

## Content

I Communication Protocol.....	4
1. Format.....	4
1.1 UART format:.....	4
1.2 Frame format:.....	4
1.3 Checksum calculation.....	5
2. Command description.....	6
2.1 Channel changing command 0x01.....	6
2.2 Volume setting command 0x02.....	7
2.3 Check module status 0x04.....	7
2.4 RSSI 0x05.....	8
2.5 Call out/Call In 0x06 & Query call in contact 0x10.....	8
2.6 SMS TX or RX command 0x07 & Query SMS content 0x11.....	10
2.7 Emergency alarm 0x09.....	12
2.8 Setting MIC gain command 0x0B.....	12
2.9 Enter/exit duty working mode command 0x0C.....	13
2.10 Set TX/RX Frequency 0x0D.....	14
2.11 Enter/exit repeater mode 0x0E.....	14
2.12 SQ setting command 0x12.....	15
2.13 Select CTCSS / CDCSS type command 0x13.....	15
2.14 Select CTCSS / CDCSS code 0x14.....	16
2.15 Set TX power 0x17.....	17
2.16 Set contact 0x18.....	17
2.17 Encryption on/off command 0x19.....	18
2.18 Check initialization status 0x1A.....	19
2.19 Check the contact ID 0x22.....	19
2.20 Check the radio ID 0x24.....	19
2.21 Check the firmware version 0x25.....	20
2.22 Check the encryption status 0x28.....	20

2.23 Add contact ID into RX group list 0x29.....	21
2.24 Clear a RX group list 0x30.....	21
2.25 Set radio ID 0x1B.....	22
2.26 Set the color code 0x31.....	22
2.27 Set bandwidth 0x32.....	23
2.28 Set time slot 0x33.....	23
2.29 On/off tone 0x1C.....	24
2.30 Check the parameters of current channel 0x1D.....	24
2.31 Reset to default parameters 0xF0.....	26
2.32 Software reset 0xF2.....	26
II Call Type Details.....	27
1. Communication Parameter Requirements.....	29
2. Relevant Command Description.....	30
III Function Description.....	31
1. Power on and Sleep.....	31
1.1 Power on.....	31
1.2 Sleep.....	31
2. Calling.....	31
2.1 Private call.....	31
2.2 Group Call.....	32
2.3 All Call.....	34
3. SMS.....	34
3.1 Private SMS.....	34
3.2 Group SMS.....	35
4. Emergency Alarm.....	36
5. Encryption.....	36
6. Duty Working Mode.....	36
7. Repeater.....	37
Appendix 1 CXCSS Code.....	38
Appendix 2: Command Properties.....	41

---

Appendix 3: Default Parameters.....	43
-------------------------------------	----

## I Communication Protocol

DMR818S provides standard UART interface to send the command to modify and read the parameters.

### 1. Format

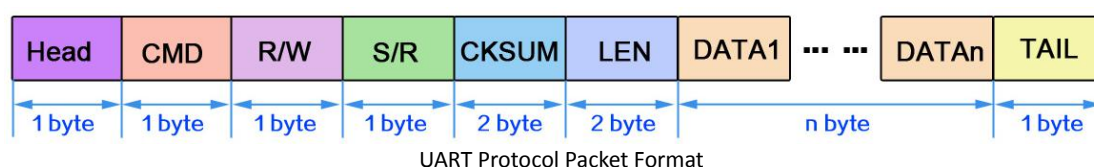
#### 1.1 UART format:

Baud Rate: 57600 bps    Data Bit: 8    Stop: 1    Parity: NONE

#### 1.2 Frame format:

All instructions in the protocol are transmitted in hex, MSB. Begin with 0x68 (Head) and end with 0x10 (Tail).

