UHF860 Series

Programming Guide
Version 1.09

November 6, 2013

History

REVISIONS

Rev Number	Date	Description
01	18 Dec 12	Initial Release
02	21 Dec 12	1. Modify Page12 EEPROM Parameter Table some phraseology 2. Page 23,24 : The “sending time “ to correct “allocation time” 3. Modify Page 23 example 4. Add history page
03	22 Dec 12	1. Page 7 : Add caution note of Active Mode 2. Page12 **:** The “Active Channel Number” to correct “ Number of Active Channel” 3. Page 21 : Modify relative phraseology 4. Page 22 : Modify ” Add the frequency to the List” and “Clear the List And add a frequency”
04	28 Mar 13	1. Page19 : Add 2 Bytes of SA for Write Tag Data [0x35] 2. Page20 : Add 2 Bytes of SA for Read Tag Data [0x37]
05	9 Apr 13	1. Page 21 : Modify “Get Frequency RSSI**”** 2. Page 21 : Modify “Get Frequency Reflected Power” G Value
06	22 Jul 13	1. Page 12,14 : Add DI Trigger Control without Retrigger and Level off Modify Wiegand [17][18],Trigger Mode -> Scan Tag Mode Add Trigger Mode[19] 0xFE : Always Trigger with Anti-Passback Delay 2. Page 12,15 : Add EPC Filter Function [1B][1C]
07	1 Aug 13	1. Page 16 : Modify item : **Frequency of Channel n**
08	4 Sep 13	1. Page 4 : add new page 4 : **Device Response DI Trigger Event Package** 2. Modify “Standalone Mode” change to “Auto Mode”,” GUI” change to “ Command” 3. Page18 : add new page18 : **Digital Input Trigger Mode Timing**
09	6 Nov 13	1. Page 3 : Modify error : The Example of checksum

Communications protocol

Default : Communications Baud Rate is 115200-8-N-1

Master Query Package (HOST)

Header	Address	Function	Data Length	Data Bytes	Error Check
1 Byte	1 Byte	1 Byte	1 Byte	0~255 Byte	2 Bytes

Note:
Header = SOH (01h)
Address = Device Address (Slave Machine ID)
Function = Query Function Command
Data Length = Total Number form Data Bytes
Data Bytes = Data
Error Check = CRC16 is for fields from "Address" to "Data Bytes"

PS: When the 16-bit CRC (twO eight-bit bytes) is transmitted in the message, the low order byte will be transmitted first, followed by the high order byte.

Slave Response Package (DEVICE)

Header	Address	Function	Data Length	Data Bytes	Error Check
1 Byte	1 Byte	1 Byte	1 Byte	0~255 Byte	2 Bytes

Note:
Header = SOH (01h)
Address = Device Address (Slave Machine ID)
Function = Response Function
ACK = 06h, Acknowledge (Passive, in response to Master message)
NAK = 15h, Negative Acknowledge (Passive, in response to Master message)
EVN = 11h, Event Message(Active, For One Host to One Device Connection)
Data Length = Total Number form Data Bytes
Data Bytes = Data
Error Check = CRC16 is for fields from "Address" to "Data Bytes"

PS: When the 16-bit CRC (twO eight-bit bytes) is transmitted in the message, the low order byte will be transmitted first, followed by the high order byte.

Response NAK Code Table (Common)

Function	Data Length	Data Bytes	Description
NAK	1	E0h	Access Denied
NAK	1	E4h	Illegal Query Code
NAK	1	E6h	Overrun, Out of record count
NAK	1	E7h	CRC Error
NAK	1	ECh	Query Number no support
NAK	1	EDh	Out Of Memory Range
NAK	1	EEh	Address Number out of range
NAK	1	EFh	Unknown

Response Event (For Active Slave) For Multi-Polling [0x16] command

Header	Address	Function	Data Length	Data Bytes		Error Check
1 Byte	1 Byte	EVN	1 Byte	Type	Data	2 Bytes

Note:
Header = SOH (01h)
Address = Device Address (Slave Machine ID)
Function = Response Function
EVN = 11H, Event Message (Active, For One Host to One Device Connection)
Data Length = Total Data Number (include of TYPE)
Data Bytes = Type : Even Type
Data : Even Data
Error Check = CRC16 check

CRC16 Generation Function (C Code)

```
unsigned short GNetPlusCRC16(unsigned char *pBuffer, int iDataLen)
{
        const CRC_PRESET=0xFFFF;
        const CRC_POLYNOM=0xA001;
        unsigned short nCRC16=CRC_PRESET;
        char i;
        while( iDataLen-- )
        {
                nCRC16 ^= *pBuffer;
                pBuffer++;
                for( i=0; i<8; i++)
                {
                        if( (nCRC16 & 1)==1 )
                                nCRC16 = (nCRC16>>1) ^ CRC_POLYNOM;
                        else
                                nCRC16 = (nCRC16>>1);
                }
        }
        return nCRC16;
}
```

Auto Mode protocol

Communications Baud Rate is 115200-8-N-1.

Device Response Package

STX	Data Bytes	Checksum	CR	LF	ETX
1 Byte	0~255 Byte	2 Bytes	1 Byte	1 Byte	1 Byte

Note:
STX = STX (02h)
Data Bytes = Data (ASCII)
Check = Check Sum form All Data Bytes (ASCII)
CR = CR (0Dh)
LF = LF (0Ah)
ETX = ETX(03h)

Device Response Package (Single channel)

STX	Data Bytes	CR	LF	ETX
1 Byte	PC + EPC	1 Byte	1 Byte	1 Byte

Device Response Package (Multi channel)

STX	Data Bytes	CR	LF	ETX
1 Byte	PC + EPC + ‘,’ + channel	1 Byte	1 Byte	1 Byte

Example :

STX	3000E20005191918020303 50EDE0,0	CR	LF	ETX

Device Response Package with Checksum (Single channel)

STX	Data Bytes	Checksum	CR	LF	ETX
1 Byte	PC + EPC	2 Bytes	1 Byte	1 Byte	1 Byte

Device Response Package with Checksum (Multi channel)

STX	Data Bytes	Checksum	CR	LF	ETX
1 Byte	PC + EPC + ‘,’ + Channel	2 Bytes	1 Byte	1 Byte	1 Byte

Example :

STX	3000E2005388880D015712209C98,031	CR	LF	ETX

Checksum:
0x31 = 0x33+0x30+0x30+0x30+0x45+0x32+0x30+0x30+0x35+0x33+0x38+0x38+0x38+0x38+0x30
+0x44+0x30+0x31+0x35+0x37+0x31+0x32+0x32+0x30+0x39+0x43+0x39+0x38+0x2C+0x30

Note:
PC = Protocol-Control (PC)
EPC = EPC of Tag
Channel = Channel At which the tag was found

Device Response DI Trigger Event Package

STX	Data Bytes	CR	LF	ETX
1 Byte	EVENT + NO	1 Byte	1 Byte	1 Byte

NO	Event Description
‘0’	DI1 trigger start inventory
‘1’	DI1 stop inventory
‘2’	DI1 trigger stop inventory by timeout
‘3’	DI3 trigger start inventory
‘4’	DI3 stop inventory

Example :

