NXA-P/H Intelligent Controller Communication Protocol

——Modbus-RTU

User Manual



ZHEJIANG CHINT ELECTRICS CO.,LTD.

08-2022

CHNT

Catalog

Foreword 1
1 Scope 2
2 Definitions 2
2.1 open system interconnection（OSI）model 2
2.2 physical layer 2
2.3 data link layer (DDL) 2
2.4 application layer 2
2.5 Frame 2
3 Modbus protocol introduction 2
4 protocol introduction 2
4.1 physical layer 2
4.2 data link layer 3
4.2.1 Transmission mode 3
4.2.2 Protocol type. 3
4.2.3 Serial transmission format 3
4.2.4 Data packet (multiple frames) format 3
4.3 application layer 3
4.3.1 Address code 3
4.3.2 Function code 3
4.3.3 Data Domain 4
4.3.4 Check code 4
4.3.5 Application Layer Explanation 4
4.3.5.1 Read Register（03H） 4
4.3.5.2 Preset single register（06H） 5
5 Comm. Address List 5
5.1 Measuring message 5
5.2 System Maintenance 7
5.3 fault record 7
5.4 parameter setting 10
5.5 remote control 13

Appendix F1 CRC-16 generation principle 14
Appendix F2 communication application example 14

Foreword

This document is put forward by the secondary distributing apparatus manufacturing department, ZHEJIANG CHINT ELECTRICS CO.,LTD.

Our company reserves the right to update this document without prior notice. Please check for latest edition.

CHINT

P/H Intelligent Controller Communication Protocol

——Modbus-RTU

1 Scope

This document specifies the basic terminology, protocol content and communication data tables of Modbus-RTU.

This document applies to NXA series ACB with P/H controller.

2 Definitions

The terminology below is applicable to this document.

2.1 open system interconnection（OSI）model

International standardization organization (ISO) set the standard in 1984 to provide a common base and a standard framework for computers from different manufacturers.

2.2 physical layer

Layer 1 of OSI model; provides physical link for communication.

2.3 data link layer (DDL)

Layer 2 of OSI model; provides transparent and reliable information transmission service between adjacent nodes.

2.4 application layer

Layer 7 of OSI model; realizes the function of data manipulation and information exchange.

2.5 Frame

In data and digital communication, a frame is a specific information structure composed of several bits or fields according to a standard. Data on the internet is transmitted frame by frame. Different parts of frame have different functions.

3 Modbus protocol introduction

Modbus is an industrial bus protocol based on ISO/OSI model (7 layers). However, only 3 layers (physical layer, data link layer and application layer) are used, which helps to simplify protocol model and reduce the difficult to use it.

Modbus has two transmission modes, ASCII and RTU. In this document, we adopt RTU mode.

4 protocol introduction

4.1 physical layer

physical layer parameter	**content**	**remark**
Communication mode	RS-485	Half duplex
Communication address	1 ~ 247 optional	default: 3
Communication baud rate	9.6kbps\19.2kbps optional	default: 9.6kbps

Communication distance	≤ 1000 m	Under situation of low baud rate
Communication medium	Shielded twisted pair	A class
Max connections	32	Computer networking

4.2 data link layer

4.2.1 Transmission mode: half duplex protocol (host computer queries and slave computers reply)

4.2.2 Protocol type: Modbus-RTU.

4.2.3 Serial transmission format: 1 start bit, 8 data bits, no check bit and 2 stop bits. (A frame of data)

start	**data**								**stop**	
Start	1	2	3	4	5	6	7	8	Stop	Stop

4.2.4 Data packet (multiple frames) format:

start	**address**	**function**	**data**	**check**	**stop**
T3.5	8 bits	8 bits	n×8 bits	16 bits	T3.5

Note: in RTU mode, information starts with 3.5-characters (frames) quiescent time at least. The quiescent time, such as T3.5 shown in table above, can be calculated easily according to the used baud rate. The delay should be taken into consideration, if using UART to establish a communication protocol. It can be ignored if configuration software or DCS is adopted, because underlying part is accomplished.

4.3 application layer

Application layer can parse the content of packet to exchange data.

When the slave computer receives the packet from the host computer, the packet will enter addressing equipment by communication port. The slave computer will strip the data headers to read valid data. If there is no error, the salve computer will execute the task requested by data, add the new-generated data into “envelop” to form a new data packet and send the packet to the host computer. The data returned includes the salve computer address (Address), executed function (Function), requested data generated by executing command (Data) and a check code (Check).

