protocol for data exchange is defined as follows:



Special case length[0] = 0, length[1] = 0:



4.1.2 Protocol definition

In the following, the values of the individual contents are defined.

Byte	Name	Character	Description
0	Start of Text	2_{HEX}	Start of Text
1, 2	Version	'0'$_{char}$, '1'$_{char}$	Version number of the message
3, 4	Msg. ID	'#'$_{char}$, '#'$_{char}$	Message ID
5, 6	Length	'#'$_{char}$, '#'$_{char}$	Length of the data field
	Data field[x] Data field[y]	'#'$_{char}$ '#'$_{char}$ ...	Data
	Checksum	'#'$_{char}$, '#'$_{char}$	CRC Checksum
	End of Text	3_{HEX}	End of Text

Special case length[0] = 0, length[1] = 0:

Byte	Name	Character	Description
0	Start of Text	2_{HEX}	Start of Text
1, 2	Version	'0'$_{char}$, '1'$_{char}$	Version number of the message
3, 4	Msg. ID	'#'$_{char}$, '#'$_{char}$	Message ID
5, 6	Length	'0'$_{char}$, '0'$_{char}$	Length of the data field
7, 8	Checksum	'#'$_{char}$, '#'$_{char}$	CRC Checksum
9	End of Text	3_{HEX}	End of Text

4.1.2.1 Definition of 'STX'

'STX' means Start of Text. This identifier marks the beginning of a message. The size of 'STX' is exactly 1 byte transport oriented, always contains the same value and is written in ASCII.

$$STX \in ['STX'_{char}] \vee [02_{HEX}]$$

4.1.2.2 Definition of 'Version'

The size of 'Version' is exactly 2 bytes transport oriented. The value within 'Version' is written in ASCII and is interpreted hexadecimal. This means that a total of 16^2=256 message versions can be defined.

$$Version[0] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$
$$Version[1] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$

4.1.2.3 Definition of 'Msg. ID'

The size of 'Msg. ID' is exactly 2 bytes transport oriented. The value within 'Msg. ID' is written in ASCII and is interpreted hexadecimal. This means that a total of 16^2=256 messages can be defined.

$$Msg.ID[0] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$
$$Msg.ID[1] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$

4.1.2.4 Definition of 'Length'

The size of 'Length' is always exactly 2 bytes transport oriented. The value of 'Length' is written in ASCII and interpreted in hexadecimal. This way up to 16^2=256 data bytes can be packed into one message.

$$Length[0] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$
$$Length[1] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$

4.1.2.5 Definition of 'Data field'

The data area contains values in ASCII form and is interpreted hexadecimal. The size of the 'Data field' depends on the specification of the 'Length'. The general definition is as follows:

$$Data\ field[x] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$
$$Data\ field[y] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$

4.1.2.6 Definition of 'Checksum'

The 'Checksum' is exactly 2 bytes large and contains the checksum of all data bytes hexadecimal, whereby an overflow resulting from this is ignored. The value of 'Checksum' is written in ASCII and interpreted in hexadecimal.

$$Checksum[0] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$
$$Checksum[1] \in ['0'_{char}\ldots'f'_{char}] \vee [30_{HEX}\ldots39_{HEX};61_{HEX}\ldots66_{HEX}]$$

4.1.2.7 Definition of 'ETX'

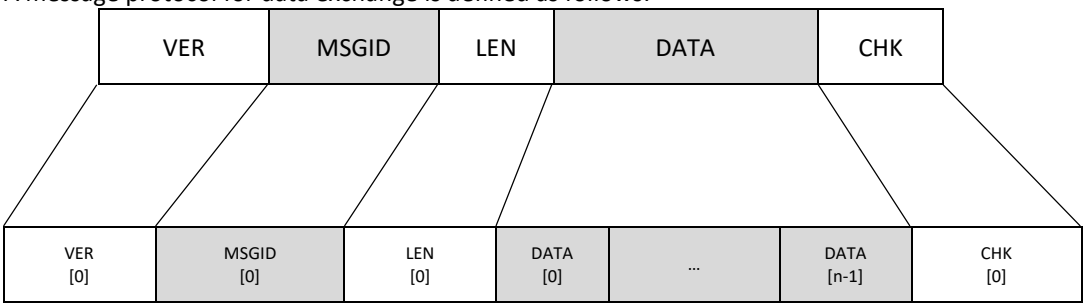
'ETX' means End of Text. This identifier marks the end of a message. The size of 'ETX' is exactly 1 byte and always contains the same value:

$$ETX \in ['ETX'_{char}] \vee [03_{HEX}]$$

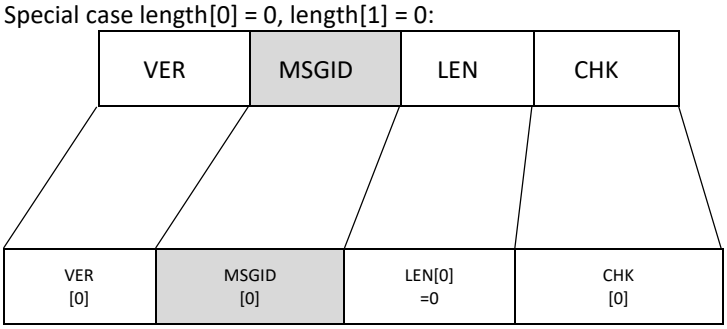
4.2 Application oriented

4.2.1 Protocol structure

A message protocol for data exchange is defined as follows:



Special case length[0] = 0, length[1] = 0:



4.2.2 Protocol definition

In the following, the values of the individual contents are defined.

The following table describes the structure of the protocol:

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x##	Message ID
2	LEN = n	0x##	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0x##, …	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Special case:

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x##	Message ID
2	LEN = n	0x00	Length of the data field
3+(n-1)+1	CHK	0x##	CRC Checksum

4.2.2.1 Definition of 'VER'

'VER' stands for Version. The version name ensures the unmistakable identification of the protocol version. The size of 'VER' is 1 byte application oriented. A total of 16^2=256 message versions can be defined.

$$VER\,[0] \in [0x00 : 0xFF]$$

4.2.2.2 Definition of the 'MSGID'

'MSGID' stands for Message ID. This ID is used to identify the message. Once this ID has been determined, the exact structure of the entire message is known. The size of 'MSGID' is 1 byte application oriented. A total of 16^2=256 messages can be defined.

MSGID [0] ∈ [0x00 : 0xFF]

4.2.2.3 Definition 'LEN'

'LEN' defines the Length of Data, i.e. the number of available data bytes. This information can be used to calculate the total message length. The size of 'LEN' is always 1 byte application oriented.
Up to 16^2=256 data bytes can be packed into one message.

LEN [0] ∈ [0x00 : 0xFF]

4.2.2.4 Definition of 'DATA'

'DATA' defines the data area within the message. The size of 'DATA' is LEN=n Byte application oriented, where i ∈ [n : 1] with i_{max}=n, where n ≥ 1. The general definition is valid:

DATA [0 : n-i] ∈ [0x00 : 0xFF]

4.2.2.5 Definition 'CHK'

'CHK' defines the checksum. The checksum can be used to check whether error-free data transmission has taken place. The checksum is 1 byte and contains the checksum of all data bytes, whereby any resulting overflow is ignored.

Algorithm: CHK = VER + MSGID + LEN + DATA

Format: CHK [0] ∈ [0x00 : 0xFF]

5 Message definition

The following sections describe the messages, their assignment, identification (MSGID) and data content (DATA) in more detail.