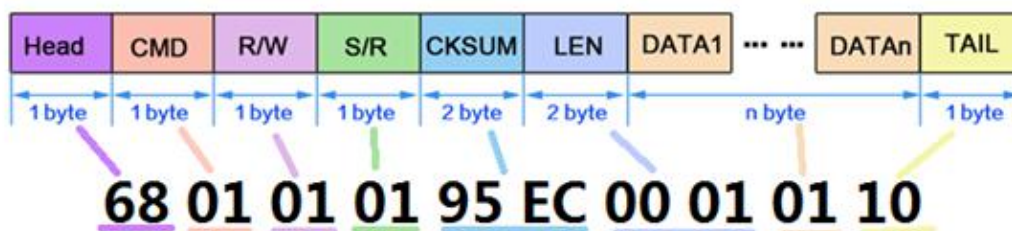
The specific format is as follows:



Offset	Flag	Length	Comment	Detail
0	Head	1	Header	0x68
1	CMD	1	Command	
2	R/W	1	Read/Write operation	0x00: Reading 0x01: Writing 0x02: Initiative sending
3	S/R	1	Setting/Responding	While setting: 0x01: Start setting While responding 0x00: Done 0x01: Busy or fail 0x02: No channel or channel errors 0x09: Check error Note: For SMS and voice, see the detailed instructions in the corresponding chapters below.
4、5	CHKSUM	2	Checksum	Checksum of the frame
6、7	LEN	2	Data length	Data length, LEN is 0 when no data
8	DATA	len	Data info	
	TAIL	1	Tail	0x10

Uart protocol field definition

Eg.



注:

Abnormal of S/R corresponding reason:	
0x01 Busy or fail	The input parameter is out of range or the function works abnormally
0x02 No channel or channel errors	The current channel doesn't support this command, such as setting CXCSS on DMR channel or setting contact on analog channel. Refer to Appendix 2 for the channel properties corresponding to the commands.
0x09 Check error	Checksum error in frame

### 1.3 Checksum calculation

The calculation method is: each two bytes form a 16-bit number, which is added in turn, and the obtained value after the XOR is the checksum of the data frame. If the length of the data frame is singular, add 0x00 after the last byte to form 16 bits and add them.

Code show as below:

```
uint16 PcChecksum(uint8 * buf, int16 len)
{
    uint32 sum=0;
    while(len >1)
    {
        sum += 0xFFFF & (*buf<<8 | *(buf+1));
        buf+=2;
        len-=2;
    }
    if (len)
    {
        sum += (0xFF & *buf)<<8;
    }
    while (sum>>16)
    {
        sum = (sum & 0xFFFF)+(sum >> 16);
    }
    return( (uint16) sum ^ 0xFFFF);
}
```

Eg

Below is the calculation process of command "68 01 01 01 95 EC 00 01 01 10".



NOTE: Normally checksum is not 0x0000. If user don't want to calculate the checksum, it can send the checksum as 0x0000, the module will ignore the checksum.

## 2. Command description

### 2.1 Channel changing command 0x01

#### 2.1.1 Instruction

**Format:** 68 01 01 01 CHKSUM 00 01 Channel 10

**Parameters:**

**Channel:** 1 byte, Channel number(0x01~0x10)

**Example:**

Change to channel 1.

68 01 01 01 95 EC 00 01 01 10

#### 2.1.2 Response

**Format:** 68 01 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R :** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Channel setting done.

68 01 00 00 87 FE 00 00 10

There are 16 channels in module. By default, Channels 1~8 are DMR channels, and 9~16 are analog channels. The default channel parameters are as follows.

Channel	Type	TX Frequency(MHz)	RX Frequency(MHz)
1	DMR	418.125	418.125
2	DMR	419.125	419.125
3	DMR	420.125	420.125
4	DMR	421.125	421.125
5	DMR	422.125	422.125
6	DMR	423.125	423.125
7	DMR	424.125	424.125
8	DMR	425.125	425.125
9	Analog	418.125	418.125
10	Analog	419.125	419.125
11	Analog	420.125	420.125
12	Analog	421.125	421.125
13	Analog	422.125	422.125
14	Analog	423.125	423.125
15	Analog	424.125	424.125
16	Analog	425.125	425.125

**Note:** The types of channel need to be changed in PC software.

## 2.2 Volume setting command 0x02

### 2.2.1 Instruction

**Format:** 68 02 01 01 CHKSUM 00 01 Volume 10

**Parameters:**

**Volume :** 1 byte, Volume level(0x01~0x09). The higher the value, the louder the volume.

**Example:**

Setting volume to level 9 .