4.3.1 Address code

Address code, in the beginning of frame, is an 8-bit code (1-247). The code addresses the user specified slave computer and the salve computer can receive data from the connected host computer. The address of every salve computer must be exclusive. The exclusively addressed terminal will respond to the inquiry including this address. When a response is sent back, the subordinate address data included in the response will assure the host computer which device is communicating with it.

4.3.2 Function code

The function code tells the located terminal of the function to be executed. All function codes, their definitions and initial functions are listed below.

Code	**Definition**	**Task**

03H	Reading data register	Obtain the binary value of one or more registers
06H	Preset single register	Put a certain given binary value into a register

4.3.3 Data Domain

The data domain contains the data needed by the terminal to execute a certain function or the data gathered by the terminal in responding an inquiry. These data could be numerical value, reference address or extreme value. For example, the function domain code tells the terminal to read a register, while the data domain indicates which register to start and how many data to read. The inline address and data varies according to types and the subordinate devices.

4.3.4 Check code

This domain makes possible the principle and terminal to check errors during transmission. Because of electrical noise or other jams, some changes may happen to data on road from one device to another. The check domain makes sure neither the principle nor the subordinate respond to changed data, which improves the safety and efficiency of the system. The check applies 16-bit-circulation-redundance.

Circulation-Redundancy-Check （CRC）domain occupies two bits, including a 16-digit binary value. The transmission device calculates out the CRC value, adds it to the data frame; the receiving device also calculates the CRC value, compares it with the received CRC domain value, and if these two values don't match, it proves a mistake has occurred.

Note: generation method of CRC16 please refers to Appendix F1 (CRC-16 generation principle)

4.3.5 Application Layer Explanation

4.3.5.1 Read Register（03H）

Any data and system parameters collected and reported by controller is available to user by 03 function code.

For example, read Ia，Ib，Ic and get the result of Ia=0001，Ib=0002，Ic=0003, according to returned data.

Host computer queries			**Salve computes responds**		
Frame domain	**content**	**explanation**	**Frame domain**	**content**	**explanation**
Address code	03	Salve computer address	Address code	03	Salve computer address
Function Code	03	Function Code	Function Code	03	Function Code
Data Domain	00	Read high byte of register address	Data Domain	06	Return the total number of data
	01	Read low byte of register address		00	high byte of data1
				01	low byte of data1
	00	Read high byte of number of data		00	high byte of data2
				02	low byte of data2
	03	Read low byte of number of data		00	high byte of data3
				03	low byte of data3
Check	55	CRC checks low byte	Check	E4	CRC checks low byte

code	E9	CRC checks high byte	code	14	CRC checks high byte

Host computer sends [03 03 00 01 00 03 55 e9]

Salve computer responds [03 03 06 00 01 00 02 00 03 e4 14]

4.3.5.2 Preset single register（06H）

Function code 06 makes possible the user change the content of the single register. Every single register in the DAE system can use this command to change values.

For example, change long delay setting current (register address: v0x2007) to 2000A (hex: 0x07d0).

Host computer queries			**Salve computes responds**		
Frame domain	**content**	**explanation**	**Frame domain**	**content**	**explanation**
Address code	03	Salve computer address	Address code	03	Salve computer address
Function Code	06	Function Code	Function Code	06	Function Code
Data Domain	20	Address high byte to write	Data Domain	20	Write high byte of address
	07	Address low byte to write		07	Write low byte of address
	07	Write data high byte		07	Write data high byte
	D0	Write data low byte		D0	Write data low byte
Check code	31	CRC checks low byte	Check code	31	CRC checks low byte
	85	CRC checks high byte		85	CRC checks high byte

Host computer sends [03 06 20 07 07 d0 31 85]

Salve computer responds [03 06 20 07 07 d0 31 85]

5 Comm. Address List

WORD is word-size. UINT is unsigned integer. SINT is signed integer. ULONG is long integer. SLONG is a signed long integer, ULONG LONG is double long integer. BCD is BCD code. R is read only. W is write only. R/W is read-write.