5.1 SET - Setting

5.1.1 MSGID: 0x10 SET Basic Parameters

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x10	Message ID
2	LEN = n	0x04	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]		Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=4:

Index [0:n-1]	Data	Description	Explanation
0, 1	P1	Rated frequency	Frequency setpoint for the motor The value is given in Hz
2	-	-	unused
3	P8	Speed display	Selection of the speed display (frequency/speed) in the SycoDrive software 0x01: in Hz 0x02: in rpm

5.1.2 MSGID: 0x18 SET Start Parameters

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x18	Message ID
2	LEN = n	0x03	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]		Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=3:

Index [0:n-1]	Data	Description	Explanation
0	P140	Input for start	Input for motor start 0x01: SycoDrive 0x02: Digital inputs 0x03: CAN
1	P141	Input for rated frequency	Input for frequency setpoint 0x01: SycoDrive 0x02: Digital inputs 0x03: CAN 0x04: Analog input U(0…10V) 0x05: Analog input I(0…20mA)
2	P146	Direction of rotation	Direction of motor rotation 0x01: Clockwise 0x02: Counter-clockwise 0x03: Digital input

NOTICE
To start the motor spindle via RS232 the Input for start (P140) and the Input for rated frequency (P141) must be set to "SycoDrive".

5.2 REQ - Request

5.2.1 MSGID: 0xCF REQ Display Values (status)

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xCF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0x59	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0x59	Request from MSGID 0x59	Request of the RES Display Values (status)

The RES Display Valued (status) contains the following data: Rated frequency (P10), Peak current (P11), Actual frequency (P13), Motor voltage (P15), DC link voltage (motor) (P16), Active current (P18), Active power (P19), Motor code (P20) and runtimes of Inverter (P25) and Motor (P26).

5.2.2 MSGID: 0xCF REQ Display Values (identification)

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xCF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0x5A	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0x5a	Request from MSGID 0x5A	Request of the RES Display Values (identification)

The RES Display Valued (identification) contains the following data: 1st error (P30), 2nd error (P31), 3rd error (P32), 4th error (P33), 5th error (P34), Inverter (P36), Firmware (P37) and Serial number (P39).

5.2.3 MSGID: 0xCF REQ Statusout

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xCF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0x60	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0x60	Request from MSGID 0x60	Request of the RES Statusout

The RES Statusout contains the following data: Motor code (P20), operating status, error status and error number.

5.3 CMD - Command

5.3.1 MSGID: 0xA0 CMD System Start

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xA0	Message ID
2	LEN = n	0x00	Length of the data field
3	CHK	0x##	CRC Checksum

5.3.2 MSGID: 0xA1 CMD System Stop

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xA1	Message ID
2	LEN = n	0x00	Length of the data field
3	CHK	0x##	CRC Checksum

5.3.3 MSGID: 0xA2 CMD System Reset

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xA2	Message ID
2	LEN = n	0x00	Length of the data field
3	CHK	0x##	CRC Checksum

5.4 RES - Response

5.4.1 MSGID: 0x59 RES Display Values (status)

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x59	Message ID
2	LEN = n	0x1B	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]		Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=27:

Index [0:n-1]	Data	Description	Explanation
0, 1	P10	Rated frequency	Display of rated frequency in Hz
2, 3	P11	Peak current	Display of actual peak current in A (Divided by 100 to normalize)
4, 5	-	-	unused
6, 7	P13	Actual frequency	Display of inverter actual frequency in Hz
8, 9	-	-	unused
10, 11	P15	Motor voltage	Display of actual output voltage in V (Divided by 100 to normalize)
12, 13	P16	DC link voltage (motor)	Display of actual DC link voltage in V (Divided by 100 to normalize)
14, 15	P18	Active current	Display of actual phase current in A (Divided by 100 to normalize)
16, 17	P19	Active power	Display of actual active power in W (Divided by 10 to normalize)
18	P20	Motor code	Display of actual motor coding
19, 20, 21, 22	P25	Inverter	Display of actual inverter runtime in h
23, 24, 25, 26	P26	Motor	Display of actual motor runtime in h

5.4.2 MSGID: 0x5A RES Display Values (identification)

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x5A	Message ID
2	LEN = n	0x0D	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]		Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=13:

Index [0:n-1]	Data	Description	Explanation
0	P30	1st Error	Error number with highest priority
1	P31	2nd Error	Error number with second highest priority
2	P32	3rd Error	Error number with third-highest priority
3	P33	4th Error	Error number with fourth-highest priority
4	P34	5th Error	Error number with fifth-highest priority
5, 6	P36	Inverter	Inverter type
7, 8	P37	Firmware	Firmware version
9, 10, 11, 12	P39	Serial number	Serial number of the inverter

5.4.3 MSGID: 0x60 RES Statusout

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x60	Message ID
2	LEN = n	0x04	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]		Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=4:

Index [0:n-1]	Data	Description	Explanation
0		statusout1	Error number 0-255
1		statusout2	Error state 0x00: no errors 0x02: warning 0x03: error
2		statusout3	Status state Bit 0: Flag STILLSTAND 1: Motor stopped Bit 1: Flag STILLSTAND 1: Motor stopped Bit 2: Flag N_Soll_OK 1: Nominal speed reached Bit 3: Flag WARNSTROM 1: Current limit active Bit 4: Flag MOTORTEMPERATUR 1: Over temperature PTC Bit 5: unused Bit 6: unused
3		statusout4	Motor coding 0x00: inactive 0x01 – 0x10: active M1 – M16

5.5 ACK - Acknowledge

5.5.1 MSGID: 0xFF ACK Basic Parameters

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xFF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0x10	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0x10	Acknowledge from MSGID 0x10	Acknowledgement of SET Basic Parameters

5.5.2 MSGID: 0xFF ACK Start Parameters

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xFF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0x18	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0x18	Acknowledge from MSGID 0x18	Acknowledgement of Start Parameters

5.5.3 MSGID: 0xFF ACK System Start

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xFF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0xA0	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0xA0	Acknowledge from MSGID 0xA0	Acknowledgement of System Start

5.5.4 MSGID: 0xFF ACK System Stop

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xFF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0xA1	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0xA1	Acknowledge from MSGID 0xA1	Acknowledgement of System Stop

5.5.5 MSGID: 0xFF ACK System Reset

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xFF	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0xA2	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0xA2	Acknowledge from MSGID 0xA2	Acknowledgement of System Reset

6 Message examples

6.1 Request display values

Structure of the instruction for requesting the operating values:

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0xcf	Message ID
2	LEN = n	0x01	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]	0x59	Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=1:

Index [0:n-1]	Data	Description	Explanation
0	0x59	Request from MSGID 0x59	Request of the RES Display Values (status)

Calculation of the cecksum:

The ASCII values of the individual characters in the command are summarized in hexadecimal. The last 2 characters are taken as checksum, the rest is ignored.

The checksum of this message:

ASCII:	0	1	+	c	f	+	0	1	+	5	9	=	1f9		
	↓	↓		↓	↓		↓	↓		↓	↓		↓		
HEX:	30	31		63	66		30	31		35	39		'f' 'g'		
	VER			MSGID			LEN			DATA			CHK		

This results in f9 for the checksum.

The message requesting the display values is as follows:

'STX'	'0' '1'	'c' 'f'	'0' '1'	'5' '9'	'f' '9'	'ETX'

Response to the request for display values:

Byte	Name	Character	Description
0	VER	0x01	Version number of the message
1	MSGID	0x59	Message ID
2	LEN = n	0x1b	Length of the data field
3 to 3+(n-1)	DATA[0:n-1]		Data
3+(n-1)+1	CHK	0x##	CRC Checksum

Data n=27:

Index [0:n-1]	Data	Description	Explanation
0, 1	P10	Rated frequency	Display of rated frequency in Hz
2, 3	P11	Peak current	Display of actual peak current in A (Divided by 100 to normalize)
4, 5	-	-	unused
6, 7	P13	Actual frequency	Display of inverter actual frequency in Hz
8, 9	-	-	unused
10, 11	P15	Motor voltage	Display of actual output voltage in V (Divided by 100 to normalize)
12, 13	P16	DC link voltage (motor)	Display of actual DC link voltage in V (Divided by 100 to normalize)
14, 15	P18	Active current	Display of actual phase current in A (Divided by 100 to normalize)
16, 17	P19	Active power	Display of actual active power in W (Divided by 10 to normalize)
18	P20	Motor code	Display of actual motor coding
19, 20, 21, 22	P25	Inverter	Display of actual inverter runtime in h
23, 24, 25, 26	P26	Motor	Display of actual motor runtime in h

