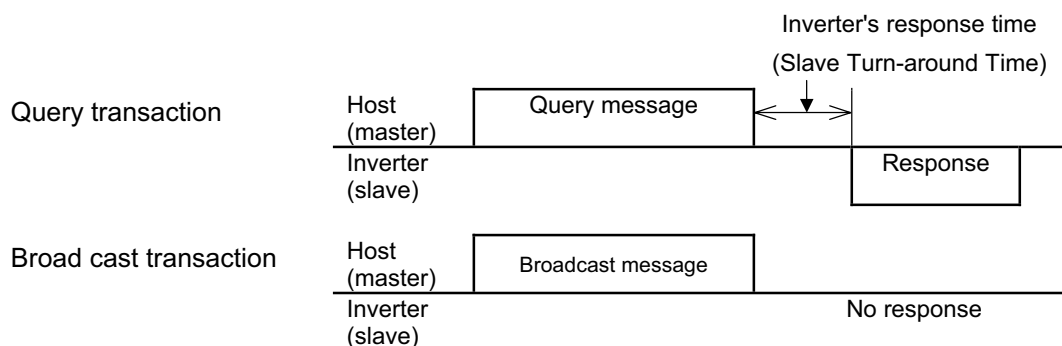


3.1 Messages

3.1.1 Message formats

The regular formats for transmitting RTU messages are shown below:



If the inverter receives from the host a message in the standby status and considers it properly received, it executes a transaction in response to the request and sends back normal response. If the inverter judges that the message has not been received properly, it returns error response. The inverter does not send back any response in the case of broadcast transactions.

3.1.2 Message types

Message types are classified into four types; query, normal response, error response, and broadcast.

Query

The host sends messages to an inverter.

Normal response

After the inverter received a query from the host, the inverter executes a transaction in response to the request, and sends back corresponding normal response.

Error response

If the inverter receives a query but cannot execute the requested function because an invalid function code is specified or for other reasons, it sends back error response.

The error response is accompanied by a message describing the reason the request cannot be executed.

The inverter cannot send back any response in case of a CRC or physical transmission error (parity error, framing error, overrun error).

Broadcast

The master uses address 0 to send messages to all slaves. All slaves, which receive a broadcast message, execute the requested function. This transaction will be terminated upon timeout of the master.

3.1.3 Message frames

As shown below, a transmission frame consists of four blocks, which are called fields. Details depend on FC (RTU function codes). To make a clear distinction between RTU function codes and the inverter's function codes, the former will be hereinafter referred to as 'FC'.

1 byte	1 byte	Up to 105 bytes	2 bytes
Station address	FC (RTU function code)	Information	Error check

Station address

The station address field is one byte long, in which a station address between 0 and 247 can be selected.

Selecting address 0 means the selection of all slave stations and a broadcast message.

'FC' (RTU function code)

The 'FC' field is one byte long, in which a function code is defined with a number from 0 to 255. The 'FCs' marked with "*" are available. Do not use any unavailable 'FC'. Failure to observe this rule results in error response.

Table 3-1 List of 'FC'

'FC'	Description	Support		
		Mini	Eco	Multi
0	Unused	-	-	-
1 *	Read Coil Status (80 coils maximum)	×	○	○
2	Unused	-	-	-
3 *	Read Holding Registers (50 registers maximum)	○	○	○
4	Unused	-	-	-
5 *	Force Single Coil	×	○	○
6 *	Preset Single Register	○	○	○
7	Unused	-	-	-
8 *	Diagnostics	○	○	○
9 to 14	Unused	-	-	-
15 *	Force Multiple Coils (16 coils maximum)	×	○	○
16 *	Preset Multiple Registers (50 registers maximum)	○	○	○
17 to 127	Unused	-	-	-
128 to 255 *	Reserved for exception response	-	-	-

Information

The information field contains all information (function code, byte count, number of data, data, etc.). For further information about the information field for each message type (broadcast, query, normal response, error response), see "3.1.4 Message categories."

Error check

The error check field is a CRC-16 check system and two bytes long. Since the length of the information field is variable, the frame length required for calculating the CRC-16 code is calculated based on the 'FC' and the byte count data.

For further information about CRC-16 calculations and algorithm, see "3.4 CRC-16."

For byte counts, see "3.1.4 Message categories."

Character format

Each byte of a message is transmitted as a character. Character formats are described on the following page.

A character comprises a start bit (logical value 0), 8-bit data, an additional (optional) parity bit, and a stop bit (logical value 1).

A character always consists of eleven bits, and the number of stop bits varies depending on whether parity exists.

Without parity

LSB											MSB
0	1	2	3	4	5	6	7	8	9	10	
Start	Data								Stop		

With parity

LSB											MSB
0	1	2	3	4	5	6	7	8	9	10	
Start	Data								Parity (optional)	Stop	



- Modbus RTU protocol has the above character format as specified by the rule. But, some devices use the format "No parity + Stop bit: 1 bit." For connection with these devices, FRENIC- Multi supports the parity bit selection: y06, y16=3. When y06, y16=3 are selected, the protocol is given the following character format.

LSB										MSB
0	1	2	3	4	5	6	7	8	9	
Start	Data								Stop	

3.1.4 Message categories

There are eight RTU message categories; read holding registers, preset single register, preset multiple registers, diagnostics, read coil status, force single coil, force multiple coils and error response.

Each category is described below:

[1] Read holding registers

Query

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	03 _H	Function code	Number of read data	Error check
		Hi Lo	Hi Lo	

Normal response

1 byte	1 byte	1 byte	2 to 100 bytes	2 bytes
Station address	03 _H	Byte count	Read data	Error check
			Hi, Lo (data 0); Hi, Lo (data 1);	

How to set a query

- This request is not available for broadcast transactions. Station address 0 will become invalid (no response).
- 'FC' = 3 (03_H)
- The function code is two bytes long. The Hi byte indicates the function code group (see Table 3.2), and the Lo byte represents a function code identification number (0 to 99).

(Example) When the function code is E15, the Hi byte is 01_H and the Lo byte is 0F_H.