5.1 Measuring message

No.	Content	Data type	Unit	Attribute	Address	Remark
1	Operating condition word	WORD	-	R	0x0000	See Appendix A
2	L1 current	UINT	A	R	0x0001	NOTE 1
3	L2 current	UINT	A	R	0x0002	NOTE 1
4	L3 current	UINT	A	R	0x0003	NOTE 1
5	LN current	UINT	A	R	0x0004	NOTE 1
6	LG current	UINT	A	R	0x0005	NOTE 1
7	L1 voltage	UINT	V	R	0x0006	×0.1
8	L2 voltage	UINT	V	R	0x0007	×0.1

9	L3 voltage	UINT	V	R	0x0008	×0.1
10	L1-2 voltage	UINT	V	R	0x0009	×0.1
11	L2-3 voltage	UINT	V	R	0x000A	×0.1
12	L3-1 voltage	UINT	V	R	0x000B	×0.1
13	Power factor	SINT	%	R	0x000C	×0.01
14	frequency	UINT	Hz	R	0x000D	×0.01
15	Line voltage average	UINT	V	R	0x0010	×0.1
16	Unbalanced rate of current	UINT	%	R	0x0011	×0.1
17	Phase L1 active power	SINT	kW	R	0x0021	NOTE 1
18	Phase L1 reactive power	SINT	kVar	R	0x0022	NOTE 1
19	Phase L1 apparent power	UINT	KVA	R	0x0023	NOTE 1
20	Phase L2 active power	SINT	kW	R	0x0024	NOTE 1
21	Phase L2 reactive power	SINT	kVar	R	0x0025	NOTE 1
22	Phase L2 apparent power	UINT	kVA	R	0x0026	NOTE 1
23	Phase L3 active power	SINT	kW	R	0x0027	NOTE 1
24	Phase L3 reactive power	SINT	kVar	R	0x0028	NOTE 1
25	Phase L3 apparent power	UINT	kVA	R	0x0029	NOTE 1
26	The total active power	SINT	kW	R	0x002A	NOTE 1
27	The total reactive power	SINT	kVar	R	0x002B	NOTE 1
28	The total apparent power	UINT	KVA	R	0x002C	NOTE 1
29	Total active power	UINT32	kWh	R	0x002D	注 1
				R	0x002E	
30	Total reactive power	UINT32	kvarh	R	0x0031	注 1
				R	0x0032	
31	Total apparent electrical energy	UINT32	kVAh	R	0x0035	注 1
				R	0x0036	
32	the number of operations	UINT16	times	R	0x0047	
33	L1 Required current	UINT16	A	R	0x0050	注 1
34	L2 Required current	UINT16	A	R	0x0051	注 1
35	L3 Required current	UINT16	A	R	0x0052	注 1
36	LN Required current	UINT16	A	R	0x0053	注 1
37	Total active power required	UINT16	kW	R	0x0054	注 1
38	Total reactive power required	UINT16	kVar	R	0x0055	注 1
39	The total apparent power required	UINT16	kVA	R	0x0056	注 1

NOTE 1: when frame current <3200, value×1. With other frame current, value×2.

Appendix A Bit format of operating conditions word

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
-	-	-	-	-	-	-	-
Bit7	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
-	-	Open/close indicating status bit 0: close 1: open	Trip indicating status bit 0: normal 1: fault	-	-	-	-

5.2 System Maintenance

No.	Content	Data type	Unit	Attribute	Address	Remark
1	ModBus address	UINT	-	R/W	0x0100	
2	ModBus baud rate	UINT	-	R/W	0x0101	
3	System time(year\month)	BCD\BCD	-	W	0x0140	High 8 bit\low 8 bit
4	System time (date\hour)	BCD\BCD	-	W	0x0141	
5	System time (min\sec)	BCD\BCD	-	W	0x0142	
6	Frame current	UINT	A	R	0x0180	See Appendix B
7	Rated current	UINT	A	R	0x0181	NOTE 3
8	Rated voltage	UINT	V	R	0x0182	

Note 3: when frame current <3200, value×1. With other frame current, value×2.