All values are given in hexadecimal, e.g. a response from the frequency inverter:

		P10	P11	-	P13	-	P15	P16	P18	P19	P20	P25	P26		
'STX'	01591b	00a7	0320	0000	00a7	0000	0230	02b2	0071	005a	00	0000 010d	0000 0006	ab	'ETX'

	HEX		DEC			
P10:	00a7	=	167	->	167 Hz	
P11:	0320	=	800	->	8 A	(Divided by 100 to normalize)
-	0000					
P13:	00a7	=	167	->	167 Hz	
-	0000					
P15:	0230	=	560	->	5,6 V	(Divided by 100 to normalize)
P16:	02b2	=	690	->	6,9 V	(Divided by 100 to normalize)
P18:	0071	=	113	->	1,13 A	(Divided by 100 to normalize)
P19:	005a	=	90	->	9 W	(Divided by 10 to normalize)
P20:	00	=	0	->	0	
P25:	0000010d	=	269	->	269 h	
P26:	00000006	=	6	->	6 h	

6.2 Start / Stop

The following commands are required to operate a spindle via RS232:

- CMD System Start
- CMD System Stop
- SET Basic Parameters
- SET Start Parameters

CMD System Start:
Starting the spindle

CMD System Stop:
Stopping the spindle

SET Start Parameters:
The following parameters must be set before the start:
P140 – Input for start

- 0x01 -> SycoDrive
- 0x02 -> Digital inputs
- 0x03 -> CAN

P141 – Input for rated frequency

- 0x01 -> SycoDrive
- 0x02 -> Digital inputs
- 0x03 -> CAN
- 0x04 -> Analog input U(0…10V)
- 0x05 -> Analog input I(0…20mA)

P146 – Direction of rotation

- 0x01 -> Clockwise
- 0x02 -> Counter-clockwise
- 0x03 -> Digital input

SET Basic Parameters:
The following values must be defined:
P1 – Rated frequency

- Frequency in Hz

P8 – Speed display in the software SycoDrive

- 0x01 -> in Hz
- 0x02 -> in rpm

Sending a message for setting the following basic parameters:

P1 = 400 Hz -> 0x190
P8 = in Hz -> 0x01

The message for setting the basic parameters is as follows:

'STX'	'0' '1'	'1' '0'	'0' '4'	'01' '90' (P1) '00' '01' (P8)	'b' '1'	'ETX'

Warranty Conditions

Under current SycoTec delivery and payment conditions, SycoTec undertakes warranty for satisfactory function and freedom from faults in material and manufacture for a period of 12 months from the date of sale certified by the vendor.
In the event of justifiable complaints, SycoTec shall supply spare parts or carry out repairs free of charge under warranty. SycoTec accepts no liability for defects and their consequences which have arisen or could have arisen as a result of natural wear and tear, improper handling, cleaning or maintenance, non-compliance with the maintenance, operating or connecting instructions, corrosion, impurities in the air supply or chemical or electrical influences which are unusual or not admissible in accordance with SycoTec's standards. The warranty claims shall become null and void if defects or their consequences can be attributed to interventions in or modifications to the product. Warranty claims can only be validated if they are notified immediately in writing to SycoTec.
A copy invoice or delivery note clearly showing the manufacture number shall be attached if products are returned.

2.002.9501 / 2020-02

(DE = Original)

SycoTec GmbH & Co. KG
Wangener Strasse 78
88299 Leutkirch
Germany

Phone +49 7561 86-0
Fax +49 7561 86-371
info@sycotec.eu
www.sycotec.eu