Table 3.2 Function code group/code conversion table

Group	Code		Name	Group	Code		Name
F	0	00 _H	Fundamental function	o	6	06 _H	Operational function
E	1	01 _H	Extension terminal function	M	8	08 _H	Monitor data
C	2	02 _H	Control function of frequency	J	13	0D _H	Application function
P	3	03 _H	Motor1 parameter	y	14	0E _H	Link function
H	4	04 _H	High performance function	W	15	0F _H	Monitor 2
A	5	05 _H	Motor2 parameter	X	16	10 _H	Alarm 1
s	7	07 _H	Command/Function data	Z	17	11 _H	Alarm 2

- The length of the read data is up to 50 words (2 byte each).
- If the read data contains an unused function code, 0 will be read, which will not result in an error.
- Data does not extend over two or more function code groups. If, for example, reading of 40 words is specified from F40 but only function codes up to F40 are available, the data of F40 will be set at the first word, and the other 49 words will be 0.

Interpretation of normal response

- The data range of byte counts is between 2 and 100. A byte count is double the number of read data (1 - 50 data) of the response.
- The read data contains each word data in order of Hi byte and Lo byte, and each word data is sent back in order of the data of the function code (address) requested by the query, the data of that address number plus 1, the data of that number address number plus 2 ... If two or more function data are read and the second or any of the following data contains an unused function code (F09, etc.), the read data will become 0.

[2] Preset single register**Query**

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	06 _H	Function code	Write data	Error check
		Hi Lo	Hi Lo	

Normal response

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	06 _H	Function code	Write data	Error check

How to set a query

- When address 0 is selected, broadcast is available. In this case, all inverters do not respond even if a broadcast request is executed.
- 'FC' = 6 (06_H)
- The function code is two bytes long. The Hi byte indicates the function code group (see Table 3.2), and the Lo byte represents a function code identification number (0 to 99).
- The written data field is fixed two bytes long. Set the data on the function code to be written.

Interpretation of normal response

The frame is the same as the query.

[3] Preset multiple registers**Query**

1 byte	1 byte	2 bytes	2 bytes	1 byte	2 to 100 bytes	2 bytes
Station address	10 _H	Function code	Number of write data	Byte count	Write data	Error check
		Hi Lo	Hi Lo		Hi, Lo; Hi, Lo...	

Normal response

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	10 _H	Function code	Number of write data	Error check

How to set a query

- When the station address 0 is selected, broadcast is available. In this case, all inverters do not respond even if a broadcast request is executed.
- 'FC' = 16 (10_H)
- The function code is two bytes long. The Hi byte indicates the function code group (see Table 3.2), and the Lo byte represents a function code identification number (0 to 99).
- The number of write data is two bytes long, and the setting range is from 1 to 50. If 51 or a higher value is set, error response will result.
- The byte count field is one byte long, and the setting range is from 2 to 100. Set a value equivalent to the double of the number of write data.
- Set the lowest order code (the data on the function code requested by the query) at the first two bytes of the write data, and the higher order data (address plus 1, address plus 2 ...) at the following bytes.
- If the write data contains an unused function code, the writing will be ignored, which will not result in an error.

Interpretation of normal response

- With regard to the function code and the number of write data, the same values as those of the query will be sent back.

[4] Diagnostics

Query

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	08 _H	Sub function code 0000 _H	Write data	Error check
		Hi Lo	Hi Lo	

Normal response

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	08 _H	Sub function code 0000 _H	Write data	Error check

How to set a query

- This request cannot use broadcast. Station address 0 will become invalid (no response).
- 'FC' = 8 (08_H)
- Set the sub function code field to be 2 bytes long fixed 0000_H. Error response will result if data other than 0000_H is set.
- The write data field is two bytes long, and any contents of data can be set.

Interpretation of normal response

- The frame is the same as the query.

[5] Read coil status (not supported by FRENIC-Mini)

Query

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	01 _H	Coil address	No. of coils	Error check
		Hi Lo	Hi Lo	

Normal response

1 byte	1 byte	1 byte	1 to 10 bytes	2 bytes
Station address	01 _H	Byte count	Read data	Error check

How to set a query

- Broadcast with station address 0 is not usable. If this address is used, no response is returned.
- 'FC'=1 (01_H)
- Read out a coil (bit data) by specifying the top address of the coil to be read out and the number of points read out (number of coils).
- For the assignment of a coil (bit data), see table 3.3. For each content, refer to the S and M codes in the remarks column.

Table 3.3 Description of coil (bit data)

Coil number	+7	+6	+5	+4	+3	+2	+1	+0	Remarks
1	X6	X5	X4	X3	X2	X1	REV	FWD	S06: Run operation command (Read/Write)
9	RST	XR	XF	-	-	X9	X8	X7	
17	VL	TL	NUV	BRK	INT	EXT	REV	FWD	M14: Run status (Read only)
25	BUSY	WR		RL	ALM	DEC	ACC	IL	
33	FAN	KP	OL	IPF	SWM2	RDY	FDT	FAR	M70: Run status 2 (Read only)
41	-	-	IDL	ID	OPL	LIFE	OH	TRY	
49	X6	X5	X4	X3	X2	X1	REV	FWD	M13: Run operation command (final command) (Read only)
57	RST	XR	XF	-	-	X9	X8	X7	
65	-	-	-	Y5	Y4	Y3	Y2	Y1	M15: General-purpose output terminal information (Read only)
73	-	-	-	-	-	-	-	30	

- The "-" symbols in the table mean that the bit is reserved and always zero.
- Coil addresses are 0 to 79, calculated by subtracting one from coil numbers. If a coil address is 80 or more, an error occurs because of an incorrect address.
- The number of coils is 1 to 80. If the number of coils exceeds the range, an error occurs because of an incorrect address.
- No error occurs even if the sum of the numbers of coil addresses and coils exceeds the coil range.

Interpretation of normal response

- Data are stored from the LSB (the rightmost bit in the table above) in ascending order of coil number. When a coil is turned on, the data becomes one, and all the remaining bits are changed to zero.
- The byte length of the read data is filled in the byte count field.
- For a data example, see table 3.4.

Table 3.4 Example of coil address = 13 and the number of coils = 9

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Data's 1st byte	BRK	INT	EXT	REV	FWD	RST	XR	XF
Data's 2nd byte	0	0	0	0	0	0	0	NUV

[6] Force single coil (not supported by FRENIC-Mini)

Query

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	05 _H	Coil address	Data	Error check
		Hi Lo	Hi Lo	

Normal response

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	05 _H	Coil address	Data	Error check

How to set a query

- Broadcast with station address 0 is not usable. If used, no response is returned.
- 'FC' = 5 (05_H)
- Turn on/off a coil (bit data) by specifying only a bit.
- For the assignment of a coil (bit data), see table 3.5. For each content, refer to the S and M codes in the remarks column.