68 02 01 01 8D EB 00 01 09 10

### 2.2.2 Response

**Format:** 68 02 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Setting volume done.

68 02 00 00 87 FD 00 00 10

## 2.3 Check module status 0x04

### 2.3.1 Instruction

**Description:** Check the status of the module if it is transmitting or receiving or standby.

**Format:** 68 04 01 01 CHKSUM 00 01 01 10

**Example:**

68 04 01 01 95 E9 00 01 01 10

### 2.3.2 Response

**Description:** Return the status of module

**Format:** 68 04 00 S/R CHKSUM 00 01 Status 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Status:** 1 byte, the status of module

0x01-> Receiving, received a call and is outputting audio

0x02-> Transmitting, calling other walkie-talkie

0x03-> Standby(no calling or receiving a call), detecting signal.

**Example:**

Module is in standby mode.

68 04 00 00 94 EA 00 01 03 10

## 2.4 RSSI 0x05

### 2.4.1 Instruction

**Description:** Read the RSSI of the module.

**Format:** 68 05 01 01 CHKSUM 00 01 01 10

**Example:**

68 05 01 01 95 E8 00 01 01 10

### 2.4.2 Response

**Description:** Return the RSSI.

**Format:** 68 05 00 S/R CHKSUM 00 01 RSSI 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**RSSI:** 1 byte, signal strength level(0x00~0x7F), The higher the value, the stronger of the signal.

**Example:**

Current RSSI is 3.

68 05 00 00 94 E9 00 01 03 10

## 2.5 Call out/Call In 0x06 & Query call in contact 0x10

### 2.5.1 Start/stop calling out instruction

**Description:** Start or stop calling out, same as PTT pressed or released.

**Format:** 68 06 01 S/R CHKSUM 00 04 Call\_type Call\_ID 10

**Parameters:**

**S/R :** 1byte, Flag of setting



0x01 -> Start calling

0xFF -> Stop calling

**Call\_type:** 1byte, Calling type

0x00-> Call on analog channels

0x01-> Private call

0x02-> Group call

0x04-> All call

**Call\_ID** : 3 bytes, contact ID to be called. Fixed to 0x000000 when the current channel is an analog channel.

**Example:**

Calling in DMR channels.

Group call to call ID 0x000001.

68 06 01 01 84 F3 00 04 02 00 00 01 10

Calling in analog channels.

Call\_type and Call\_ID are 0.

68 06 01 01 86 F4 00 04 00 00 00 00 10

### 2.5.2 Response of start/stop calling out errors

**Description:** Response when there is an error in the instruction of start/stop calling.

**Format:** 68 06 01 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x01 -> Busy or fail. Call\_type or Call\_ID out of range under DMR channels.

0x02 -> Channel errors. Call\_type is not 0 under analog channels.

0x09 -> Checksum error

**Example:**

Checksum error.

68 06 00 09 87 F0 00 00 10

### 2.5.3 Upload calling out/called in state 1(without parameters)

**Description:** After calling out or being called ends, module uploads status via serial port.

**Format:** 68 06 02 S/R CHKSUM 00 00 10

**Parameters:**

**S/R** : 1 byte, Flag of uploading

0x62 -> Calling out ends (PTT released or stop calling out is sent out by command 0x06)

0x6D -> Calling out fails

0x6F -> Being called ends (When the module is receiving a call, the other side stop calling.)

**Example:**

Calling out ends.

68 06 02 62 85 97 00 00 10

### 2.5.4 Upload calling out/called in state 1(with parameters)

**Description:** After calling out or being called starts, module uploads status via serial port

**Format:** 68 06 02 S/R CHKSUM 00 04 Call\_type Call\_ID 10

**Parameters:**

**S/R** : 1 byte, Flag of uploading

0x60 -> Being called starts(The other side start calling this module)

0x61 -> Calling out starts(PTT pressed or start calling out is sent out by command 0x06)

**Call\_type:** 1byte, Calling type

0x00-> Call on analog channels

0x01-> Private call

0x02-> Group call

0x04-> All call

**Call\_ID:** 3 bytes, contact ID being called(Calling out starts) or radio ID which is calling in(Being called starts)

**Example:**

Calling out starts.