STX	EVENT0	CR	LF	ETX

DI1 trigger start inventory

UHF Query Function Code Table

	Query (Master/Host)			Response (Slave/Device)		Description
Desc	**Func**	**Len**	**Data Bytes**	**Len**	**Data Bytes**	
Get FirmWare/ Hardware ID	**10h**	1	Value	n	Return Firmware / Hardware String	Value 0：Firmware 1：Hardware
Active Mode	**12h**	1	Mode ID	1	Return mode ID	ID=0 : Auto ID=1 : Command ID=2 : Auto+Chksum ID=4: Auto + Event
Multi-Polling	**16h**	1	0x00	1	Return Total Number	Multi-RS485 for Auto Mode
Set Power Amplifier	**18h**	1	PWR	1	Replied as 0x00 rfu (reserved for further use)	PWR 0x00 : PA Off 0xFF : PA On
Write Register	**1Ah**	2/4	Address(1 byte) Data (1/3 bytes)	1	Replied as 0x00 rfu (reserved for further use)	Set Register Data to Address
Read Register	**1Ch**	1	Reg. Address	1/3	Datas(1/3 bytes)	Get Register Data form Address
Select Antenna Channel	**1Eh**	1	CN	1	CN	Set Channel
Get DI/DO Status or Control DO	**20h**	2	DO Mask (1 byte) DO Control (1 byte)	1	Return DO/DI Status	Get DO/DI status or Control DO
Write EEPROM	**22h**	n	High Addr (1 byte) Low Addr (1 byte) Data (n-2 byte)	n	Return Write to EEPROM data	Data Number < 33
Update EEPROM To Register	**22h**	3	0xFF (1 byte) 0xFF (1 byte) 0x00 (1 byte)	0		Set Register Data Form EEPROM
Read EEPROM	**24h**	3	High Addr (1 byte) Low Addr (1 byte) Length (1byte)	n	Return EEPROM data	
Inventory	**31h**	1	0x01	n	Number (1 byte) Length of EPC (1 byte) EPC (n bytes) CN (1 byte)	Reply Tag EPC
Select Tag	**33h**	n	PC(2bytes) EPC(n bytes)	1	Return Select status	0x00 : Success 0x01 : Error 0x09 : Tag Not in the Filed

	Query (Master/Host)			Response (Slave/Device)		Description
Desc	**Func**	**Len**	**Data Bytes**	**Len**	**Data Bytes**	
Write Tag Data	**35h**	n	MB(1byte) SA(1byte) AP(4bytes) DL(1bytes) DATA(n bytes)	2	Status (1 byte) Count (1 byte)	Status 0x00 : Success 0x01 : Error 0x09 : Tag Not in the Filed
Read Tag Data	**37h**	4	MB(1byte) SA(1byte) DL(1bytes) 0x00 (1byte)	n	Status (1byte) Count (1 byte) Data (n bytes)	Status 0x00 : Success 0x01 : Error 0x09 : Tag Not in the Filed
Frequency Operation	**41h**	n	Mask (1 byte) Data (n bytes)	n	Reply (n bytes)	Mask 0x01: RSSI scan 0x02: reflected power scan 0x04: add the frequency to the List 0x08: clear the List 0x10: set LBT params 0x11: Get Frequency Hopping related Parameters
Inventory with RSSI	**43h**	1	0x01		Number (1 byte) RSSI (1 byte) Frequency (3 bytes) Length of EPC (1 byte) EPC (n bytes) CN (1 byte)	
Read Bulk Register	**57h**	0		n	Data (n bytes)	Gets the complete register list
ConfigGen2	**59h**	12	Linkfreq set (1 byte) Linkfreq (1 byte) Miller set (1 byte) Miller (1 byte) Session set (1 byte) Session (1 byte) Trext set (1 byte) Trext (1 byte) Qbegin set (1 byte) Qbegin (1 byte) Sensitivity set (1 byte) Sensitivity (1 byte)	12	0x00 (1 byte) Linkfreq (1 byte) 0x00 (1 byte) Miller (1 byte) 0x00 (1 byte) Session (1 byte) 0x00 (1 byte) Trext (1 byte) 0x00 (1 byte) Qbegin (1 byte) 0x00 (1 byte) Sensitivity (1 byte)	Set Value 0x00 : Ignore 0x01 : Write

Get Firmware / Hardware ID [0x10]

Host	SOH	ADDR	0x10	0x01	ID	CRC16
Device	SOH	ADDR	0x06	Len	ASCII String	CRC16

Note:
ID = 0x00 Firmware
0x01 Hardware
Len = ASCII String Length

Example : (Firmware)

Host	0x01	0x00	0x10	0x01	0x00	0x71	0x00

Device	0x01	0xFF	0x06	0x1E						
	0x55	0x48	0x46	0x38	0x36	0x30	0x20	0x52	0x65	0x61
	0x64	0x65	0x72	0x20	0x46	0x69	0x72	0x6D	0x77	0x61
	0x72	0x65	0x20	0x56	0x31	0x2E	0x30	0x33	0x52	0x34
	0xE5	0x80								

Firmware ID = UHF860 Reader Firmware V1.03R4

Example : (Hardware)

Host	0x01	0x00	0x10	0x01	0x01	0xB1	0xC1

Device	0x01	0xFF	0x06	0x1A						
	0x55	0x48	0x46	0x38	0x36	0x30	0x20	0x52	0x65	0x61
	0x64	0x65	0x72	0x20	0x48	0x61	0x72	0x64	0x77	0x61
	0x72	0x65	0x20	0x31	0x2E	0x32				
	0x31	0x1C								

Hardware ID = UHF860 Reader Hardware V1.2

Active Mode [0x12]

Host	SOH	ADDR	0x12	0x01	Mode	CRC16
Device	SOH	ADDR	0x06	0x01	Mode	CRC16

When the device can read EPC tag

Device	STX	EPC	CR	LF	ETX		When Mode = 0x00
Device	STX	EPC	CHKSUM	CR	LF	ETX	When Mode = 0x02

Note:
Mode = 0x00 Auto Mode
0x01 GUI Mode
0x02 Auto Mode + Checksum
0x04 Auto Mode + Event
EPC = Tag EPC (ASCII)

It is only way to switch the Command or Auto mode by command !!

Example : (Standalone mode to GUI mode) standalone mode can read EPC tag

Host	0x01	0x00	0x12	0x01	0x01	0x71	0x60				
Device	0x02	0x33	0x34	0x30	0x30	0x45	0x32	0x30	0x30	0x35	0x33
		0x38	0x38	0x38	0x38	0x30	0x44	0x30	0x30	0x38	0x31
		0x31	0x32	0x32	0x30	0x39	0x42	0x36	0x38	0x2C	0x30
		0x0D	0x0A	0x03							
Device	0x02	0x33	0x34	0x30	0x30	0x45	0x32	0x30	0x30	0x35	0x33
		0x38	0x38	0x38	0x38	0x30	0x44	0x30	0x30	0x38	0x31
		0x31	0x32	0x32	0x30	0x39	0x42	0x36	0x38	0x2C	0x30
		0x0D	0x0A	0x03							
Device	0x01	0xFF	0x06	0x01	0x01	0x61	0x10				

Tag EPC = 3400-E200-5388-880D-0081-1220-9B68 @ channel 1

Example : (GUI mode to Standalone mode) standalone mode can read EPC tag

Host	0x01	0x00	0x12	0x01	0x00	0xB1	0xA1				
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1				
Device	0x02	0x33	0x34	0x30	0x30	0x45	0x32	0x30	0x30	0x35	0x33
		0x38	0x38	0x38	0x38	0x30	0x44	0x30	0x30	0x38	0x31
		0x31	0x32	0x32	0x30	0x39	0x42	0x36	0x38	0x2C	0x30
		0x0D	0x0A	0x03							

