

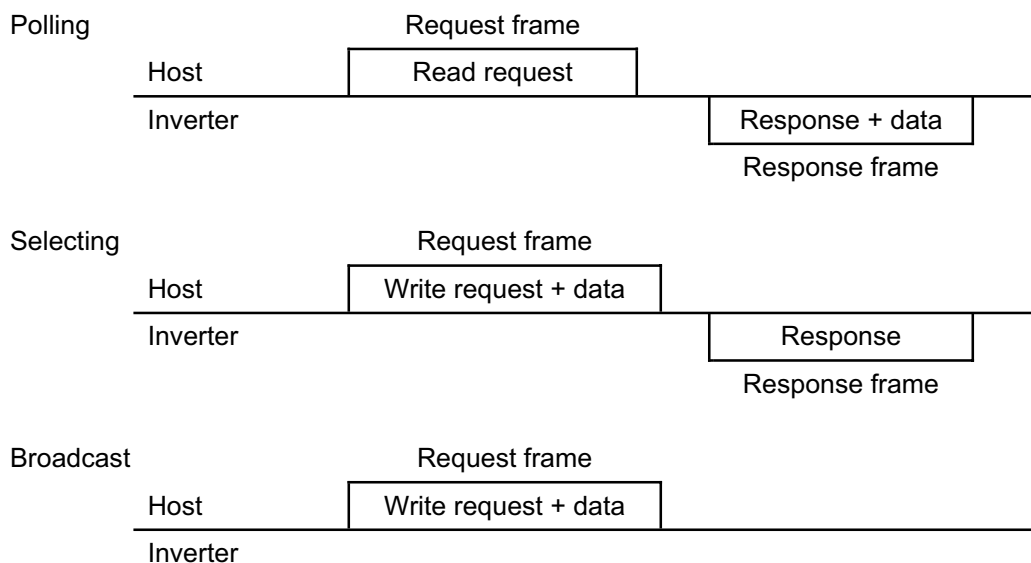
4.1 Messages

4.1.1 Message formats

The polling/selecting system is used to transmit and receive messages. The inverter always waits for selecting (write requests) or polling (read requests) from a host such as a personal computer or PLC.

When the inverter in the standby status receives a request frame from the host addressed to itself (local station) and considers the request frame to have been normally received, the inverter executes the transaction in response to the request, and sends back an acknowledgement (ACK) frame (or response and data in the case of polling). If the inverter judges that the receiving failed, it returns negative acknowledgment (NAK) frame. In the case of broadcast (all station batch selecting), the inverter does not send back response.

(Each frame is described in "4.1.2 Transmission frames.")



Broadcast (all station batch selecting)

A frame with the station address set to 99 is treated by all inverters as broadcast.

By using broadcast, operation or frequency commands can be simultaneously assigned to all inverters.

In broadcast communications, only selecting of S01, S05, S06, S13, and S14 in the standard frame, and commands (W, E, a, e, f, and m) in the optional frame are valid.

4.1.2 Transmission frames

Transmission frames are classified into two types; standard frames with which all communications functions are available, and optional frames, allowing high-speed communications, but whose function is limited to issuing commands to and monitoring the inverter.

All characters (including BCC) comprising both standard and optional frames are represented by ASCII codes. The lengths of standard and optional frames are as shown in Table 4.1 below:

Table 4.1 Lengths of transmission frames

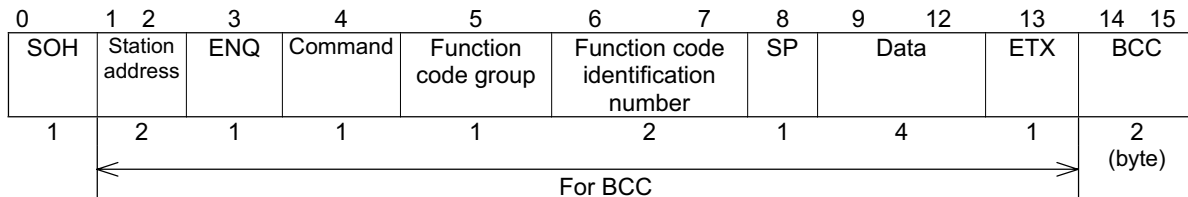
Frame type			Frame length
Standard frame	Selecting	Request	16 bytes
		Response	16 bytes
	Polling	Request	16 bytes
		Response	16 bytes
Optional frame	Selecting	Request	12 bytes
		Response	8 bytes
	Polling	Request	8 bytes
		Response	12 bytes