68 06 02 61 83 93 00 04 02 00 00 01 10

**Note:** Call\_type and Call\_ID are 0 on analog channels.

### 2.5.5 Ask for the calling in contact ID

**Description:** Under the DMR channel, query the radio ID of the incoming call after receiving a call.

**Format:** 68 10 01 01 CHKSUM 00 01 01 10

**Example:**

68 10 01 01 95 DD 00 01 01 10

**Note:** This command is not supported under analog channels.

### 2.5.6 Response of asking for the calling in radio ID

**Description:** Return the information of calling in contact.

**Format:** 68 10 00 01 CHKSUM 00 04 Call\_type Call\_ID 10

**Parameters:**

**Call\_type:** 1byte, Calling type

0x01-> Private call

0x02-> Group call

0x04-> All call

**Call\_ID:** 3 bytes, radio ID which is calling in

**Example:**

The walkie-talkie of radio ID 0x000001 is calling under a group call.

68 10 00 01 85 E9 00 04 02 00 00 01 10

**Note:** If this module never received a call before sending this command, it will return Call\_ID 0x000000.

## 2.6 SMS TX or RX command 0x07 & Query SMS content 0x11

### 2.6.1 SMS TX

**Description:** Send out message

**Format:** 68 07 01 01 CHKSUM LEN Msg\_type Call\_ID Msg 10

**Parameters:**

**LEN:** 2 bytes, length of frame=length of message+4

**Msg\_type:** 1 byte, message type

0x01-> Private message

0x09-> Group message

**Call\_ID:** 3 bytes, the contact of receiver of this message

**Msg:** Message content, ASCII format

**Example:**

Send private message '123' to contact ID 0x0000C8.

68 07 01 01 BF 24 00 0A 01 00 00 C8 31 00 32 00 33 00 10

**2.6.2 Response of SMS TX**

**Description:** The result of SMS TX.

**Format:** 68 07 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R** : 1 byte, Flag of response

0x71 -> Message sent success

0x7E -> Message sent fail

**Example:**

Message sent success .

68 07 00 71 87 87 00 00 10

**2.6.3 Upload SMS RX**

**Description:** Module output the contact ID and message via UART after received message.

**Format:** 68 07 02 70 CHKSUM LEN Call\_ID Msg 10

**Parameters:**

**LEN:** 2 byte, length of message +3

**Call\_ID:** 3 bytes, the radio ID of walkie-talkie which send this message

**Msg:** Message content

**Example:**

Received the message 'ABC' from radio ID 0x000002.

68 07 02 70 92 A9 00 09 00 00 02 41 00 42 00 43 00 10

**Note:**

It will not indicate that the received message are private or a group message.

Since the length of message is processed as even numbers, when length is odd, a 0x00 will be automatically added to the end of the message to make it even.

**2.6.4 Query SMS content**

**Description:** Read the content of last message.

**Format:** 68 11 01 01 CHKSUM 00 01 01 10

**Example:**

68 11 01 01 95 DC 00 01 01 10

**2.6.5 Response of query SMS content**

**Description:** Output the content of last message.

**Format:** 68 11 00 01 CHKSUM LEN Call\_ID Msg 10

**Parameters:**

**LEN:** 2 byte, length of message +3

**Call\_ID:** 3 bytes, the radio ID of walkie-talkie which send this message

**Msg:** Message content

**Example:**

Received the message '123' from radio ID 0x000001.

68 11 00 01 96 3E 00 09 00 00 01 31 00 32 00 33 00 10

**Note:** If this module never received a message before sending this command, it will return 68 11 00 01 87 ED 00 00 10.

## 2.7 Emergency alarm 0x09

### 2.7.1 Send alarm

**Description:** Send emergency alarm to receiver.

**Format:** 68 09 01 01 CHKSUM 00 01 01 Call\_ID 10

**Parameters:**

**Call\_ID:** 3bytes, The group contact ID of receiver.(Only support group call)

**Example:**

Send alarm to group contact ID 0x000001.