Tag EPC = 3400-E200-5388-880D-0081-1220-9B68 @ channel 1

Multi-Polling [0x16]

Host	SOH	ADDR	0x16	0x01	0x00	CRC16
Device	SOH	ADDR	0x06	0x01	CNT	CRC16

When the device have EPC tag

Device	SOH	ADDR	0x11	Len	0x49	Data	CN	CRC16

When the device not have EPC tag

Device	SOH	ADDR	0x11	0x01	0x49	CRC16

Note:

CNT = Total number of the tag
Data = Tag EPC Code
CN = channel number

Example : (No tag)

Host	0x01	0x02	0x16	0x01	0x00	0xC8	0xE1	
Device	0x01	0x02	0x06	0x01	0x00	0x0D	0xE0	Reply Tag = 0
Device	0x01	0x02	0x11	0x01	0x49	0xFB	0x21	Even ending

Device ID = 0x02 , No tag

Example : (2 Tag)

Host	0x01	0x02	0x16	0x01	0x00	0xC8	0xE1			
Device	0x01	0x02	0x06	0x01	0x02	0xCC	0x61			Reply Tag = 2
Device	0x01	0x02	0x11	0x10	0x49	0x30	0x00	0xE2	0x00	0x53
		0x88	0x88	0x0D	0x01	0x57	0x12	0x20	0x9C	0x98
		0x00	0x93	0x44	Tag1					
Device	0x01	0x02	0x11	0x10	0x49	0x34	0x00	0xE2	0x00	0xBB
		0xBB	0xAA	0xAA	0xBB	0xBB	0x66	0x66	0xBB	0xBB
		0x00	0xE3	0x07	Tag2					
Device	0x01	0x02	0x11	0x01	0x49	0xFB	0x21			Even ending

Device ID = 0x02 ,get two tags]
Tag1 = 3000E2005388880D015712209C98 @ CH = 1
Tag2 = 3400E200BBBBAAAABBBB6666BBBB @ CH = 1

Set Power Amplifier [0x18]

Host	SOH	ADDR	0x18	0x01	PWR	CRC16
Device	SOH	ADDR	0x06	0x01	0x00	CRC16

Note:
PWR = 0x00 Power Amplifier Off
0xFF Power Amplifier On

Example :

Host	0x01	0x00	0x18	0x01	0xFF	0xF3	0xC1
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1

Power Amplifier Turn On

Example :

Host	0x01	0x00	0x18	0x01	0x00	0xB3	0x81
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1

Power Amplifier Turn Off

Write Register [0x1A]

Host	SOH	ADDR	0x1A	LEN	REG	DATA	CRC16
Device	SOH	ADDR	0x06	0x01	0x00	CRC16	

Note:
LEN = REG +DATA numbers (Value = 2 or 4)
REG = Register Address (0x00 ~ 0x1E)
DATA = Register Value (1Byte or 3Bytes when register address = 0x12,0x14,0x15,0x16,0x17)

Example :

Host	0x01	0x00	0x1A	0x02	0x0C	0x00	0xF3	0xC1
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	

Write Register Address [0x0C] = 0x00

Example :

Host	0x01	0x00	0x1A	0x04	0x15	0x00	0x3F	0x20	0xDE	0xFC
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1			

Write Register Address [0x15] = 0x00 , 0x3F , 0x20

Read Register [0x1C]

Host	SOH	ADDR	0x1C	0x01	REG	CRC16
Device	SOH	ADDR	0x06	LEN	VALUE	CRC16

Note:
LEN = REG + DATA numbers (Value = 2 or 4)
REG = Register Address (0x00 ~ 0x1E)
DATA = Register Value (1Byte or 3Bytes when register address = 0x12,0x14,0x15,0x16,0x17)

Example :

Host	0x01	0x00	0x1C	0x01	0x0C	0x77	0xC0
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1

Read Register Address [0x0C] = 0x00

Example :

Host	0x01	0x00	0x1C	0x01	0x15	0xBD	0x01
Device	0x01	0xFF	0x06	0x03	0x00	0x3F	0x20	0x78	0x8C

Read Register Address [0x15] = 0x00 , 0x3F , 0x20

Select Antenna Channel [0x1E]

Host	SOH	ADDR	0x1E	0x01	CH	CRC16
Device	SOH	ADDR	0x06	0x01	CH	CRC16

Note:
CH = Select Antenna Channel
0x00 = Ch 1 , 0x01 = Ch2 , 0x02 = Ch3 , 0x03 = Ch4 (for UHF860)

Example :

Host	0x01	0x00	0x1E	0x01	0x01	0x66	0x90
Device	0x01	0xFF	0x06	0x01	0x01	0x61	0x10

Select Antenna Channel 2

Get DI/DO Status or Control DO [0x20]

Host	SOH	ADDR	0x20	0x02	MASK	CONT	CRC16
Device	SOH	ADDR	0x06	0x01	STATUS	CRC16	

Note:

MASK = Select DO Channel byte (bit = 0 not selected, bit = 1 selected)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	DO2	DO1

0x00: Just return DI/DO status
0x01: Select DO1
0x02: Select DO2
0x03: Select DO1 and DO2

CONT = Control DO Action byte (bit = 0 Off, bit =1 On)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	DO2	DO1

0x01: DO1 On
0x02: DO2 On
0x03: DO1 and DO2 On

STATUS = Return DI/DO Status byte

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	DO2	DO1	DI4	DI3	DI2	DI1

Action Table

MASK bit	CONT bit	Action
0	0	-
0	1	-
1	0	DO Off
1	1	DO On

Example :

Host	0x01	0x00	0x20	0x02	0x00	0x00	0x00	0x8E
Device	0x01	0x00	0x06	0x01	0x0F	0xB1	0xA1	

Get DI/DO status = DO1/DO2 Off, DI1/DI2/DI3/DI4 Open

Example :

Host	0x01	0x00	0x20	0x02	0x02	0x02	0xA1	0x0E
Device	0x01	0xFF	0x06	0x01	0x2F	0x69	0xA0	

Control DO2 On

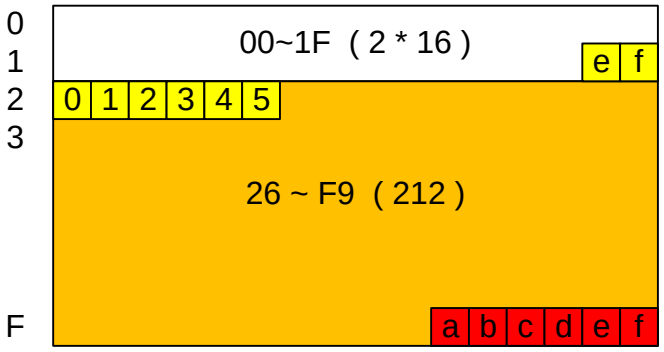
EEPROM Parameter Table

EEPROM Parameter Table is UHF Device default setting after turn on. Take Care for any change for EEPRROM Parameter .