[1] Standard frame

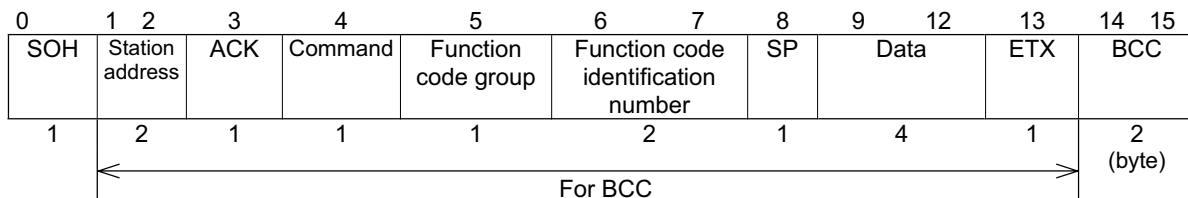
Standard frames are classified into request frame, ACK frame, and NAK frame, and their frame configurations are as shown below.

For the meanings of the fields comprising each frame, see the tables shown on the pages that follow.

Request frame [host ⇒ inverter]



ACK frame [inverter ⇒ host]



NAK frame [inverter ⇒ host]

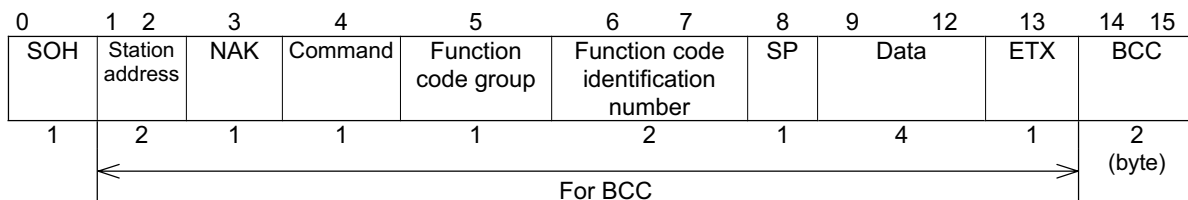


Table 4.2 Request frame

Byte	Field	Value		Description
		ASCII format	Hexadecimal format	
0	SOH	SOH	01 _H	Start of message
1	Station address	0 to 3, 9	30 _H to 33 _H 39 _H	Station address of the inverter (decimal: ten's figure)
2		0 to 9	30 _H to 39 _H	Station address of the inverter (decimal: one's figure)
3	ENQ	ENQ	05 _H	Transmission request
4	Command	R W A E	52 _H 57 _H 41 _H 45 _H	Request command Polling (read) Selecting (write) High-speed response selecting (write) *2 Alarm reset
5	Function code group *1	F E C P H A J y o s M W X Z	46 _H 45 _H 43 _H 50 _H 48 _H 41 _H 4A _H 59 _H 4F _H 53 _H 4D _H 57 _H 58 _H 5A _H	Function code group *3 Fundamental function Extension terminal function Control function of frequency Motor1 parameter High performance function Motor2 parameter Application function Link function Option function Command data Monitor data 1 Monitor data 2 Alarm data 1 Alarm data 2
6	Function code identification number *1	0 to 9	30 _H to 39 _H	Function code identification number (decimal: ten's figure)
7		0 to 9	30 _H to 39 _H	Function code identification number (decimal: one's figure)
8	Special additional data	SP	20 _H	Unused (space fixed)
9	Data	0 to F	30 _H to 3F _H	Data's first character (hexadecimal: thousand's figure)
10		0 to F	30 _H to 3F _H	Data's second character (hexadecimal: hundred's figure)
11		0 to F	30 _H to 3F _H	Data's third character (hexadecimal: ten's figure)
12		0 to F	30 _H to 3F _H	Data's fourth character (hexadecimal: one's figure)
13	ETX	ETX	03 _H	End of message
14	BCC	0 to F	30 _H to 3F _H	Checksum 1 (hexadecimal: ten's figure)
15		0 to F	30 _H to 3F _H	Checksum 2 (hexadecimal: one's figure)

*1 A space (SP = 20_H) will be set for an alarm reset command.

*2 Use high-speed response selecting to read the monitor when a command, which takes time for selecting (see Table 4.12 in "4.2 Host Side Procedures"), is written. The inverter does not respond to the regular write command W until writing is completed. With regard to high-speed response command A, the inverter sends back response upon receipt of a write request and communications can, therefore, continue even during writing. To confirm whether writing is completed in this case, read the BUSY flag (M14: 15 bits). If additional writing is performed during writing, NAK (error during writing) will result.