Table 3.5 Description of coil (bit data)

Coil number	+7	+6	+5	+4	+3	+2	+1	+0	Remarks
1	X6	X5	X4	X3	X2	X1	REV	FWD	S06: Run operation command (Read/Write)
9	RST	XR	XF	-	-	X9	X8	X7	

- The "-" symbol in the table means that the bit is reserved, and writing is ignored.
- The coil address is 0 to 15, calculated by subtracting one from the coil number. If a coil address is 16 or more, an error occurs because of an incorrect address.
- When a coil is turned off, data are 0000_H. When a coil is turned on, data are FF00_H.

Interpretation of normal response

- The format of normal response is the same as that of inquiry.
- No response is returned to the broadcast command.

[7] Force multiple coils (not supported by FRENIC-Mini)

Query

1 byte	1 byte	2 bytes	2 bytes	1 byte	1 to 2 bytes	2 bytes
Station address	0F _H	Coil address	No. of coils	Byte account	Write data	Error check
		Hi Lo	Hi Lo		Hi Lo	

Normal response

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Station address	0F _H	Coil address	No. of coils	Error check
		Hi Lo	Hi Lo	

How to set a query

- Broadcast with station address 0 is not usable. If is used, no response is returned.
- 'FC' = 15 (0F_H)
- Write a coil (bit data) by specifying the top address of the coil to be written, the number of points written (number of coils), and data to be written.
- For the assignment of a coil (bit data), see table 3.6. For each content, refer to the S and M codes in the remarks column.

Table 3.6 Description of coil (bit data)

Coil number	+7	+6	+5	+4	+3	+2	+1	+0	Remarks
1	X6	X5	X4	X3	X2	X1	REV	FWD	S06: Run operation command (Read/Write)
9	RST	XR	XF	-	-	X9	X8	X7	

- The "-" symbol in the table means that the bit is reserved and always zero.
- The coil address is 0 to 15, calculated by subtracting one from the coil number. If a coil address is 16 or more, an error occurs because of an incorrect address.
- If the byte count is 0 or 3 or more, an error occurs because of an incorrect data.
- The number of coils is 1 to 16. If 0 or 17 or more, an error occurs because of an incorrect address.
- No error occurs even if the coil address plus number of coils exceeds the coil range.
- If the number of coils is 9 or more and the byte count is 1 or less, an error occurs because of an incorrect data.
- If the number of coils is 8 or less and the byte count is 2, no error occurs.
- Data are stored from the LSB (the rightmost bit in the table above) in ascending order of coil number. When a coil is turned on, the data becomes one. When a coil is turned off, the data becomes zero. All the remaining bits are ignored.
- The byte count field indicates the byte length of the write data.
- For a data example, see table 3.7.

Table 3.7 Example of coil address = 2 and the number of coils = 9

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Data's 1st byte	X8	X7	X6	X5	X4	X3	X2	X1
Data's 2nd byte	0	0	0	0	0	0	0	X9

Interpretation of normal response

- The forms of coil address and number of coils are the same as the forms of query.
- No response is returned to the broadcast command.

[8] Error response

If the inverter receives an improper query, it will not execute it, which will result in error response.

Error response

1 byte	1 byte	1 byte	2 bytes
Station address	Exception function	Subcode	Error check

Interpretation of error response

- The station address is the same as that of the query.
- The exception function is a value obtained by adding 80_H to the 'FC' of the query message (or the value of the 'FC' if the 'FC' is larger than 80_H).
For example, when the 'FC' is 3, the exception function is 3 + 128 = 131 (83_H).
- The subcode represents the code of the reason for the improper query.

Table 3.8 Subcodes

Subcode	Item		Description
1	Improper 'FC'		FRENIC-Mini: A 'FC' other than 3, 6, 8, or 16 has been received. FRENIC-Eco: A 'FC' other than 1, 3, 5, 6, 8, 15, or 16 has been received.
2	Improper address	Improper function code	An unused function code or a function code out of range was received. When the read/write data (except the first one) containing an unused function code. - In Read holding registers Zero (0) will be read, which will not result in an error. - In preset multiple registers The writing will be ignored, which will not result in an error.
		Improper number of data	- When the number of read/write data is not between 1 and 50. - No error will result when the value of the function code plus the number of data is beyond the setting range of the function code.
		Subfunction code error (diagnostics)	A value other than 0 was received although the subfunction code as the diagnostics was fixed to 0.
3	Improper data	Data range error	The write data is beyond the permissible write range.
7	NAK	No right of writing	FRENIC-Mini: This error does not occur. FRENIC-Eco/Multi: No right of writing by H30/y98/y99
		Write disable	- Writing was attempted to the functions to which writing from RTU is prohibited or to which writing is disabled during operation. - Writing was attempted to a function code (other than S01, S05, S06, S13, and S14) that could not be written when the voltage was insufficient.

- If response is sent back to an improper query, a subcode will be set in an error code (that can be referred to with M26).

3.1.5 Communications examples

Typical communications examples are shown below (the station address is 5 in all cases).

(Example 1) M06: Output frequency 1 will be read.

Query (host ⇒ inverter)

05	03	08	06	00	01	67	EF
----	----	----	----	----	----	----	----

Normal response (inverter ⇒ host)

05	03	01	27	10	A3	B8
----	----	----	----	----	----	----

The detected speed value is 2710_H , or 10000_d . The actual frequency is 30 Hz according to the expression shown below:

$$10000 \times \frac{\text{Maximum frequency}}{20000} = 30 \text{ (Hz)} \quad (\text{Maximum frequency: 60 Hz})$$

(Example 2) S01: The value of 15Hz will be written to frequency command (maximum frequency: 60 Hz).