Appendix B

Frame code	Frame current	**Remark**
0	1600	
1	2000	
2	2500	
3	3200	
4	4000	
5	6300	

5.3 fault record

No.	Content	Data type	Unit	Attribute	Address	Remark
1	1st trip time(year\month)	BCD\BCD	-	R	0x0200	
2	1st trip time (date\hour)	BCD\BCD	-	R	0x0201	
3	1st trip time (min\sec)	BCD\BCD	-	R	0x0202	
4	Reason for 1st trip	WORD	-	R	0x0203	See Appendix C
5						
6	L1 current of 1st trip	UINT	A	R	0x0208	Note 4
7	L2 current of 1st trip	UINT	A	R	0x0209	Note 4

8	L3 current of 1st trip	UINT	A	R	0x0210	Note 4
9	L4 current of 1st trip	UINT	A	R	0x0211	Note 4
10	1st trip time	UINT	s	R	0x0213	Note 5
11	2nd trip time(year\month)	BCD\BCD		R	0x0214	
12	2nd trip time (date\hour)	BCD\BCD		R	0x0215	
13	2nd trip time (min\sec)	BCD\BCD		R	0x0216	
14	Reason for 2nd trip	WORD		R	0x0217	See Appendix C
15	L1 current of 2nd trip	UINT	A	R	0x021C	Note 4
16	L2 current of 2nd trip	UINT	A	R	0x021D	Note 4
17	L3 current of 2nd trip	UINT	A	R	0x021E	Note 4
18	L4 current of 2nd trip	UINT	A	R	0x021F	Note 4
19	2nd trip time	UINT	s	R	0x0227	Note 5
20	3rd trip time(year\month)	BCD\BCD		R	0x0228	
21	3rd trip time (date\hour)	BCD\BCD		R	0x0229	
22	3rd trip time (min\sec)	BCD\BCD		R	0x022A	
23	Reason for 3rd trip	WORD		R	0x022B	See Appendix C
24	3rd trip time	UINT	s	R	0x0230	Note 5
25	4th trip time(year\month)	BCD\BCD		R	0x0231	
26	4th trip time (date\hour)	BCD\BCD		R	0x0232	
27	4th trip time (min\sec)	BCD\BCD		R	0x0233	
28	Reason for 4th trip	WORD		R	0x0234	See Appendix C
29	4th trip time	UINT	s	R	0x0239	Note 5
30	5th trip time(year\month)	BCD\BCD		R	0x023A	
31	5th trip time (date\hour)	BCD\BCD		R	0x023B	
32	5th trip time (min\sec)	BCD\BCD		R	0x023C	
33	Reason for 5th trip	WORD		R	0x023D	See Appendix C
34	5th trip time	UINT	s	R	0x0242	Note 5
35	6th trip time(year\month)	BCD\BCD		R	0x0243	
36	6th trip time (date\hour)	BCD\BCD		R	0x0244	
37	6th trip time (min\sec)	BCD\BCD		R	0x0245	
38	Reason for 6th trip	WORD		R	0x0246	See Appendix C
39	6th trip time	UINT	s	R	0x024B	Note 5
40	7th trip time(year\month)	BCD\BCD		R	0x024C	
41	7th trip time (date\hour)	BCD\BCD		R	0x024D	
42	7th trip time (min\sec)	BCD\BCD		R	0x024E	
43	Reason for 7th trip	WORD		R	0x024F	See Appendix C

CHINT

44	7th trip time	UINT	s	R	0x0254	Note 5
45	8th trip time(year\month)	BCD\BCD		R	0x0255	
46	8th trip time (date\hour)	BCD\BCD		R	0x0256	
47	8th trip time (min\sec)	BCD\BCD		R	0x0257	
48	Reason for 8th trip	WORD		R	0x0258	See Appendix C
49	8th trip time	UINT	s	R	0x025D	Note 5
50	9th trip time(year\month)	BCD\BCD		R	0x0255	
51	9th trip time (date\hour)	BCD\BCD		R	0x0256	
52	9th trip time (min\sec)	BCD\BCD		R	0x0257	
53	Reason for 9th trip	WORD		R	0x0258	See Appendix C
54	9th trip time	UINT	s	R	0x025D	Note 5
55	10th trip time(year\month)	BCD\BCD		R	0x0255	
56	10th trip time (date\hour)	BCD\BCD		R	0x0256	
57	10th trip time (min\sec)	BCD\BCD		R	0x0257	
58	Reason for 10th trip	WORD		R	0x0258	See Appendix C
59	10th trip time	UINT	s	R	0x025D	Note 5

NOTE 4: when frame current <3200, value×1. With other frame current, value×2.

NOTE 5: Required current Fault, value×0.1. With other Fault, value×0.01.