*3 Function codes are divided into function codes that can be edited from the keypad of the inverter, and communications dedicated function codes.

1) Function codes editable from the keypad

Fundamental function:	F code
Extension terminal function:	E code
Control function of frequency:	C code
Motor1 parameter:	P code
High performance function:	H code
Motor2 parameter:	A code
Application function:	J code
Link function:	y code
Option function:	o code

For details about the contents of the function code, see "Chapter 2, 2.4 Making RS-485-related Settings" and "Chapter 9 Function Codes" in User's Manual of each inverter type.

2) Communications dedicated function codes

Command data:	S code
Monitor data 1:	M code
Monitor data 2:	W code
Alarm data 1:	X code
Alarm data 2:	Z code

For further information about these codes, see "Chapter 5 Function Codes and Data Formats."

Table 4.3 ACK frame

Byte	Field	Value		Description
		ASCII format	Hexadecimal format	
0	SOH	SOH	01 _H	Start of message
1	Station address	0 to 3	30 _H to 33 _H	Station address of the inverter (decimal: ten's figure)
2		0 to 9	30 _H to 39 _H	Station address of the inverter (decimal: one's figure)
3	ACK	ACK	06 _H	Transmission response Acknowledgement: There was no receiving or logical error.
4	Command	R W A E	52 _H 57 _H 41 _H 45 _H	Answerback of request command Polling (read) Selecting (write) High-speed response selecting (write) Alarm reset
5	Function code group *1	F E C P H A J y o s M W X Z	46 _H 45 _H 43 _H 50 _H 48 _H 41 _H 4A _H 59 _H 4F _H 53 _H 4D _H 57 _H 58 _H 5A _H	Function code group Fundamental function Extension terminal function Control function of frequency Motor1 parameter High performance function Motor2 parameter Application function Link function Option function Command data Monitor data 1 Monitor data 2 Alarm data 1 Alarm data 2
6	Function code identification number *1	0 to 9	30 _H to 39 _H	Function code identification number (decimal: ten's figure)
7		0 to 9	30 _H to 39 _H	Function code identification number (decimal: one's figure)
8	Special additional data	SP -	20 _H 2D _H	Fixed to "sp (space)" normally. "-" for negative data
9	Data	0 to F	30 _H to 3F _H	Data's first character (hexadecimal: thousand's figure)
10		0 to F	30 _H to 3F _H	Data's second character (hexadecimal: hundred's figure)
11		0 to F	30 _H to 3F _H	Data's third character (hexadecimal: ten's figure)
12		0 to F	30 _H to 3F _H	Data's fourth character (hexadecimal: one's figure)
13	ETX	ETX	03 _H	End of message
14	BCC	0 to F	30 _H to 3F _H	Checksum 1 (hexadecimal: ten's figure)
15		0 to F	30 _H to 3F _H	Checksum 2 (hexadecimal: one's figure)

*1 A space (SP = 20_H) will be set for an alarm reset command.

Table 4.4 NAK frame

Byte	Field	Value		Description
		ASCII format	Hexadecimal format	
0	SOH	SOH	01 _H	Start of message
1	Station address	0 to 3	30 _H to 33 _H	Station address of the inverter (decimal: ten's figure)
2		0 to 9	30 _H to 39 _H	Station address of the inverter (decimal: one's figure)
3	NAK	NAK	15 _H	Transmission response Negative acknowledgement: There was a logical error in the request.
4	Command *1	R W A E	52 _H 57 _H 41 _H 45 _H	Answerback of request command Polling (read) Selecting (write) High-speed response selecting (write) Alarm reset
5	Function code group *1	F E C P H A J y o s M W X Z	46 _H 45 _H 43 _H 50 _H 48 _H 41 _H 4A _H 59 _H 4F _H 53 _H 4D _H 57 _H 58 _H 5A _H	Function code group Fundamental function Extension terminal function Control function of frequency Motor1 parameter High performance function Motor2 parameter Application function Link function Option function Command data Monitor data 1 Monitor data 2 Alarm data 1 Alarm data 2
6	Function code identification number *1	0 to 9	30 _H to 39 _H	Function code identification number (decimal: ten's figure)
7		0 to 9	30 _H to 39 _H	Function code identification number (decimal: one's figure)
8	Special additional data	SP	20 _H	Unused (space fixed)
9	Data	SP	20 _H	Unused (space fixed)
10		SP	20 _H	Unused (space fixed)
11		0 to F	30 _H to 3F _H	Communications error code higher order (hexadecimal: ten's figure)
12		0 to F	30 _H to 3F _H	Communications error code lower order (hexadecimal: one's figure)
13	ETX	ETX	03 _H	End of message
14	BCC	0 to F	30 _H to 3F _H	Checksum 1 (hexadecimal: ten's figure)
15		0 to F	30 _H to 3F _H	Checksum 2 (hexadecimal: one's figure)