According to the expression shown below, the value to be written is 1388_H .

$$15\text{Hz} \times \frac{20000}{60 \text{ (Hz)}} = 5000_d = 1388_H$$

Query (host ⇒ inverter)

05	06	07	01	13	88	D5	AC
----	----	----	----	----	----	----	----

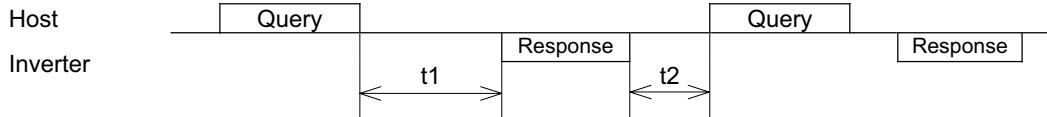
Normal response (inverter ⇒ host)

05	06	07	01	13	88	D5	AC
----	----	----	----	----	----	----	----

3.2 Host Side Procedures

3.2.1 Inverter's response time

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



t1: Response interval time

The response interval time is the longest time out of the time setting by a function code, 3-character time, or inverter's processing time.

- (1) y09/y19: setting of response interval time
0.00-1.00(s), factory shipment setting: 0.01(s)

You can set the time from receiving a request issued from a host to starting to send a response. By setting a response interval time, even the host side which is slower than the inverter can meet timing.

- (2) 3-character time (maximum value)

Table 3.9 3-character time (maximum time)

Baud rate (bps)	2400	4800	9600	19200	38400 (Not supported by FRENIC-Mini)
3-character time (ms)	15	10	5	5	5

- (3) Inverter processing time (The data volume shown below indicates the number of words.)

- 1) Read holding registers, read coil status, multiple read holding registers

Table 3.10 Inverter processing time

Data count	Inverter processing time (minimum to maximum)
1 to 7	5 to 10 (ms)
8 to 16	10 to 15 (ms)
n	$\text{Int}((n-1)/8) \times 5$ to $\text{int}((n-1)/8) \times 5 + 5$ (ms)

2) Preset single register, preset multiple registers, force single coil, and force multiple coils

Table 3.11 Inverter processing time

Data count	Inverter processing time (minimum to maximum)
1	25 to 30 (ms)
2	45 to 50 (ms)
3	65 to 70 (ms)
4	85 to 90 (ms)
n	$n \times 20 + 5$ to $n \times 20 + 10$ (ms)

However, the inverter processing time is 5 (s) maximum when H03=1 is written, and 500(ms) maximum when H03=2 is written in P02.

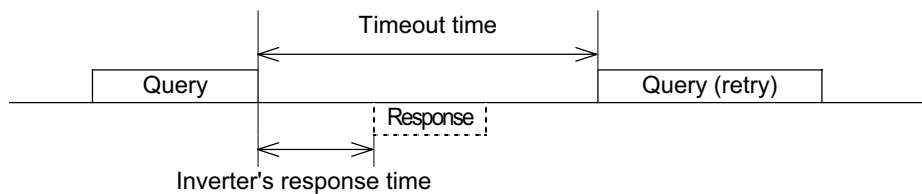
3) Maintenance code: 10 (ms)

t2: See section 3.2.3 "Receiving preparation complete time and message timing from the host."

3.2.2 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 time elapses, the requested frame cannot be received properly.)

The timeout time must be set longer than the response time of the inverter. In case of a timeout, retransmit the same frame or read details of the error (M26) 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 for 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 of the host. Investigate and correct the cause.



3.2.3 Receiving preparation complete time and message timing from the host

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

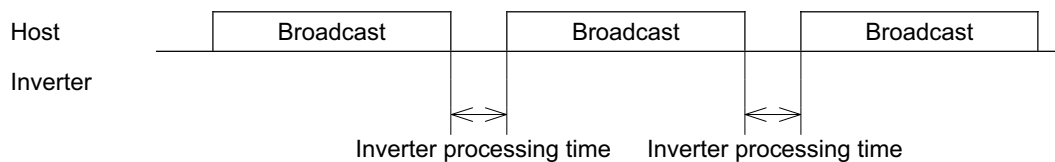
Transmit the following messages after the receiving preparation complete time:

Receiving preparation complete time: 3-character time

In the case of broadcast

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

When sending a message from the host after broadcast is performed, send the message after the inverter processing time shown in section 3.2.1 "Inverter response time" has passed.

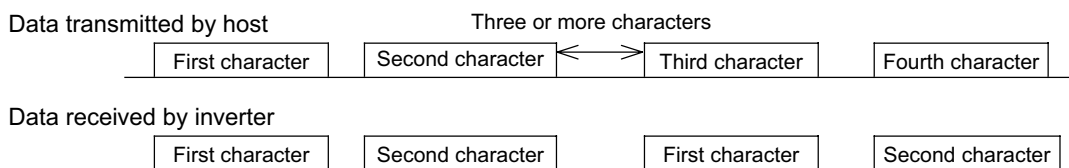


3.2.4 Frame synchronization method

Since the RTU transmits and receives binary data without using header characters for frame synchronization, a frame synchronization system is defined as a time without data to identify the head of the frame.

If data communications does not occur for a period equal to three characters (33 bits including the start and stop bits) at the current transmission speed during receiving standby, initialize the frame information, and consider the first received data the first byte of the frame. If a character interval reaches the length of three characters or more while a frame is received, the frame is discarded.

For this reason, the host must transmit data at a time interval of three or less characters between two characters.



With regard to data to another station, messages from the host and response from that station will be received. In response transmission to identify the head of the frame, a waiting time of three characters (33 bits including the start and stop bits) is required between the completion of data receipt by the station and the start of transmission.

Any devices multidropped also requires such a waiting time.

3.3 Communications Errors

3.3.1 Categories of communications errors

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

Table 3.12 Communications errors detected by inverter

Error category	Error name	Description	Error code (M26)
Logical error	Improper 'FC'	See "Table 3.8 Subcodes" shown in 3.1.4 [8].	1(01 _H)
	Improper address		2(02 _H)
	Improper data		3(03 _H)
	NAK		7(07 _H)
Transmission error	CRC error	The frame to the local station is found unmatched in CRC 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)
Communications disconnection error	Communications disconnection error	The inverter did not receive a normal frame addressed to local or to other stations within the communications disconnection time set with the function code.	—

Logical error (error codes 1 to 7)

When a logical error is detected, an error response frame reports it. For further information, see "3.1.4 [8] Error response."

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.

Communications disconnection error

If the inverter in operation does not receive a normal frame to itself or to other stations 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.

If the status of disconnection continues for the communication disconnection time set up by function code (y08, y18), error processing is performed as a communication 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.