00	**Power**
01	**Sensitivity**
02	**RX_Decode**
03	**EPC Session**
04	**Link Frequency**
05	**Antenna Scan 1/2**
06	**Antenna Scan 3/4**
07	**EPC Trext**
08	**EPC Qbegin**
09	
0A	
0B	
0C	
0D	
0E	
0F	

10	**Baud Rate**
11	**Address**
12	**Active Mode**
13	**Remove Time**
14	**Repeat Time**
15	**Hopping Delay**
16	**Standalone Interface**
17	**Wiegand Start Bit Position**
18	**Wiegand Total bits**
19	**Scan Tag Mode**
1A	**Trigger Timeout**
1B	**EPC Filter Last Start Digit**
1C	**EPC Filter Total Digit Number**
1D	
1E	**Profile**
1F	**Number of Active Channel**

20	**Listening Time high**
21	**Listening Time low**
22	**Allocation Time high**
23	**Allocation Time low**
24	**Idle Time high**
25	**Idle Time low**
26	**Frequency of CH0 high**
27	**Frequency of CH0 mid**
28	**Frequency of CH0 low**
29	**Frequency of CH0 RSSI**
2A : F9	: :
FA	Product SN0
FB	Product SN1
FC	Product SN2
FD	Product SN3
FE	Product SN4
FF	--



Item	Revsion 1.0	Default	Remark
RF Power Output Level [00]	0x00 ~ 0x13 : (-0) ~ (-19) dBm Other : 0 dBm	0xFF : 0 dBm	
RF RX Sensitivity [01]	0x00~0x7F : 0 ~ 127 dBm 0x80~0xFE : (-128) ~ (-2) dBm 0xFF : -84 dBm	0xFF : -84 dBm	When Bit7 = 1 2's complement
RX Decode [02]	0x00 : FM0 0x01 : Miller 2 0x02 : Miller 4 0x03 : Miller 8 Other : Miller 2	0xFF : Miller 2	
EPC Session [03]	0x00 : S0 0x01 : S1 0x02 : S2 0x03 : S3 0x04 : SL Other : S0	0xFF : Session = S0	
EPC Link Frequency [04]	0x00 : 40kHz 0x03 : 80kHz 0x06 : 160kHz 0x08 : 213.3kHz 0x09 : 256kHz 0x0C : 320kHz 0x0F : 640kHz Other : 160kHz	0xFF : 160kHz	
Antenna Scan 1/2 [05]	0xAB : Scan1 = chA Scan2 = chB 0xFF : Scan1,2 = Ch1 (Fix Channel)	0xFF : Scan1,2 = ch1	
Antenna Scan 3/4 [06]	0xAB : Scan3 = chA Scan4 = chB 0xFF : Scan3,4 = Ch1 (Fix Channel)	0xFF : Scan3,4 = ch1	
EPC Trext [07]	0x00 : Don't use 0x01~0xFF : Use long pilot tone	0xFF : Use long pilot tone	
EPC Qbegin [08]	0x00 ~ 0xFE : Q value 0xFF : Q = 4	0xFF : Q = 4	

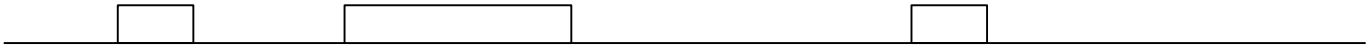
Item	Revsion 1.0	Default	Remark
Baudrate [10]	0x00~0x03: 115200 bps 0x04: 2400 bps 0x05: 4800 bps 0x06: 9600 bps 0x07: 19200 bps 0x08: 38400 bps 0x09: 57600 bps 0x0A: 115200 bps 0x0B~0xFF: 115200 bps	0xFF : 115200 bps	
Address [11]	0x01~0xFF : ADDRESS 0x00 : For Broadcast	0xFF : address	
Active Mode [12]	0x00 : GUI Mode 0x01 : Auto Mode 0x02 : Auto + Check Sum 0x04 : Auto + Event 0xFF : Auto Mode Other : No Use	0xFF : Auto Mode	
Remove Time [13]	0x00,0xFF : Not Active 0x01~0xFE : Bounce Time	0xFF : Not Active	2012/1 add (UHF860 Only)
Repeat Time [14]	0x00,0xFF : Not Active 0x01~0xFE : Repeat Time	0xFF : Not Active	2012/1 add (UHF860 Only)
Hopping Delay [15]	0x00 ~ 0x14 : Hopping Delay = 0~20 ms 0x15~0xFF : Hopping Delay = 2 ms	0xFF : 2 ms	2012/2 add (UHF860 Only)
Standalone Output Interface [16]	0x00 : Wiegand Interface 0x01 : Multi-Standalone RS485 0x02~ 0xFF : RS232/RS485 Interface	0xFF : RS232/RS485	2012/11 add (UHF860 Only)
Wiegand Start Bit Position [17]	0x00 ~ 0xFE : Wiegand From 0x00~0xFE bits Start 0xFF : Start Bit Form 0	0xFF : 0x00 Bits start	2012/11 add (UHF860 Only)
Wiegand Total bits [18]	nn: Get Wiegand nn bits	0xFF : All EPC Number	2012/11 add (UHF860 Only)

Item	Revsion 1.0	Default	Remark
Scan Tag Mode **[19]**	0x00 : DI1 Level Control 0x01 : DI1 Trigger Control with Retrigger & Level 0x02 : DI1 Trigger Control with Retrigger 0x03 : DI1 Trigger Control without Retrigger 0x04 : DI1 Trigger Control without Retrigger and Level off 0x30 : DI1Trigger Control without Retrigger and Level off + DI3 Level Control 0x05~0xFD : Reserve 0xFE : Always Scan with Anti-Passback Delay 0xFF : Always Scan	0xFF : Always Trigger	2012/11 add (UHF860 Only) The Anti-passback Delay time by set trigger timeout Note : See Page 18
Trigger Timeout **[1A]**	nn : Timeout = nn / 5 seconds FF: Timeout = 5 seconds	0xFF : 5 seconds	2012/11 add (UHF860 Only)
EPC Filter Last Start Digit **[1B]**	nn : Start digit nn form last EPC FF: Filter function disable	0xFF : Filter function disable	2012/11 add (UHF860 Only)
EPC Filter Total Digit Number **[1C]**	nn : Get nn digit number	0xFF : All of EPC	2012/11 add (UHF860 Only)
Profile **[1E]**	0x01 : Europe 0x02 : Japan 0x03 : USA 0x04 : China920 0x05 : China840 0x06 : Korea 0x07 : Taiwan 0x00 , 0x08~0xFE : User Define 0xFF : Europe	0xFF : Europe	Default by Europe
Number of Active Channel **[1F]**	0x01 ~ 0x35 : 1 ~ 53 channel 0x36 ~ 0xFF : 4 channel 0x00 : 4 channel	0xFF : 4 channel	Default by Europe
LBT Listen Time **[20][21]**	[20] : HIGH [21] : LOW But when: [20][21] = 0xFFFF = 1ms	[20] : 0xFF [21] : 0xFF Listen Time = 1ms	Default by Europe
LBT Allocation Time **[22][23]**	[22] : HIGH [23] : LOW But when: [22][23] = 0xFFFF = 10000ms	[22] : 0xFF [23] : 0xFF Listen Time = 10000ms	Default by Europe