*1 A space (SP = 20_H) will be set for a transmission format or transmission command error.

[2] Optional frame

This section describes the structure and meaning of each optional frame.

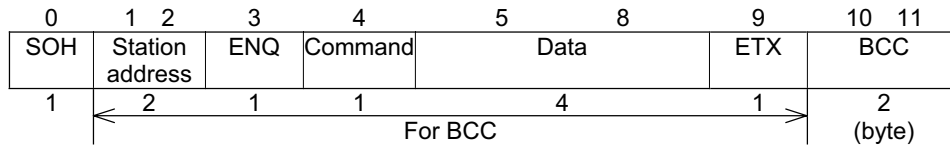
Selecting request frame [host ⇒ inverter]

Table 4.5 Selecting request frame

Byte	Field	Value		Description
		ASCII format	Hexadecimal format	
0	SOH	SOH	01 _H	Start of message
1	Station address	0 to 3, 9	30 _H to 33 _H 39 _H	Station address of the inverter (decimal: ten's figure)
2		0 to 9	30 _H to 39 _H	Station address of the inverter (decimal: one's figure)
3	ENQ	ENQ	05 _H	Transmission request
4	Command	a e f m	61 _H 65 _H 66 _H 6D _H	Request command Speed setting (S01) Frequency command (S05) Operation command (S06) Reset command (The data part is all zero)
5	Data	0 to F	30 _H to 3F _H	Data's first character (hexadecimal: thousand's figure)
6		0 to F	30 _H to 3F _H	Data's second character (hexadecimal: hundred's figure)
7		0 to F	30 _H to 3F _H	Data's third character (hexadecimal: ten's figure)
8		0 to F	30 _H to 3F _H	Data's fourth character (hexadecimal: one's figure)
9	ETX	ETX	03 _H	End of message
10	BCC	0 to F	30 _H to 3F _H	Checksum 1 (hexadecimal: ten's figure)
11		0 to F	30 _H to 3F _H	Checksum 2 (hexadecimal: one's figure)

Selecting response frame [inverter ⇒ host]

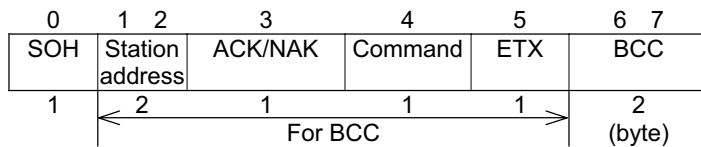


Table 4.6 Selecting response frame

Byte	Field	Value		Description
		ASCII format	Hexadecimal format	
0	SOH	SOH	01 _H	Start of message
1	Station address	0 to 3	30 _H to 33 _H	Station address of the inverter (decimal: ten's figure)
2		0 to 9	30 _H to 39 _H	Station address of the inverter (decimal: one's figure)
3	ACK/NAK	ACK	06 _H	Transmission response Acknowledgement: There was no receiving or logical error.
		NAK	15 _H	Negative acknowledgment: There was a logical error in the request.
4	Command	a	61 _H	Request command
		e	65 _H	Speed setting (S01)
		f	66 _H	Frequency command (S05)
		m	6D _H	Operation command (S06) Reset command
5	ETX	ETX	03 _H	End of message
6	BCC	0 to F	30 _H to 3F _H	Checksum 1 (hexadecimal: ten's figure)
7		0 to F	30 _H to 3F _H	Checksum 2 (hexadecimal: one's figure)

Polling request frame [host ⇒ inverter]