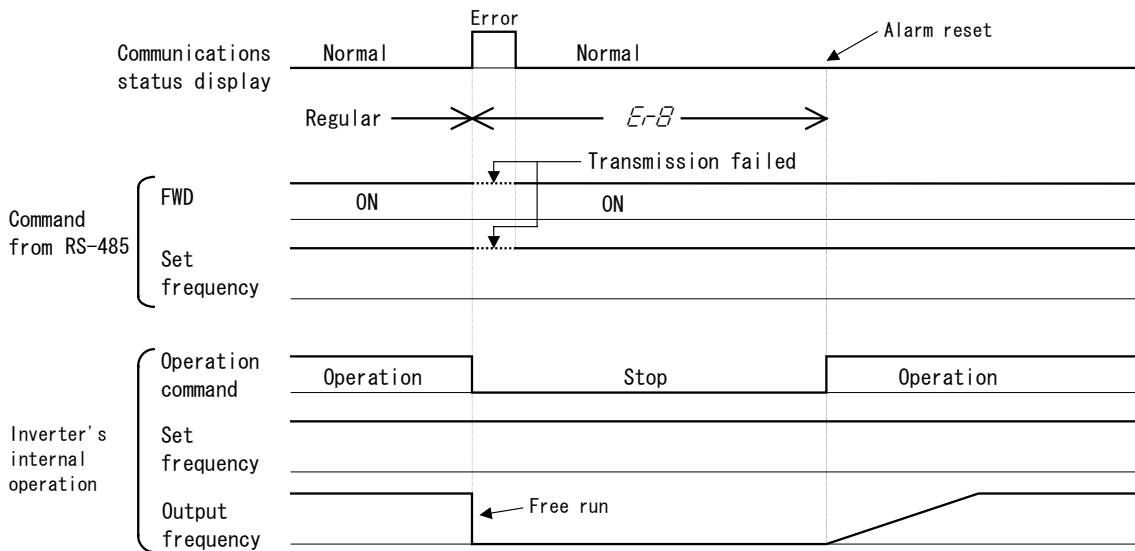
When it is necessary to take action against errors by factor, the factor can be identified by reading M26. (M26 stores the latest communications error codes.)

3.3.2 Operations in case of errors

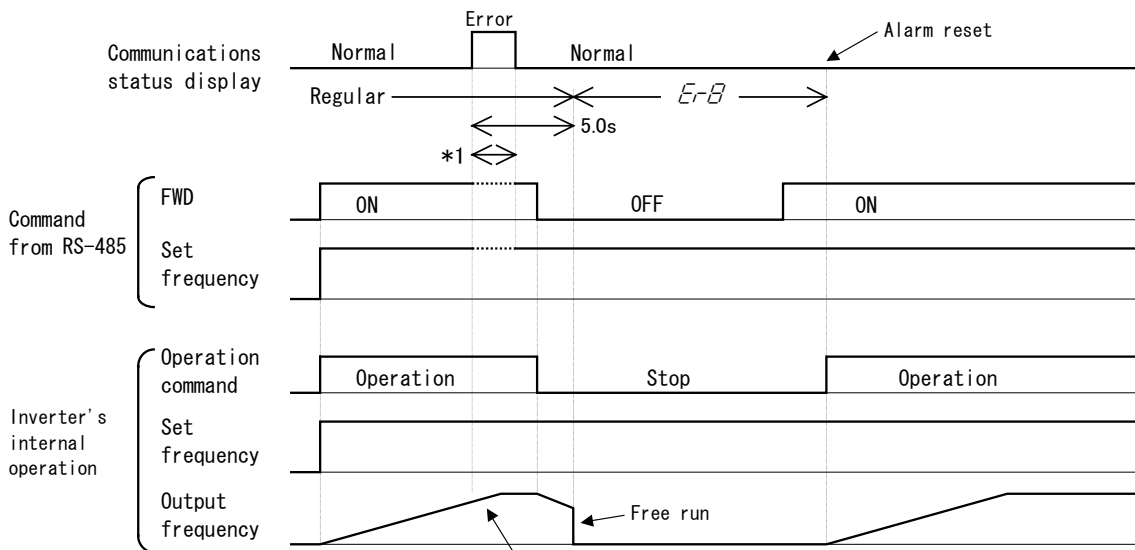
The action when a transmission or communications disconnection error occurs can be selected with function code y02, y12. (For further information, see "2.4 Making RS-485-related settings.")

This section shows specific examples of action by different settings of function code y02. (The same operation is performed for y12 as well. In this case, the y02 and y03 in the figure are replaced with y12 and y13, and the error indication becomes E_rP .)

When y02 = 0 (mode in which the inverter is forced to immediately stop in case of communications error)



When y02 = 1 and y03 = 5.0 (seconds) (mode in which the inverter is forced to stop five seconds after a communications error occurred)

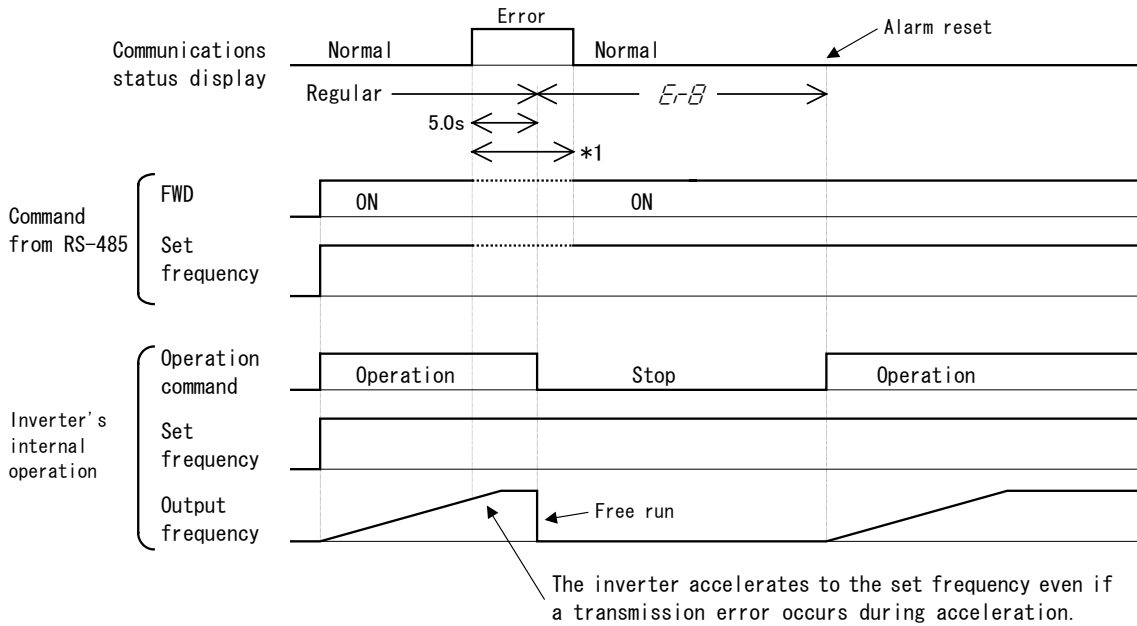


The inverter accelerates to the set frequency even if a transmission error occurs during acceleration.