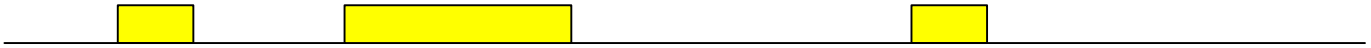
Item	Revsion 1.0	Default	Remark
LBT Idle Time [24][25]	[24] : HIGH [25] : LOW But when: [24][25] = 0xFFFF = 0ms	[24] : 0xFF [25] : 0xFF Listen Time = 0ms	Default by Europe
Frequency of Channel n **[26+n*4]** **[27+n*4]** **[28+n*4]** **[29+n*4]** **n = 0 ~ 52**	[26+n*4] = Frequency high byte of CHn [27+n*4] = Frequency mid byte of CHn [28+n*4] = Frequency low byte of CHn Frequency= [26+n*4][27+n*4][28+n*4] /1000 (Mhz) [29+n*4] : RSSI Threshold of CHn When bit7=1 is 2's complement	When Profile = 0xFF or Active Number > 53 [26+n*4][27+n*4][28+n*4] n=0~3 for Europe Frequency [29+n*4] n=0~3 for Europe RSSI [26+n*4] : 0xFF [27+n*4] : 0xFF [28+n*4] : 0xFF [29+n*4] : 0xFF	If 865.700Mhz at CH0 (n=0) [26+n*4]=[26] = 0D [27+n*4]=[27] = 35 [28+n*4]=[28] = A4 [29+n*4]=[29] = D8(-40dBm) If 902.75Mhz at CH1 (n=1) [26+n*4]=[2A] = 0D [27+n*4]=[2B] = C6 [28+n*4]=[2C] = 5E [29+n*4]=[2D] = D8(-40dBm) : : [29+n*4] When bit7=1 2's complement Example : Value= D8 : -40 dBm Value= A9 : -87 dBm
Product Serial Number [FA]~[FE]	[FA]~[FE] : Product Serial Number	[FA]~[FE] : Product Serial Number	

Digital Input Trigger Mode Timing

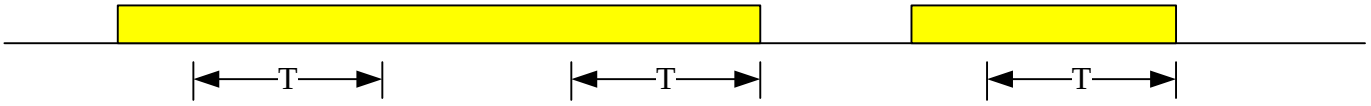
DI1 Trigger Singnal



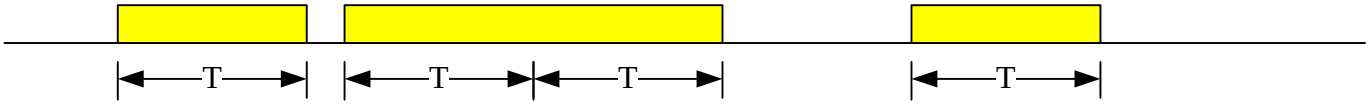
1. DI1 Level Control [SCAN MODE = 0x00]



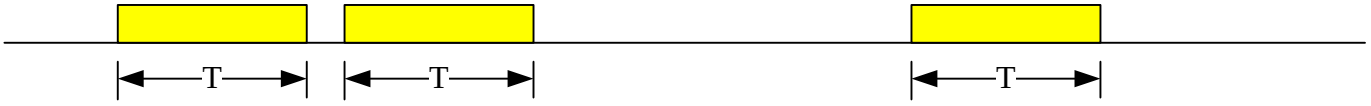
2. DI1 Trigger with Retrigger & Level [SCAN MODE = 0x01]



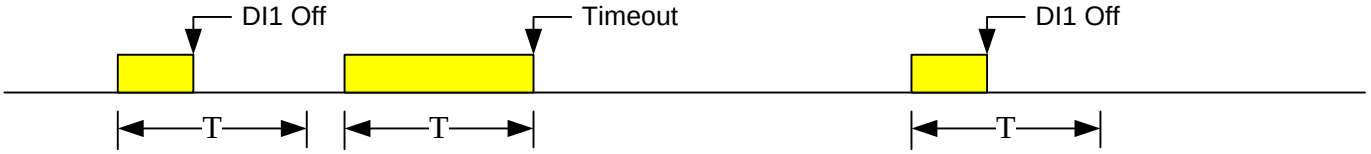
3. DI1 Trigger with Retrigger [SCAN MODE = 0x02]



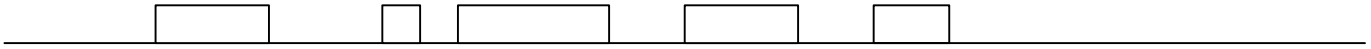
4. DI1 Trigger without Retrigger [SCAN MODE = 0x03]



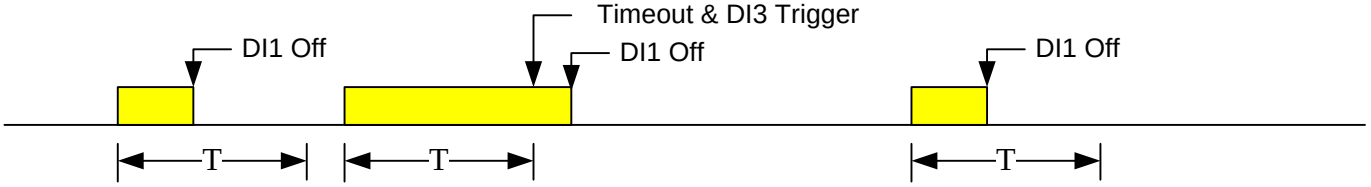
5. DI1 Trigger without Retrigger and DI Level Off [SCAN MODE = 0x04]



DI3 Trigger Singnal



6. DI1 Trigger without Retrigger and DI Level Off + DI3 Level Control [SCAN MODE = 0x30]



Write EEPROM [0x22]

Host	SOH	ADDR	0x22	Len	EEAH	EEAL	DATA	CRC16
Device	SOH	ADDR	0x06	0x01	DATA	CRC16		

Note:
EEAH = EEPROM start address high byte
EEAL = EEPROM start address low byte
DATA = Write to EEPROM , maximum 32 Bytes

Example : (Single address write)

Host	0x01	0x00	0x22	0x03	0x00	0x11	0x01	0xC8	0x35
Device	0x01	0x00	0x06	0x01	0x01	0x75	0x20		

Write EEPROM address [0x11] = 0x01 (Device ID = 0x01)

Example : (Continuous address write)

Host	0x01	0x00	0x22	0x05	0x00	0x09	0x01	0x02	0x03	0x61	0x51
Device	0x01	0x01	0x06	0x03	0x01	0x02	0x03	0x2F	0x99		

Write EEPROM address [0x09] = 0x01, [0x0A] = 0x02 , [0x0B] = 0x03

*** All setting value is valid not yet before run" Update EEPROM To Register" command or reset power !!**

Update EEPROM To Register [0x22]

Host	SOH	ADDR	0x22	0x03	0xFF	0xFF	0x00	CRC16
Device	SOH	ADDR	0x06	0x00	CRC16			for ACK
Device	SOH	ADDR	0x06	0x00	CRC16			for Update Finish

Note:
All setting value is valid after this command is send.