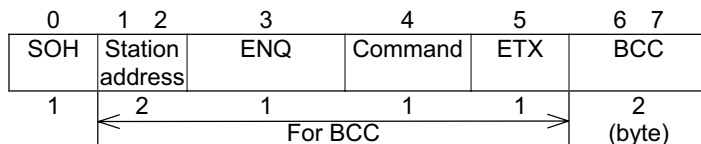


Table 4.7 Polling request frame

Byte	Field	Value		Description
		ASCII format	Hexadecimal format	
0	SOH	SOH	01 _H	Start of message
1	Station address	0 to 3	30 _H to 33 _H	Station address of the inverter (decimal: ten's figure)
2		0 to 9	30 _H to 39 _H	Station address of the inverter (decimal: one's figure)
3	ENQ	ENQ	05 _H	Transmission request
4	Command	g	67 _H	Request command
		j	6A _H	Actual frequency, actual speed (M06)
		k	6B _H	Output frequency monitor (M09)
		h	68 _H	Operation status monitor (M14) Torque monitor (M07) (not supported by FRENIC-Mini)
5	ETX	ETX	03 _H	End of message
6	BCC	0 to F	30 _H to 3F _H	Checksum 1 (hexadecimal: ten's figure)
7		0 to F	30 _H to 3F _H	Checksum 2 (hexadecimal: one's figure)

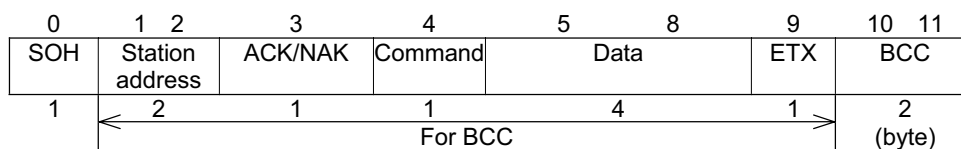
Polling response frame [inverter ⇒ host]

Table 4.8 Polling response frame

Byte	Field	Value		Description
		ASCII format	Hexadecimal format	
0	SOH	SOH	01 _H	Start of message
1	Station address	0 to 3	30 _H to 33 _H	Station address of the inverter (decimal: ten's figure)
2		0 to 9	30 _H to 39 _H	Station address of the inverter (decimal: one's figure)
3	ACK/NAK	ACK	06 _H	Transmission request Acknowledgement: There was no receiving or logical error.
		NAK	15 _H	Negative acknowledgment: There was a logical error in the request.
4	Command	g	67 _H	Request command Actual frequency, actual speed (M06) Output frequency monitor (M09) Operation status monitor (M14) Torque monitor (M07) (not supported by FRENIC-Mini)
		j	6A _H	
		k	6B _H	
		h	68 _H	
5	Data	0 to F	30 _H to 3F _H	Data's first character (hexadecimal: thousand's figure)
6		0 to F	30 _H to 3F _H	Data's second character (hexadecimal: hundred's figure)
7		0 to F	30 _H to 3F _H	Data's third character (hexadecimal: ten's figure)
8		0 to F	30 _H to 3F _H	Data's fourth character (hexadecimal: one's figure)
9	ETX	ETX	03 _H	End of message
10	BCC	0 to F	30 _H to 3F _H	Checksum 1 (hexadecimal: ten's figure)
11		0 to F	30 _H to 3F _H	Checksum 2 (hexadecimal: one's figure)

[3] NAK frame

When the response frame length is determined by the command type and the command type character is correctly identified, response will be given according to the frame length specified by the command in principle.

Table 4.9 Negative acknowledgment (NAK) frame

No.	Frame/Command type	Cause of error	NAK response frame	Error code (M26)
1	Standard frame Optional frame	The ENQ was not detected in the specified position.	Standard frame (16 bytes long)	Format error [74]
2	Selecting command (a, e, f, m)	The ETX was not detected in the specified position.	Optional frame (8 bytes long)	Format error [74]
3	Polling command (g, j, k, h)	The ETX was not detected in the specified position.	Optional frame (12 bytes long)	Format error [74]
4	Other than specified commands	A command other than the specified commands (R, W, A, E, a, e, f, g, j, k, h, m) was detected.	Standard frame (16 bytes long)	Command error [75]



When negative acknowledgment (NAK) for a format or command error is returned with the standard format as in the case of No. 1 and No. 4, the contents of the command type, function code group, and function code identification number fields will be undefined.

4.1.3 Descriptions of fields

[1] Command field

The table below shows command types. The applicable frame is different among the command types.