*1 For the period until communications is recovered, the command (command data, operation data) executed just before the communications error had occurred is retained.

When y02 = 2 and y03 = 5.0 (seconds)

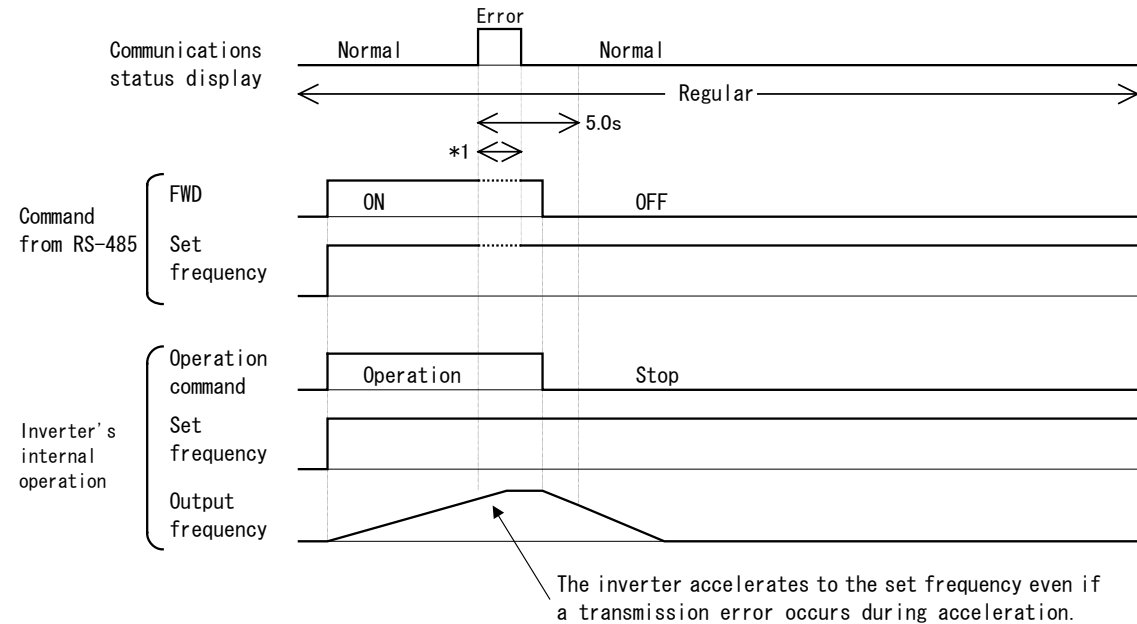
(when communications is not recovered although five seconds elapsed from the occurrence of a communications error, and an *E-rB* trip occurs)



*1 For the period until communications is recovered, the command (command data, operation data) executed just before the communications error had occurred is retained.

When y02 = 2 and y03 = 5.0 (seconds)

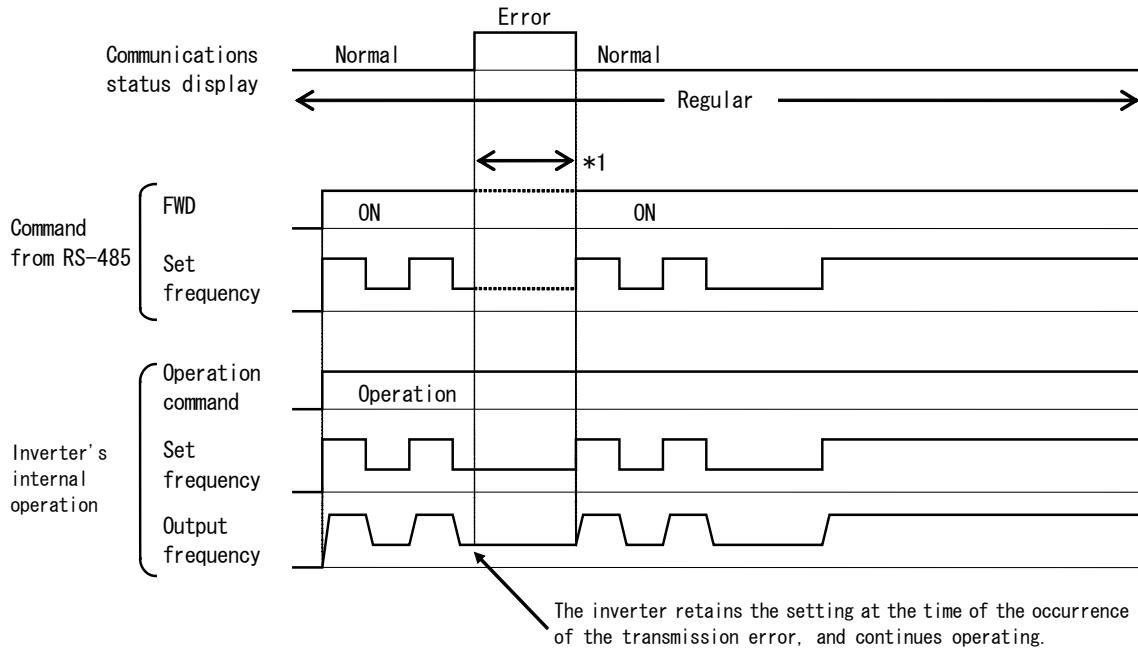
(when a communications error occurred but communications was recovered within five seconds)



*1 For the period until communications is recovered, the command (command data, operation data) executed just before the communications error had occurred is retained.

When y02 = 3

(mode in which the inverter continues operating when a communications error occurs)



*1 For the period until communications is recovered, the command (command data, operation data) executed just before the communications error had occurred is retained.