Appendix C Fault Type Sheet

Code	Type	remark
00H(hex)	Fault free	0(dec)
01H	Overload long delay fault	1
02H	Short circuit short delay fault	2
03H	Short circuit transient fault	3
04H	Earth fault	4
05H	Leakage trip fault	5
06H	Neutral fault	6
07H	Unbalanced current fault	7
09H	Short circuit interlocking fault	9
0AH	Earth interlocking fault	10
0BH	Undervoltage fault	11
0CH	Overvoltage fault	12
0DH	Unbalanced voltage fault	13
0EH	Under frequency fault	14
0FH	Over frequency fault	15
10H	Phase sequence fault	16
11H	Reverse power fault	17
12H	Required current Fault	18

5.4 parameter setting

No.	Content	Data type	Unit	Attribute	Address	Remark
1	Protection function set 0 (current basic protection)		-	R/W	0x2000	See Appendix D
2	Protection function set 1		-	R/W	0x2001	See Appendix E
3	Protection function set 2		-	R/W	0x2002	See Appendix F
4	Long delay setting current	UINT16	A	R/W	0x2007	Note6-8
5	Long delay setting time	UINT16	s	R/W	0x2008	Note 7
6	Short delay setting current	UINT16	A	R/W	0x2009	Note6-8
7	Short delay setting time	UINT16	ms	R/W	0x200A	Note 7
8	instantaneous current	UINT16	A	R/W	0x200B	Note6-8
9	Earth protection setting current	UINT16	A	R/W	0x200C	Note7-8
10	Earth protection setting time	UINT16	ms	R/W	0x200D	Note 7
11	I unbalance starting value	UINT16	%	R/W	0x200E	
12	I unbalance starting time	UINT16	s	R/W	0x200F	×0.02
13	I unbalance returned value	UINT16	%	R/W	0x2010	
14	I unbalance returned time	UINT16	s	R/W	0x2011	×0.02
15	U unbalance starting value	UINT16	%	R/W	0x2016	
16	U unbalance starting time	UINT16	s	R/W	0x2017	×0.02
17	U unbalance returned value	UINT16	%	R/W	0x2018	
18	U unbalance returned time	UINT16	s	R/W	0x2019	×0.02
19	Undervoltage starting value	UINT16	V	R/W	0x201A	
20	Undervoltage starting time	UINT16	s	R/W	0x201B	×0.02
21	Undervoltage returned value	UINT16	V	R/W	0x201C	
22	Undervoltage returned time	UINT16	s	R/W	0x201D	×0.02

CHINT

23	Overvoltage starting value	UINT16	V	R/W	0x201E	
24	Overvoltage starting time	UINT16	s	R/W	0x201F	×0.02
25	Overvoltage returned value	UINT16	V	R/W	0x2020	
26	Overvoltage returned time	UINT16	s	R/W	0x2021	×0.02
27	Under frequency starting value	UINT16	Hz	R/W	0x2022	×0.1
28	Under frequency starting time	UINT16	s	R/W	0x2023	×0.02
29	Under frequency returned value	UINT16	Hz	R/W	0x2024	×0.1
30	Under frequency returned time	UINT16	s	R/W	0x2025	×0.02
31	Over frequency starting value	UINT16	Hz	R/W	0x2026	×0.1
32	Over frequency starting time	UINT16	s	R/W	0x2027	×0.02
33	Over frequency returned value	UINT16	Hz	R/W	0x2028	×0.1
34	Over frequency returned time	UINT16	s	R/W	0x2029	×0.02
35	Reverse power setting	UINT16	kW	R/W	0x202E	Note 6
36	Reverse power time setting	UINT16	s	R/W	0x202F	×0.02
37	Reverse power Return setting	UINT16	kW	R/W	0x2030	Note 6
38	Reverse power Return time setting	UINT16	s	R/W	0x2031	×0.02
39	A Required current setting	UINT16	A	R/W	0x2032	Note 6
40	A Required current time setting	UINT16	s	R/W	0x2033	
41	A Required current Return setting	UINT16	A	R/W	0x2034	Note 6
42	A Required current Return time setting	UINT16	s	R/W	0x2035	
43	B Required current setting	UINT16	A	R/W	0x2036	Note 6
44	B Required current time setting	UINT16	s	R/W	0x2037	