Table 4.10 Command formats

Command	Description	Applicable frame
ASCII R	Reads function code data (polling).	Standard frame
ASCII W	Writes function code data (selecting).	
ASCII A	Writes function code data at high speed (writing that does not wait for writing to be completed).	
ASCII E	Resets an alarm.	
ASCII a	Gives a frequency command (S01). *1	Optional frame
ASCII e	Gives a frequency command (S05). *1	
ASCII f	Gives an operation command (S06). *1	
ASCII g	Reads the output frequency (M06). *1	
ASCII h	Reads the torque monitor (M07). *1 (not supported by FRENIC-Mini)	
ASCII j	Reads the output frequency (M09). *1	
ASCII k	Reads the operation status monitor (M14). *1	
ASCII m	Resets an alarm.	

*1 The above commands "a" to "k" are used to read or write data in the function code data format specified in parentheses.

[2] Data field

Standard frame

8	9	10	11	12
Special additional data	Data's first character	Data's second character	Data's third character	Data's fourth character

Optional frame

9	10	11	12
Data's first character	Data's second character	Data's third character	Data's fourth character

All data, except for some special ones, are treated as 16 bits long. In the data field of the communications frame, data is hexadecimal (0000_H – FFFF_H), and each digit is represented by an ASCII code. Negative integer data (signed data) is treated as a complement of 2 of the integer data without the sign.



- The alphabetic characters A to F of hexadecimal data must be uppercase.
- Set 0 in all the data fields of the request frame for polling.
- In selecting, the data field of the ACK frame will be undefined.

(Example) When setting 20Hz with function code S01 (speed setting 1) (maximum frequency = 60Hz)

- 1) Calculate the set value according to the data format of S01 ($\pm 20000/\text{maximum frequency}$).

Data = $20\text{Hz} \times \pm 20000/60\text{Hz}$ (+ for forward rotation, – for reverse rotation)

= ± 6666.6

$\approx \pm 6667$

- 2) Convert the data into hexadecimal (a complement of 2 in the case of negative data).

Data = 6667 (forward rotation)

= $1A0B_H$

Data = -6667 (reverse rotation)

= $0 - 6667$

Thus,

$65536 - 6667 = 58869 = E5F5_H$

- 3) Set the data.

Position	Set value (forward rotation)	Set value (reverse rotation)
Data's first character	ASCII 1	ASCII E
Data's second character	ASCII A	ASCII 5
Data's third character	ASCII 0	ASCII F
Data's fourth character	ASCII B	ASCII 5

[3] Checksum field

The data in this field is intended to check whether there is any error in the communications frame at the time of data transmission. Calculate the data by adding one byte to all fields, except for S0H and the checksum field, treating the last byte of the result as a two-digit hexadecimal value, and converting each digit into an ASCII code.

(Example) When the result of addition is 0123_H

Position	Set value (forward rotation)
Checksum 1	ASCII 2
Checksum 2	ASCII 3

4.1.4 Communications examples

Typical communications examples are shown below (the station number is 12 in all cases):

[1] Standard frame

(Example 1) Selecting S01: speed setting 1 (write)

10Hz command x 20,000/maximum frequency 50Hz = 4000d = 0FA0_H

Request frame (host ⇒ inverter)

SOH	1	2	ENQ	W	S	0	1	SP	0	F	A	0	ETX	7	D
-----	---	---	-----	---	---	---	---	----	---	---	---	---	-----	---	---

ACK frame (inverter ⇒ host)

SOH	1	2	ACK	W	S	0	1	SP	0	F	A	0	ETX	7	E
-----	---	---	-----	---	---	---	---	----	---	---	---	---	-----	---	---

NAK frame (inverter ⇒ host) ... Link priority error

SOH	1	2	NAK	W	S	0	1	SP	SP	4	C	0	ETX	8	D
-----	---	---	-----	---	---	---	---	----	----	---	---	---	-----	---	---

(Example 2) Polling of M09: output frequency (read)

Request frame (host ⇒ inverter)

SOH	1	2	ENQ	R	M	0	9	SP	0	0	0	0	ETX	5	3
-----	---	---	-----	---	---	---	---	----	---	---	---	---	-----	---	---

ACK frame (inverter ⇒ host)

SOH	1	2	ACK	R	M	0	9	SP	0	B	B	8	ETX	8	0
-----	---	---	-----	---	---	---	---	----	---	---	---	---	-----	---	---

[2] Optional frame

(Example 1) Selecting of operation command (write)

Request frame (host ⇒ inverter) ... FWD command

SOH	1	2	ENQ	f	0	0	0	1	ETX	9	2
-----	---	---	-----	---	---	---	---	---	-----	---	---

ACK frame (inverter ⇒ host)

SOH	1	2	ACK	f	ETX	D	2
-----	---	---	-----	---	-----	---	---

NAK frame (inverter ⇒ host)

The cause of the error can be confirmed with function code M26 (transmission error transaction code).