Example : (Single address write)

Host	0x01	0x00	0x22	0x03	0xFF	0xFF	0x00	0x98	0x89
Device	0x01	0x01	0x06	0x00	0xA0	0x23			for ACK
	Updating……								
Device	0x01	0x01	0x06	0x00	0xA0	0x23			for Update Finish

Update to Register

Read EEPROM [0x24]

Host	SOH	ADDR	0x24	0x03	EEAH	EEAL	CNT	CRC16
Device	SOH	ADDR	0x06	Len	DATA			CRC16

Note:
EEAH = EEPROM start address high byte
EEAL = EEPROM start address low byte
CNT = Read data counter, Maximum 32
DATA = Read form EEPROM

Example : (Single address read)

Host	0x01	0x00	0x24	0x03	0x00	0x11	0x01	0xC8	0xBD
Device	0x01	0x01	0x06	0x01	0x01	0x89	0x21		

Read EEPROM address [0x11] = 0x01 (Device ID = 0x01)

Example : (Continuous address read)

Host	0x01	0x00	0x24	0x03	0x00	0x09	0x03	0x09	0x36
Device	0x01	0x01	0x06	0x03	0x01	0x02	0x03	0x2F	0x99

Read EEPROM address [0x09] = 0x01, [0x0A] = 0x02 , [0x0B] = 0x03

Select Tag [0x33]

Host	SOH	ADDR	0x33	Len	CNT	EPC	CRC16
Device	SOH	ADDR	0x06	0x01	RES	CRC16	

Note:
CNT = EPC byte counter
EPC = select tag EPC
RES = 0x00 : Success , 0x01 : Error, 0x09 : Tag Not in the Filed

Example : (EPC = E2000519191802030350EDE0h ***)***

Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
	0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D			
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Device	0x01	0xFF	0x06	0x01	0x09	0xA7	0x11	Fail			

Select Tag EPC = (E2000519191802030350EDE0) h

Write Tag Data [0x35]

Host	SOH	ADDR	0x35	Len	MB	SA	AP	DL	DATA	CRC16

Device	SOH	ADDR	0x06	0x02	RES	CNT	CRC16

Note:

MB	= Memory Bank , 0x00 : Reserve , 0x01 : EPC , 0x02 : TID , 0x03/0x83 : User Bank
SA	= Start address of word size (when MB = 0x83 , SA length 2 bytes)
AP	= Access Password (4 bytes) , 0x00000000 not need access password
DL	= Write date count of word size, maximum length = 0x1D
DATA	= Write Data to tag , Total count = (2 x DL) bytes
RES	= 0x00 : Write Success , 0x01 : Write Incomplete , 0x09 : Tag Not in the Filed
CNT	= Real write word success counter

Example : (** Write process not smoothly **)

Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
	0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D	Select		
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Host	0x01	0xFF	0x35	0x0F	0x03	0x04	0x00	0x00	0x00	0x00	0x04
	0x11	0x22	0x33	0x44	0x55	0x66	0x77	0x88	0x24	0xA6	
Device	0x01	0xFF	0x06	0x02	0x01	0x00	0xCC	0x90	Write error , 0 word be write		
Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
	0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D	Select		
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Host	0x01	0xFF	0x35	0x0F	0x03	0x04	0x00	0x00	0x00	0x00	0x04
	0x11	0x22	0x33	0x44	0x55	0x66	0x77	0x88	0x24	0xA6	
Device	0x01	0xFF	0x06	0x02	0x01	0x01	0x0C	0x51	Write error , 1 word be write		
Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
	0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D	Select		
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Host	0x01	0xFF	0x35	0x0D	0x03	0x05	0x00	0x00	0x00	0x00	0x03
	0x33	0x44	0x55	0x66	0x77	0x88	0xDC	0x5F			
Device	0x01	0xFF	0x06	0x02	0x01	0x02	0x0D	0x11	Write error , 2 word be write		
Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
	0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D	Select		
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Host	0x01	0xFF	0x35	0x09	0x03	0x07	0x00	0x00	0x00	0x00	0x01
	0x77	0x88	0xB4	0x2E							
Device	0x01	0xFF	0x06	0x02	0x00	0x01	0x9C	0x50	Write success , 1 word be write		

Write 0x11,0x22 to [0x04] / 0x33,0x44 to [0x05] / 0x55,0x66 to [0x06] / 0x77,0x88 to [0x07]

Example : (Write process smoothly)

Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
		0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D	Select	
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Host	0x01	0xFF	0x35	0x0F	0x03	0x04	0x00	0x00	0x00	0x00	0x04
		0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xEB	0x06
Device	0x01	0xFF	0x06	0x02	0x00	0x04	0x9F	0x90	Write error , 0 word be write		

Write 0x11,0x22 to [0x04] / 0x33,0x44 to [0x05] / 0x55,0x66 to [0x06] / 0x77,0x88 to [0x07]

Read Tag Data [0x37]

Host	SOH	ADDR	0x37	0x04	MB	SA	DL	0x00	CRC16

Device	SOH	ADDR	0x06	Len	RES	DL	DATA	CRC16

Note:

MB = Memory Bank , 0x00 : Reserve , 0x01 : EPC , 0x02 : TID , 0x03/0x83 : User Bank
SA = Start address of word size (when MB = 0x83 , SA length 2 bytes)
DL = Read data count of word size, maximum length = 0x1D
RES = Reply, 0x00 : Transfer Buffer full , 0x08 : Not yet finish , 0x83 : Finish or Not available
DATA = Read Data form Tag , Total count = (2 x DL) bytes

Example : (Read User Bank)

Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
		0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D	Select	
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Host	0x01	0xFF	0x37	0x04	0x03	0x00	0x00	0x00	0xCC	0xE1	Read form 0x00
Device	0x01	0xFF	0x06	0x3C	0x00	0x1D	0x00	0x00	0x00	0x00	0x00
		0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
		0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
		0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
		0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
		0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
		0x00	0x00	0x00	0x36	0x96	Success , but buffer full , next				
Host	0x01	0xFF	0x33	0x0D	0x0C	0xE2	0x00	0x05	0x19	0x19	0x18
		0x02	0x03	0x03	0x50	0xED	0xE0	0xE5	0x4D	Select	
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1	Success			
Host	0x01	0xFF	0x37	0x04	0x03	0x1D	0x00	0x00	0xCA	0x71	Read form 0x1D
Device	0x01	0xFF	0x06	0x08	0x83	0x03	0x00	0x00	0x00	0x00	0x00
		0x00	0x00	0x59	0xC9	Finish					

Preliminary

Frequency Operation [0x41]

Host	SOH	ADDR	0x41	Len	Mask	DATA	CRC16
Device	SOH	ADDR	0x06	Len	REPLY		CRC16

Note:

With the mask byte, it is possible to select either the RSSI value that is scanned with no carrier (LBT) or the reflected power that is received with activated carrier.

Mask 0x01: Get Frequency RSSI
Mask 0x02: Get Frequency Reflected Power
Mask 0x04: Add the frequency to the List
Mask 0x08: Clear the List and add a frequency
Mask 0x10: Set Frequency Hopping Related Parameters
Mask 0x11: Get Frequency Hopping Related Parameters

Get Frequency RSSI (Mark = 0x01)

Host	SOH	ADDR	0x41	0x04	0x01	$FREQ_L$	$FREQ_M$	$FREQ_H$	CRC16
Device	SOH	ADDR	0x06	0x03	I	Q	dBm	CRC16	

Note:

FREQ = (3 bytes) Target frequency
I = RSSI (I) value
Q = RSSI (Q) value
dBm = Sensitivity (Signed Byte)

Example : (922.5MHz RSSI)

Host	0x01	0xFF	0x41	0x04	0x01	0x84	0x13	0x0E	0xF0	0x28
Device	0x01	0xFF	0x06	0x03	0x00	0x00	0x80	0xF0	0x9D	

Get Frequency Reflected Power (Mark = 0x02)

Host	SOH	ADDR	0x41	0x04	0x02	$FREQ_L$	$FREQ_M$	$FREQ_H$	CRC16
Device	SOH	ADDR	0x06	0x02	MA	MB	CRC16		

Note:

FREQ = (3 bytes) Target frequency
MA = Rec. A mixer DC byte (Signed Byte)
MB = Rec. B mixer DC byte (Signed Byte)

Reflected Power = $20 \times \log (\sqrt{MA^2 + MB^2} / G)$

G = A constant depending on the Settings of the lower 3 bits in Register (*0Ah*).

