

## **CRONY-H-485 Reader Command Listing**

Following is the format of Protocol.

### **HEAD**

**SOH TYPE ID FC DATA BCC CHECK 8 BITS BCC END** 0x09 A BCC1 BCC2 0x0D

**1. SOH** and **END** are control character of a byte.

**SOH** Controller Transceiver Message <0x0A> (PC Transceiver Message): <0x09> **END** Controller and PC Transceiver Message <0x0D> <0x> is Hexadecimal.

**2. TYPE** is assigned number of Module Type and fixed a byte.

This type assigned fixed number calling “A”.

**3. ID** is the Identity Code of Controller Transceiver Message, and is a byte of **ASCII**. **ASCII** shall be within the range between **1<0x31>** and **9 <0x38>**.

If the ID of PC transceiver message is same with the controller ID code, the controller will receive the transceiver message of PC. After the controller received, the controller will reply same ID code to PC.(default is **1<0x31>**)

**4. FC** is Function Code related to DATA. It’s a byte. Please refer the description of Controller & PC Protocol Table.

**5. 8 BITS BCC** is to check all byte. It’s binary. The sample and description of **8 BITS BCC** please refer to the default transmission protocol of RS485 is”19200,E,8,1”.

### **UART CONFIGURATION”19200,E,8,1”**

#### **1. Get Module Factory Code ‘B’**

	<b>SHO</b>	<b>TYPE</b>	<b>I D</b>	<b>FC</b>	<b>DATA</b>	<b>BCC CHECK</b>		<b>END</b>
<b>Command</b>	0x09	A	1	B		BCC1	BCC2	0x0D
<b>Response</b>	0x0A	A	1	B	SERIAL NO. 8 DIGIT	BCC1	BCC2	0x0D

**2. Set Module ID Code ‘C’**

	SH O	TYPE	ID	F C	DATA	BCC CHECK		END	
<b>Command</b>	0x09	A	X	C	Serial no	New ID	BCC1	BCC2	0x0D
<b>Response</b>	0x0 A	A	NEW ID	C			BCC1	BCC2	0x0D

Module ID can be 0x1 to 0xF

**3. Get Module ID Code ‘D’**

	SHO	TYP E	ID	F C	DATA	BCC CHECK		END
<b>Command</b>	0x09	A	X	D	Serial no. 8 digit	BCC1	BCC2	0x0D
<b>Response</b>	0x0A	A	ID	D	ID	BCC1	BCC2	0x0D

**4. Get Module Version number ‘V’**

	SHO	TYPE	ID	FC	DATA	BCC CHECK		END
<b>Command</b>	0x09	A	X	V		BCC1	BCC2	0x0D
<b>Response</b>	0x0A	A	ID	V	Version Number	BCC1	BCC2	0x0D

**5. Read Card SNR and clear memory ‘F’**

	SHO	TYP E	ID	FC	DATA	BCC CHECK		END
<b>Command</b>	0x09	A	ID	F		BCC1	BCC2	0x0D
<b>Response</b>	0x0A	A	ID	F	CARD NO. IN HEX	BCC1	BCC2	0x0D

CARD NO. IN HEX is 9 char first char is 0(0x30) fix remaining is SNR 4 Byte in hex.

**6. Read SC(Mifare) Card Sector ‘S’**

	SHO	TYPE	ID	FC	DATA	BCC CHECK		END
<b>Command</b>	0x09	A	X	S	Sector No. in HEX	BCC1	BCC2	0x0D
<b>Response</b>	0x0A	A	ID	S	Sector Data	BCC1	BCC2	0x0D

Sector Number in Hex e.g. F for sector 15 Sector data: 16Byte in Hex total 32 Character  
OR

1Byte error code 2 character.

**Note:** Work with default key only i.e 0xFFFFFFFFFFFF

**7. Beep Module Buzzer ‘T’**

	SHO	TYPE	ID	FC	DATA	BCC CHECK		END	
<b>Command</b>	0x09	A	X	T	Beep Duration	Count	BCC1	BCC2	0x0 D
<b>Response</b>	0x0A	A	ID	T			BCC1	BCC2	0x0 D

**Beep duration:** Hex Value (01,0xFF) in multiple of 10ms Max 2.55 Second

**Beep Count:** Number of time beep 0 to 9

**8. Lock Open ‘L’**

	SHO	TYPE	ID	FC	DATA	BCC CHECK		END
<b>Command</b>	0x09	A	X	L	Open Duration 2 char.	BCC1	BCC2	0x0D
<b>Response</b>	0x0A	A	ID	L		BCC1	BCC2	0x0D

Lock open duration in seconds 00-99

Appendix A make one bcc check code All transmitting data uses XOR(exclusive-or)  
Calculating to produce BCC check code.

**A B A XOR B**

1 1 0

1 0 1

0 1 1

0 0 0

**The format looks like table below:**

**HEAD**

**SOH TYPE ID FC DATA BCC CHECK 8 BITS BCC END**

09(HEX) "A" "1" "F" NONE BCC1 BCC2 0D(HEX)

**There are 4 steps:**

1. No-including BCC and END code
2. First XOR "SOH" and "TYPE"
3. "ID" XOR with XOR of step 2
4. "FC" XOR with step 3, until to the "protocol data" end. The XOR result is BCC code.

**Example:**

1. Assign the values:

**MESSAGE**

**SEGMENT**

**HEX**

**VALUE COMMENTS**

SOH 09(HEX) start code

TYPE 41(HEX) Type "A"(41)

ID 31(HEX) ID=1

FC 46(HEX) Function code= "F"

DATA -- None

BCC ??

END 0D(HEX) End code

2. "SOH" XOR "TYPE"

SOH 0000 1001

XOR XOR

TYPE 0100 0001

---- ----

ANS 0100 1000

3. ID XOR ANS

ANS 0100 1000

XOR XOR

ID 0011 0001

---- ----

ANS 0111 1001

ANS 0111 1001

XOR XOR

FC 0100 0110

---- ----

ANS 0011 1111

3 F

BCC code:

Hi Byte is 33(HEX) ("3" ASCII code is 33(HEX)).

Low Byte is 46(HEX) ("F" ASCII code is 46(HEX)).

**Program to generate BCC values:**

```
10 *****
20 * Generate Block Check Character *
30 *****
40 MESSAGE$ = CHR$(10) +TYPE$ + ID$ + FC$ + DATA$
50 BCC = 0
60 FOR I = 1 TO LEN (MESSAGE$)
70 BCC = BCC XOR ASC (MID$ (MESSAGE$,I,1 ) )
80 NEXT I
```

**\*\*\*\*The End\*\*\*\***