SOH	1	2	NAK	f	ETX	E	1
-----	---	---	-----	---	-----	---	---

(Example 2) Selecting of operation command in broadcast (write)

Request frame (host ⇒ inverter) ... REV command

SOH	9	9	ENQ	f	0	0	0	2	ETX	A	2
-----	---	---	-----	---	---	---	---	---	-----	---	---

The inverter does not respond to broadcast.

Table 4.11 ASCII code table

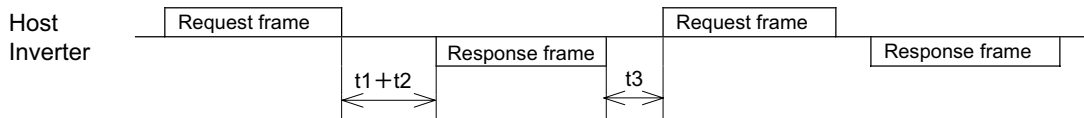
	00 _H	10 _H	20 _H	30 _H	40 _H	50 _H	60 _H	70 _H
0 _H	NUL	DLE	SP	0	@	P	`	p
1 _H	SOH	DC1	!	1	A	Q	a	q
2 _H	STX	DC2	“	2	B	R	b	r
3 _H	ETX	DC3	#	3	C	S	c	s
4 _H	EOT	DC4	\$	4	D	T	d	t
5 _H	ENQ	NAK	%	5	E	U	e	u
6 _H	ACK	SYN	&	6	F	V	f	v
7 _H	BEL	ETB	‘	7	G	W	g	w
8 _H	BS	CAN	(8	H	X	h	x
9 _H	HT	EM)	9	I	Y	i	y
A _H	LF	SUB	*	:	J	Z	j	z
B _H	VT	ESC	+	;	K	[k	{
C _H	FF	FS	,	<	L	\	l	
D _H	CR	GS	-	=	M]	m	}
E _H	SO	RS	.	>	N	-	n	~
F _H	SI	US	/	?	O	-	o	DEL

The shaded codes are used for this communications protocol.

4.2 Host Side Procedures

4.2.1 Inverter's response time

Upon receipt of a query request from the host, the inverter executes the requested command, and sends back response after the response time shown below:



$t1 + t2$: Inverter's response time

$t1$: Response interval time (function code: y09)

The time until the inverter starts to send response to the request from the host can be set. Setting the response interval time enables even the host side with a slow transaction execution speed to adjust timing.

$t2$: Inverter's transaction time

This is the time until the inverter executes the request and sends back response as shown in Table 4.12 below.

$t3$: See "4.2.3 Receiving preparation complete time and message timing from the host."

Table 4.12 Inverter's transaction time

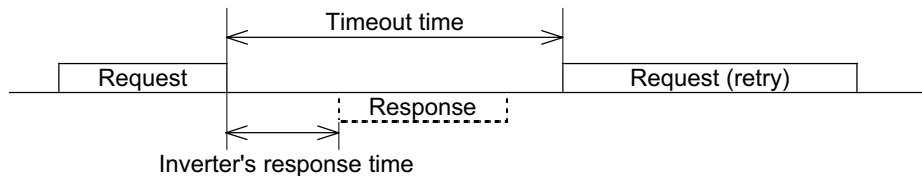
Command	Transaction	Description	$t2$	Timeout time (recommended)
R	Function code read data		$\leq 10\text{ms}$	0.1 sec
W	Function code write data	S code commands other than S08 or S09	$\leq 10\text{ms}$	0.1 sec
		H03 = 2: Motor parameter initialization	$\leq 500\text{ms}$	1.0 sec
		H03 = 1: Data initialization	$\leq 5\text{s}$	10.0 sec
		Function code other than above	$\leq 100\text{ms}$	0.5 sec
A	Function code data high-speed writing		$\leq 10\text{ms}$	0.1 sec
E, m	Alarm reset		$\leq 10\text{ms}$	0.1 sec
a, e, f	Specific function code write data		$\leq 10\text{ms}$	0.1 sec
g, h, j, k	Specific function code read data		$\leq 10\text{ms}$	0.1 sec

4.2.2 Timeout processing

To read/write data from/to the host, transmit the next frame after confirming response. If response is not transmitted from the inverter for more than a specified period of time (timeout time), it is a timeout, and perform a retry. (If a retry begins before a timeout, the requested frame cannot be received properly.)