CHINT

45	B Required current Return setting	UINT16	A	R/W	0x2038	Note 6
46	B Required current Return time setting	UINT16	s	R/W	0x2039	
47	C Required current setting	UINT16	A	R/W	0x203A	Note 6
48	C Required current time setting	UINT16	s	R/W	0x203B	
49	C Required current Return setting	UINT16	A	R/W	0x203C	Note 6
50	C Required current Return time setting	UINT16	s	R/W	0x203D	
51	N Required current setting	UINT16	A	R/W	0x203E	Note 6
52	N Required current time setting	UINT16	s	R/W	0x203F	
53	N Required current Return setting	UINT16	A	R/W	0x2040	Note 6
54	N Required current Return time setting	UINT16	s	R/W	0x2041	
55	Current load monitoring1 setting	UINT16	A	R/W	0x2042	Note 6
56	Current load monitoring1 time setting	UINT16	%Tr	R/W	0x2043	
57	Current load monitoring2 setting	UINT16	A	R/W	0x2044	Note 6
58	Current load monitoring2 time setting	UINT16	s	R/W	0x2045	×0.02
59	Power load monitoring1 setting	UINT16	kW	R/W	0x2046	
60	Power load monitoring1 time setting	UINT16	s	R/W	0x2047	×0.1
61	Power load monitoring2 setting	UINT16	kW	R/W	0x2048	
62	Power load monitoring2 time setting	UINT16	s	R/W	0x2049	×0.1

Note 6: when frame current <3200, value×1. With other frame current , value×2.

Note 7: setting range of each parameter please refer to operation instruction

Note 8: 65535（0XFFFF）means the function is turned off.

Appendix D Bit format of protection function set

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
Long delay cooling time 000: instant 001: 10min 010: 20min 011: 30min			-	-	-	Long delay protection curve type 00: I^2T 01: IT 10: I^4T	
Bit7	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
-	-	-	-	-	-	-	-

Appendix E Protection function set 1

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
		Reverse power protection 00: turned off 01: trip 10: alarm		I unbalance protection 00: turned off 01: trip 10: alarm		load monitoring 00: turned off 01: Current 10: Power	
Bit7	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
N Required current protection 00: turned off 01: trip 10: alarm		C Required current protection 00: turned off 01: trip 10: alarm		B Required current protection 00: turned off 01: trip 10: alarm		A Required current protection 00: turned off 01: trip 10: alarm	

Appendix F Protection function set 2

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
Undervoltage protection 00: turned off 01: trip 10: alarm		Overvoltage protection 00: turned off 01: trip 10: alarm					
Bit7	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
U unbalance protection 00: turned off 01: trip 10: alarm				Under frequency protection 00: turned off 01: trip 10: alarm		Over frequency protection 00: turned off 01: trip 10: alarm	

5.5 remote control

No.	Content	Data type	Unit	Attribute	Address	Remark
1	Control command	UINT16	-	W	0x2800	See Appendix G

Appendix G Bit format of control command

Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
-	-	-	Relay trip control bit 0: no trip 1: trip	-	-	Open/close control 01: open 10: close	
Bit7	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
-	-	-	-	-	-	-	-

Appendix F1 CRC-16 generation principle

CRC set every digit of a 16-bit register to 1, calculates the 8 digits of the data frame and the current value of the register; only 8 bits of every byte are involved in the make of the CRC. The initial bit, end bit and parity bit are irrelevant to CRC.

In the make of CRC, every 8-bit byte carries out “ XOR ” operation with the content of the register, shift the result to the lower bit, supplement the higher bit with “0”, measure the lowest bit（LSB）, and if 1, this register carries an “ XOR ” operation with a pre-set fixed value, and if the LSB is 0, no treatment is to be given.

Repeat the above circle 8 operations. Next 8-bit byte carries out an “XOR” operation with the current value of the register and do the above circle after the last bit in the data frame are shifted like this. The finally generated value is the CRC after all byte in data frame was done with the above circle operations.

The making process of a CRC:

a) Preset a 16-bit register as 0FFFFH（all 1）, (CRC register).

b) The first 8-bit byte carries out an “XOR” operation with the low byte of the CRC register; store the result in the CRC register

c) Shift 1 bit of the CRC register to the right, replace the HSB with “0”, shift out and measure the LSB.

d) If the LSB is 0: repeat the step 3(next shift).

e) If the LSB is 1: carry out the “ XOR ” operation of the CRC register and a preset fixed value（0A001H）.

f) Repeat step c and d until 8digit shift.

g) Repeat step b to step e to handle the next 8-bit until all bytes are processed.

h) The final CRC register value is the CRC.