Register 0x0A	G Value
xxxx x001	0x0A
xxxx x000	0x1A
xxxx x011	0x5A
xxxx x010	0x19
xxxx x101	0x2E

Reference Page 10 : Read Register [0x1C]

Example :

Host	0x01	0xFF	0x41	0x04	0x02	0xD8	0x05	0x0E	0xC9	0xE9
Device	0x01	0xFF	0x06	0x02	0x1B	0x29	0x72	0x5A		

Reflected Power = 20 x log ($\sqrt{27^2 + 41^2}$ / 26) = 5.521 dBm @ 919.0Mhz

Register [0x0A] = 0x08 ; G = 0x1A = 26

Add the frequency to the List (Mark = 0x04)

Host	SOH	ADDR	0x41	0x05	0x04	FREQL	FREQM	FREQH	RTH	CRC16
Device	SOH	ADDR	0x06	0x02	RES		CRC16			

Note: FREQL/M/H = (3 bytes) Add frequency
RTH = RSSI thresh hold (dBm)
RES = 0xFCFF : Success
0x0000 : Error (Over maximum frequency)

Example :

Host	0x01	0x00	0x41	0x05	0x04	0x84	0x13	0x0E	0xD8	0x95	0x3D
Device	0x01	0xFF	0x06	0x02	0xFC	0xFF	0x1C	0x90			

Add Frequency = 0x0E1384 = 922.5MHz ; RSSI = 0xD8 = -40 dBm

Note: These Settings will be lost after turn off , and return to the setting of the EEPROM !!

Clear the List and Create (Mark = 0x08)

Host	SOH	ADDR	0x41	0x06	0x08	FREQL	FREQM	FREQH	RTH	PID	CRC16
Device	SOH	ADDR	0x06	0x02	RES		CRC16				

Note:
FREQL/M/H = (3 bytes) Add frequency
RTH = RSSI thresh hold (dBm)
PID = Profile ID
RES = 0xFEFF : Success

Example :

Host	0x01	0x00	0x41	0x06	0x08	0x84	0x13	0x0E
	0xD8	0x07	0xDF	0x26				
Device	0x01	0xFF	0x06	0x02	0xFE	0xFF	0x7C	0x91

Clear frequency list & Create frequency = 0x0E1384 = 922.5MHz ; RSSI = 0xD8 = -40 dBm ; Profile ID = 7

Note: These Settings will be lost after turn off , and return to the setting of the EEPROM !!

Set Frequency Hopping Related Parameters (Mark = 0x10)

Host	SOH	ADDR	0x41	0x07	0x10	LSTNL	LSTNH	LOCAL	LOCAH
		IDLEL	IDLEH	CRC16				

Device	SOH	ADDR	0x06	0x02	RES	CRC16

Note:

LSTNL = Listening Time low byte
LSTNH = Listening Time high byte
LOCAL = Maximum Allocation Time low byte
LOCAH = Maximum Allocation Time high byte
IDLEL = Idle Time low byte
IDLEH = Idle Time high byte
RES = 0xFEFF : Success
0xFFFF : maximum allocation time < 50 and then maximum allocation time = 50

Example :

Host	0x01	0xFF	0x41	0x07	0x10	0x01	0x00	0x10	0x27
		0x00	0x00	0x69	0xA9				

Device	0x01	0xFF	0x06	0x02	0xFE	0xFF	0x7C	0x91

Note:

Listening Time = 0x0001 = 1 ms
Maximum Allocation Time = 0x2710 = 10 s
Idle Time = 0x0000 = 0 s

Note: These Settings will be lost after turn off , and return to the setting of the EEPROM !!

Get Frequency Hopping related Parameters (Mark = 0x11)

Host	SOH	ADDR	0x41	0x01	0x11	CRC16

Device	SOH	ADDR	0x06	0x12	0xFE	0xFF	PID	LSTNL	LSTNH	LOCAL	LOCAH
		IDLEL	IDLEH	MINFL	MINFM	MINFH	MAXFL	MAXFM	MAXFH	NUM	RTH
		ANUM	CRC16								

Note:

PID	= Profile ID
LSTNL	= Listening Time low byte
LSTNH	= Listening Time high byte
LOCAL	= Maximum Allocation Time low byte
LOCAH	= Maximum Allocation Time high byte
IDLEL	= Idle Time low byte
IDLEH	= Idle Time high byte
MINFL	= Minimum Frequency low byte in the frequency list used for hopping
MINFM	= Minimum Frequency middle byte in the frequency list used for hopping
MINFH	= Minimum Frequency high byte in the frequency list used for hopping
MAXFL	= Maximum Frequency low byte in the frequency list used for hopping
MAXFM	= Maximum Frequency middle byte in the frequency list used for hopping
MAXFH	= Maximum Frequency high byte in the frequency list used for hopping
NUM	= Maximum in the frequency list used for hopping
RTH	= RSSI thresh hold (dBm)
ANUM	= Real Number in the frequency list used for hopping

Example :

Host	0x01	0xFF	0x41	0x01	0x11	0xB8	0xA1

Device	0x01	0xFF	0x06	0x12	0xFE	0xFF	0x01	0x01	0x00	0x10	0x27
		0x00	0x00	0xA4	0x35	0x0D	0xAC	0x3C	0x0D	0x04	0xD8
		0x04	0x81	0xDA							

Note:
Profile ID = 0x01
Listening Time = 0x0001 = 1 ms
Maximum Allocation Time = 0x2710 = 10 s
Idle Time = 0x0000 = 0 s
Minimum Frequency = 0x0D35A4 /1000 = 865.7 Mhz
Maximum Frequency = 0x0D3CAC /1000 = 867.5 Mhz
Frequency list number = 4
RSSI thresh hold = 0xD8 = -40 dBm
Real Frequency list number = 4

Inventory [0x31]

Host	SOH	ADDR	0x31	0x01	0x01	CRC16		
Device	SOH	ADDR	0x06	Len	NUM	NB	EPC	CN	CRC16

Note:

NUM = Number of found tags
NB = Length of EPC byte Plus PC
EPC = PC plus EPC
CN = Channel At which the tag was found (0x00:ch1,0x01:ch2……)

Example : (No Tag)

Host	0x01	0xFF	0x31	0x01	0x01	0xAF	0xA1
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1

Number of Found Tags = 0

Example : (Read one Tag)

Host	0x01	0xFF	0x31	0x01	0x01	0xAF	0xA1				
Device	0x01	0xFF	0x06	0x11	0x01	0x0E	0x30	0x00	0xE2	0x00	0x05
	0x19	0x19	0x18	0x02	0x03	0x03	0x50	0xED	0xE0	0x00	
	0x38	0xAD									

Number of Found Tags = 1
EPC = (3000E200051919180203035OEDE0) h
CN = 1

Example : (Read two Tags)

Host	0x01	0xFF	0x31	0x01	0x01	0xAF	0xA1				
Device	0x01	0xFF	0x06	0x11	0x02	0x0E	0x34	0x00	0xE2	0x00	0x53
	0x88	0x88	0x0D	0x00	0x81	0x12	0x20	0x9B	0x68	0x00	
	0x2D	0x6F									
Device	0x01	0xFF	0x06	0x11	0x01	0x0E	0x30	0x00	0xE2	0x00	0x05
	0x19	0x19	0x18	0x02	0x03	0x03	0x50	0xED	0xE0	0x00	
	0x38	0xAD									

Number of Found Tags = 2
Tag 1:
EPC = (3400E2005388880D008112209B68) h
CN = 1

Tag 2:
EPC = (3000E200051919180203035OEDE0) h
CN = 1

Inventory with RSSI [0x43]