3.4 CRC-16

3.4.1 Overview of the CRC-16

The CRC (cyclic redundancy check) is a system to confirm whether there is any error in the communications frame during data transmission.

The CRC is among the most effective error check systems. The transmission station calculates and adds CRC data to the last block of the frame, and the receiving station also calculates CRC data against the data received, and compares them with each other.

Steps to calculate CRC data

- Divide data expressed as a polynomial (for example, 0000 0001 0000 0011 0000 0011 0000 0010 0000 0000 0001 0100, the 48-bit data shown in section 3.4.3 "Calculation example" → $X^{40}+X^{33}+X^{32}+X^{25}+X^{24}+X^{17}+X^4+X^2$) by a generative polynomial expression (17 bits; $X^{16}+X^{15}+X^2+1$). CRC data is the remainder (16 bits) of this division.
- Ignore the quotient, and send a message with the remainder added to the final two characters of the data.
- The receiving station divides this message (with the CRC added) by the generative polynomial expression, and considers the transmitted message to have been received without any error if the "remainder" is 0.

CRC-16

The generative polynomial expression is expressed as a multiplier of X, such as $X^3 + X^2 + 1$, in place of the description of binary code 1101. Although any prime polynomial expression is acceptable as the generative polynomial expression, some standard generative polynomial expressions for optimizing error detection are defined and proposed. The RTU protocol uses the generative polynomial expression ($X^{16} + X^{15} + X^2 + 1$) corresponding to binary code 1 (1000 0000 0000 0101). In this case, the CRC generated is well known as CRC-16.

3.4.2 Algorithm

Figure 3.1 on the following page shows the algorithm for calculating CRC-16. Consult it together with the calculation example that follows.

In this figure, the transmission station calculates CRC data and finally adds it to the transmission frame as a check code.

The receiving station uses the same algorithm to perform a transaction. However, it collates the CRC data it calculated with the transmitted CRC data.

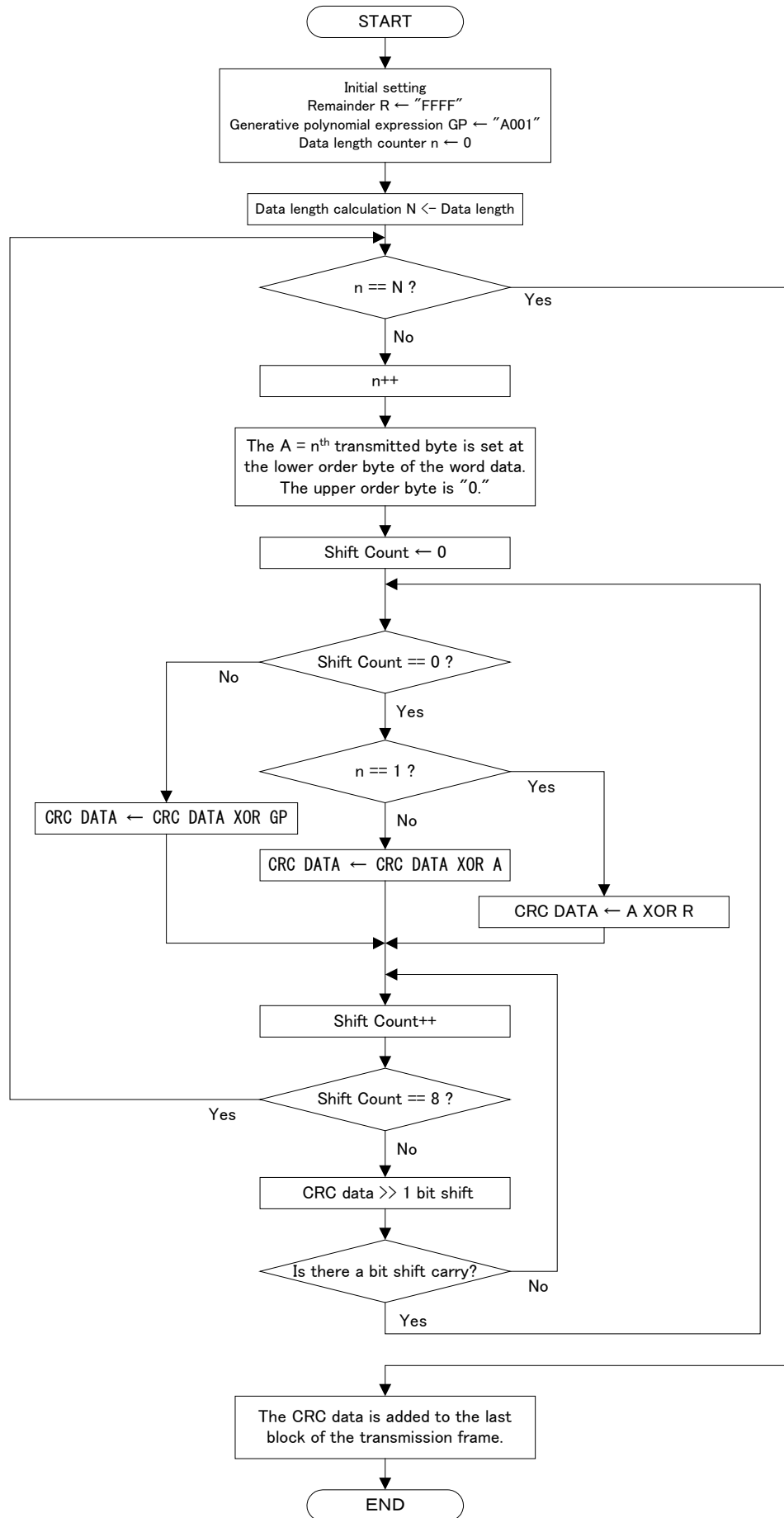


Figure 3.1 CRC algorithm

3.4.3 Calculation example

Example of transmitting read data

Station address = 1, 'FC' = 3, function code = P02 (P = 03_H, 02 = 02_H), number of read data = 20, GP = generative polynomial expression(1010 0000 0000 0001)

Station address	'FC'	Function code		Number of read data	
01 _H	03 _H	03 _H	02 _H	00 _H	14 _H