The timeout time must be set longer than the response time of the inverter. Table 4.12 above mentioned shows recommended timeout times when no response interval time is set.

In case of a timeout, retransmit the same frame or perform polling (M26) for reading details of an error to confirm whether the inverter sends back normal response. If normal response is returned, this indicates that some transient transmission error occurred due to noise or other reasons, and subsequent communications is normal. (However, if this phenomenon frequently occurs even when normal response is sent back, some problem may exist. Perform a close investigation.) In case of no response, perform another retry. If the number of retries exceeds the set value (generally about three times), there may be a problem with the hardware and the software for the host controller. Investigate and correct the cause.



4.2.3 Receiving preparation complete time and message timing from the host

The time from the return of response by the inverter to the completion of receiving preparation of the communications port (switching from transmission to receiving) is called a receiving preparation complete time.

Transmit the following messages after the receiving preparation complete time:

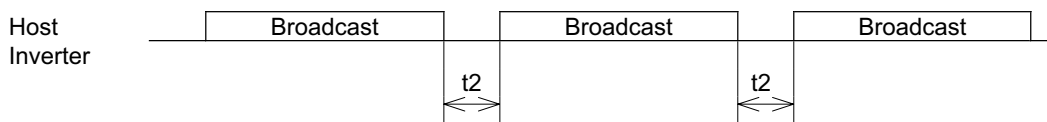
Receiving preparation complete time: 5ms or less

Message timing from the host (t_3): $t_3 > 5\text{ms}$

In the case of broadcast

Upon receipt of a request for a query message from the host by broadcast, the inverter executes the command and enters the receiving enabled status.

Transmit the next message from the host following broadcast after the transaction time (t_2) of the inverter.



4.3 Communications Errors

4.3.1 Categories of communications errors

The communications-related errors the inverter detects are listed below:

Table 4.13 Communications errors detected by inverter

Error category	Error name	Description	Error code (M26)
Transmission error	Checksum error	The frame to the local station is found unmatched in checksum collation.	71(47 _H)
	Parity error	The parity is unmatched.	72(48 _H)
	Other errors	Receiving errors other than the abovementioned (framing error, overrun error)	73(49 _H)
Logical error	Format error	- The characters of the transmission request are incorrect. - The last character of the message is not in the specified position.	74(4A _H)
	Command error	A command that does not exist was transmitted.	75(4B _H)
	Link priority error (does not occur for FRENIC-Mini)	A frequency command, PID command, or change command of the run command (writing request to S01, S05, S06, and S13) are sent through the communication route other than that specified with H30.	76(4C _H)
	Function code error	A function code that does not exist was requested.	78(4E _H)
	Write disabled error	An attempt was made during operation to write the function code for write disabled or for write disabled during operation.	79(4F _H)
	Data error	The write data is beyond the writable range.	80(50 _H)
	Error during writing	An attempt was made to write another function data during function writing with command A.	81(51 _H)
Communications disconnection error	Communications disconnection error	The inverter did not receive a normal frame addressed to local station or to other stations within the communications disconnection detection time set with the function code.	—

Transmission error (error codes 71 to 73)

When a transmission error occurs eight straight times, it is handled as a communications error. However, the inverter does not return response in order to avoid overlapping of response from multiple inverters. The count of eight straight times will be cleared upon normal receipt of a frame to another station or to the local inverter (station) itself.

Logical error (error codes 74 to 81)

When a logical error is detected, a negative acknowledgment (NAK) frame reports it. For further information, see the NAK response of each frame.

Communications disconnection error

If the inverter in operation does not receive a normal frame to itself (local station) or to another station when it has received a normal frame more than once and is operating via communications (frequency command or operation command), this status is considered disconnected.

When a disconnection status is set and remains over the setting time of function code y08, y18 (communications disconnection detection time), it is treated as a communications error.

- 1) Communications disconnection detection time (y08, y18): 0 (without detection), 1 to 60 (seconds)
- 2) Condition to clear communications disconnection detection timer: It will be cleared in a status other than disconnection.

When it is necessary to take action against errors by factor, the factor can be identified by reading M26. (M26 stores the latest communications error codes.)

4.3.2 Operations in case of communications errors

Operations in case of a transmission or communications disconnection error are the same as those of the Modbus RTU protocol. See "3.3.2 Operations in case of errors" in Chapter 3 Modbus RTU Protocol.