Host	SOH	ADDR	0x43	0x01	0x01	CRC16					
Device	SOH	ADDR	0x06	0x03	NUM	RSSI	FREQ	NB	PC+EPC	CN	CRC16

Note:

NUM	= Number of found tags
RSSI	= RSSI value
FREQ	= (3 bytes) Base frequency at which the tag was found
NB	= Length of EPC byte Plus PC
PC+EPC	= PC plus EPC
CN	= Channel At which the tag was found (0x00:ch1,0x01:ch2……)

Example : (No Tag)

Host	0x01	0xFF	0x43	0x01	0x01	0xB4	0x01
Device	0x01	0xFF	0x06	0x01	0x00	0xA1	0xD1

Number of Found Tags = 0

Example : (Read one Tag)

Host	0x01	0xFF	0x43	0x01	0x01	0xB4	0x01				
Device	0x01	0xFF	0x06	0x15	0x01	0x0F	0x84	0x13	0x0E	0x0E	0x30
	0x00	0xE2	0x00	0x05	0x19	0x19	0x18	0x02	0x03	0x03	
	0x50	0xED	0xE0	0x00	0x72	0x2C					

Number of Found Tags = 1
EPC = (3000E200051919180203050EDE0) h
Frequency = (0E1384)h /1000 = 922.500 Mhz
RSSI = 0x0F /0XFF = 15/255= 5.88%
CN = 1

Example : (Read two Tags)

Host	0x01	0xFF	0x43	0x01	0x01	0xB4	0x01				
Device	0x01	0xFF	0x06	0x15	0x02	0x0F	0x84	0x13	0x0E	0x0E	0x34
	0x00	0xE2	0x00	0x53	0x88	0x88	0x0D	0x00	0x81	0x12	
	0x20	0x9B	0x68	0x00	0x2D	0x6F					
Device	0x01	0xFF	0x06	0x15	0x01	0x0F	0x84	0x13	0x0E	0x0E	0x30
	0x00	0xE2	0x00	0x05	0x19	0x19	0x18	0x02	0x03	0x03	
	0x50	0xED	0xE0	0x00	0x72	0x2C					

Number of Found Tags = 2
Tag 1:
EPC = (3400E2005388880D008112209B68) h
Frequency = (0E1384)h /1000 = 922.500 Mhz
RSSI = 0x0F /0XFF = 15/255= 5.88%
CN = 1

Tag 2:
EPC = (3000E200051919180203050EDE0) h
Frequency = (0E1384)h /1000 = 922.500 Mhz
RSSI = RSSI = 0x0F /0XFF = 15/255= 5.88%
CN = 1

Read Bulk Register [0x57]

Host	SOH	ADDR	0x57	0x00	CRC16
Device	SOH	ADDR	0x06	0x2A	VALUE	CRC16

Note:
VALUE = All of Register (0x00 ~ 0x1F)
Total number = 1Byte * 27 + 3Bytes * 5 = 42 Bytes

Example :

Host	0x01	0x00	0x57	0x00	0xF0	0x4F

Device	0x01	0xFF	0x06	0x2A						
	0x0E	0x09	0xF0	0x90	0x41	0x83	0x00	0x05	0x04	0x37
	0x4A	0x02	0x00	0x37	0x13	0x00	0x88	0x02	0x0C	0x40
	0x00	0x51	0x83	0x84	0x0A	0x00	0x3F	0x20	0x1A	0xED
	0xE3	0x46	0x18	0x01	0x00	0x00	0x00	0x00	0x00	0x00
	0x00	0x00								
	0xD3	0x51								

Read Register Address [0x00 ~ 0x1F]

ConfigGen2 [0x59]

Host	SOH	ADDR	0x59	Len	LF/S	LF	MR/S	MR	SN/S	SN
		TR/S	TR	QB/S	QB	SEN/S	SEN	CRC16		
Device	SOH	ADDR	0x06	Len	0x00	LF	0x00	MR	0x00	SN
		0x00	TR	0x00	QB	0x00	SEN	CRC16		

Note:

LF/S = Link Frequency Setting Flag (0x00 : Read / 0x01 : Write)
LF = Link Frequency
MR/S = Miller Setting Flag (0x00 : Read / 0x01 : Write)
MR = Miller
SN/S = Session Setting Flag (0x00 : Read / 0x01 : Write)
SN = Session
TR/S = Trext Setting Flag (0x00 : Read / 0x01 : Write)
TR = Trext
QB/S = Qbegin Setting Flag (0x00 : Read / 0x01 : Write)
QB = Qbegin
SEN/S = Sensitivity Setting Flag (0x00 : Read / 0x01 : Write)
SEN = Sensitivity

The " Setting Flag" bytes define if the subsequent byte should be set in the reader firmware. The answer To this command returns all the actually set values. This allows reading out the values without Actually changing anything.

Values for the different parameters are:

Name	Value	Default
Link Frequency	0x00 = 40 kHz 0x06 =160 kHz 0x08 = 213 kHz 0x09 = 256 kHz 0x12 = 320 kHz 0x15 = 640 kHz	256 kHz
Miller	0x00 = FM0 0x01 = Miller 2 0x02 = Miller 4 0x03 = Miller 8	Miller 2
Session	0x00 = S0 0x01 = S1 0x02 = S2 0x03 = SL	S0
Trext	0x00 = short preamble, pilot tone 0x01 = long preamble, pilot tone	short preamble
Qbegin	0 ~ 8 = Initial gen2 round is 2^Qbegin long	4
Sensitivity	-128 ~ 127 (dBm)	-84

Example : (Get GEN2)

Host	0x01	0xFF	0x59	0x0C	0x00	0x00	0x00	0x00	0x00	0x00	0x00
	0x00	0x00	0x00	0x00	0x00	0x6A	0xB7				
Device	0x01	0xFF	0x06	0x0C	0x00	0x06	0x00	0x01	0x00	0x00	0x00
	0x00	0x00	0x04	0x00	0xAC	0x91	0xBA				

Note:
Link Frequency = 160 kHz
Miller = Miller 2
Session = S0
Trext = short preamble, pilot tone
Qbegin = 4
Sensitivity = - 84 dBm

Example : (set Miller 4)

Host	0x01	0xFF	0x59	0x0C	0x00	0x00	0x01	0x02	0x00	0x00	0x00
	0x00	0x00	0x00	0x00	0x00	0xCF	0xFF				
Device	0x01	0xFF	0x06	0x0C	0x00	0x06	0x00	0x02	0x00	0x00	0x00
	0x00	0x00	0x04	0x00	0xAC	0x61	0xAE				

Note:
Link Frequency = 160 kHz
Miller = Miller 4
Session = S0
Trext = short preamble, pilot tone
Qbegin = 4
Sensitivity = - 84 dBm

Example : (set Sensitivity = -71 dBm)

Host	0x01	0xFF	0x59	0x0C	0x00	0x00	0x00	0x00	0x00	0x00	0x00
	0x00	0x00	0x00	0x01	0xB9	0x48	0x77				
Device	0x01	0xFF	0x06	0x0C	0x00	0x06	0x00	0x02	0x00	0x00	0x00
	0x00	0x00	0x04	0x00	0xB9	0xAE	0x6F				

Note:
Link Frequency = 160 kHz
Miller = Miller 4
Session = S0
Trext = short preamble, pilot tone
Qbegin = 4
Sensitivity = - 71

Note: These Settings will be lost after turn off , and return to the setting of the EEPROM !!