Table 3.13 CRC data calculation table

N	PROCESS	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Flag
1	Initial data R = "FFFF"	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	1 st data byte	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
3	CRC = No.1 Xor No.2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	
4	Shift >> 2 (up to flag = 1)	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	CRC = No.4 Xor GP	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	
6	Shift >> 2	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1
7	CRC = No.6 Xor GP	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	
8	Shift >> 2	0	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1
9	CRC = No.8 Xor GP	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	
10	Shift >> 2 (shift of No. 8 terminated)	0	0	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1
11	CRC = No.10 Xor GP	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	
12	2 nd data byte	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
13	CRC = No.11 Xor No.12	1	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	
14	Shift >> 1	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1
15	CRC = No.14 Xor GP	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	
16	Shift >> 1	0	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1
17	CRC = No.16 Xor GP	1	1	0	1	0	0	0	0	0	0	0	1	1	1	1	0	
18	Shift >> 2	0	0	1	1	0	1	0	0	0	0	0	0	0	1	1	1	1
19	CRC = No.18 Xor GP	1	0	0	1	0	1	0	0	0	0	0	0	0	0	1	1	0
20	Shift >> 2	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	1	1
21	CRC = No.20 Xor GP	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	
22	Shift >> 2 (shift of No. 8 terminated)	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0
23	3 rd data byte	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
24	CRC = No.22 Xor No.23	0	0	1	0	0	0	0	1	0	1	0	0	0	0	1	1	
25	Shift >> 1	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	1	1
26	CRC = No.25 Xor GP	1	0	1	1	0	0	0	0	1	0	1	0	0	0	0	0	
27	Shift >> 6	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	1
28	CRC = No.27 Xor GP	1	0	1	0	0	0	1	0	1	1	0	0	0	0	1	1	
29	Shift >> 1	0	1	0	1	0	0	0	1	0	1	1	0	0	0	0	1	1
30	CRC = No.29 Xor GP	1	1	1	1	0	0	0	1	0	1	1	0	0	0	0	0	
31	4 th data byte	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
32	CRC = No.30 Xor No.31	1	1	1	1	0	0	0	1	0	1	1	0	0	0	1	0	
33	Shift >>2	0	0	1	1	1	1	0	0	0	1	0	1	1	0	0	0	1
34	CRC = No.33 Xor GP	1	0	0	1	1	1	0	0	0	1	0	1	1	0	0	1	
35	Shift >> 1	0	1	0	0	1	1	1	0	0	0	1	0	1	1	0	0	1
36	CRC = No.35 Xor GP	1	1	1	0	1	1	1	0	0	0	1	0	1	1	0	1	
37	Shift >> 1	0	1	1	1	0	1	1	1	0	0	0	1	0	1	1	0	1

(To be continued)

Table 3.13 CRC data calculation table (Continued)

N	PROCESS	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Flag
38	CRC = No.37 Xor GP	1	1	0	1	0	1	1	1	0	0	0	1	0	1	1	1	
39	Shift >> 1	0	1	1	0	1	0	1	1	1	0	0	0	1	0	1	1	1
40	CRC = No.39 Xor GP	1	1	0	0	1	0	1	1	1	0	0	0	1	0	1	0	
41	Shift >>2	0	0	1	1	0	0	1	0	1	1	1	0	0	0	1	0	1
42	CRC = No.41 Xor GP	1	0	0	1	0	0	1	0	1	1	1	0	0	0	1	1	
43	Shift >> 1	0	1	0	0	1	0	0	1	0	1	1	1	0	0	0	1	1
44	CRC = No.43 Xor GP	1	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0	
45	5 th data byte	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
46	CRC = No.44 Xor No.45	1	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0	
47	Shift >> 5	0	0	0	0	0	1	1	1	0	1	0	0	1	0	1	1	1
48	CRC = No.47 Xor GP	1	0	1	0	0	1	1	1	0	1	0	0	1	0	1	0	
49	Shift >> 2	0	0	1	0	1	0	0	1	1	1	0	1	0	0	1	0	1
50	CRC = No.49 Xor GP	1	0	0	0	1	0	0	1	1	1	0	1	0	0	1	1	
51	Shift >> 1	0	1	0	0	0	1	0	0	1	1	1	0	1	0	0	1	1
52	CRC = No.51 Xor GP	1	1	1	0	0	1	0	0	1	1	1	0	1	0	0	0	
53	6 th data byte	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	
54	CRC = No.52 Xor No.53	1	1	1	0	0	1	0	0	1	1	1	1	1	1	0	0	
55	Shift >> 3	0	0	0	1	1	1	0	0	1	0	0	1	1	1	1	1	1
56	CRC = No.55 Xor GP	1	0	1	1	1	1	0	0	1	0	0	1	1	1	1	0	
57	Shift >> 2	0	0	1	0	1	1	1	1	0	0	1	0	0	1	1	1	1
58	CRC = No.57 Xor GP	1	0	0	0	1	1	1	1	0	0	1	0	0	1	1	0	
59	Shift >> 2	0	0	1	0	0	0	1	1	1	1	0	0	1	0	0	1	1
60	CRC = No.59 Xor GP	1	0	0	0	0	0	1	1	1	1	0	0	1	0	0	0	
61	Shift >> 1 (shift of No. 8 terminated)	0	1	0	0	0	0	0	1	1	1	1	0	0	1	0	0	0
Transmitted CRC data		4				1				E				4				

From the above calculation, the transmitted data is as shown below:

Station address	'FC'	Function code		Number of read data		CRC check	
01 _H	03 _H	03 _H	02 _H	00 _H	14 _H	E4 _H	41 _H

3.4.4 Frame length calculation

To calculate CRC-16, it is necessary to know the length of variable length messages. The length of all types of messages can be determined according to Table 3.14 Lengths of response messages.

Table 3.14 Length of response messages

'FC'	Description	Query/Broadcast message length (except CRC code)	Length of response message (except CRC code)
1	Read coil status	6 bytes	3+(3 rd) bytes*
3	Read holding registers	6 bytes	3 + (3 rd) bytes*
5	Force single coil	6 bytes	6 bytes
6	Preset single register	6 bytes	6 bytes
8	Diagnostics	6 bytes	6 bytes
15	Force multiple coils	7 + (7 th) bytes*	6 bytes
16	Preset multiple registers	7 + (7 th) bytes*	6 bytes
128 to 255	Exception function	Unused	3 bytes

* 7th, 3rd: The 7th and 3rd byte count values stored in the frame.