68 09 01 01 85 F0 00 04 01 00 00 01 10

### 2.7.2 Response of sending alarm

**Description:** Result of sending alarm.

**Format:** 68 09 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Send alarm successfully.

68 09 00 00 87 F6 00 00 10

**Note:** If there is no receiver, it will return 68 09 00 01 87 F5 00 00 10.

### 2.7.3 Upload alarm message

**Description:** Upload the alarm message via UART after received an alarm.

**Format:**

68 09 02 91 CHKSUM 00 03 Call\_ID 10

**Parameters:**

**Call\_ID:** 3 byte, the radio ID of walkie-talkie which sent this alarm

**Example:**

Received alarm from radio ID 0x000001.

68 09 02 91 94 52 00 03 00 00 01 10

## 2.8 Setting MIC gain command 0x0B

### 2.8.1 Instruction

**Description:** Setting the MIC gain. User needs to select the appropriate gain according to the MIC. If the gain is too small, the sound will be too small; if the gain is too large, the sound will be saturated and inaudible.

**Format:** 68 0B 01 01 CHKSUM 00 01 Gain 10

**Parameters:**

**Gain:** 1 byte, level of MIC gain(0x00~0x0F). The higher the value, the greater the gain.

**Example:**

Set Mic Gain with level 2.

68 0B 01 01 94 E2 00 01 02 10

### 2.8.2 Response

**Format:** 68 0B 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Setting MIC gain done.

68 0B 00 00 87 F4 00 00 10

## 2.9 Enter/exit duty working mode command 0x0C

### 2.9.1 Instruction

**Description:** Enter/exit duty working mode. Duty working mode can save power.

**Format:** 68 0C 01 01 CHKSUM 00 03 Switch 0A Mode 10

**Parameters:**

**Switch:** 1 byte, Enter/exit duty working mode

0x01 -> Enter duty working mode

0xFF -> Exit duty working mode

**Mode:** 1 byte, duty cycle. 1:4 is the most power-saving. The larger the duty cycle, the smaller the sleep, but the longer the response time.

0x01-> 1:1

0x02-> 1:2

0x04-> 1:4

**Example:**

Enter duty working mode, duty cycle is 1:4.

68 0C 01 01 91 D5 00 03 01 0A 04 10

### 2.9.2 Response

**Format:** 68 0C 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Enter duty working mode done.

68 0C 00 00 87 F3 00 00 10

**Note:**

In duty working mode, MCU is in sleep state and will not be able to respond to UART commands. It needs to send UART packets to wake up until it receives a wake-up reply. After it wakes up, if there is no command within 3 seconds, the module will re-enter duty working mode. The content of wake-up packet is at least 20 bytes of 0x55.

55 55 55 55 55 55 55 55 55 55 55 55 55 55 55 55 55 55 55 55

The module will reply '68 55 00 00 87 AA 00 00 10' after wake-up.

## 2.10 Set TX/RX Frequency 0x0D

### 2.10.1 Instruction

**Description:** Set the TX frequency and RX frequency. Only when the TX frequency of TX side is the same as the RX frequency of RX side, the TX side can communicate with RX side.

**Format:** 68 0D 01 01 CHKSUM 00 08 Rx\_Freq Tx\_Freq 10

**Parameters:**

**Rx\_Freq:** 4 bytes, RX frequency

**Tx\_Freq:** 4 bytes, TX frequency

**Example:**

RX frequency = 409.75M, TX frequency = 415.75M

68 0D 01 01 F2 96 00 08 F0 49 6C 18 70 D7 C7 18 10

**Note:** The frequency is LSB. For example, 415.75MHz=415750000Hz=0x18C7D770. So the data to be sent are 0x70,0xD7,0xC7,0x18.

### 2.10.2 Response

**Format:** 68 0D 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Set TX/RX Frequency done.

68 0D 00 00 87 F2 00 00 10

## 2.11 Enter/exit repeater mode 0x0E

### 2.11.1 Instruction

**Description:** Setting module to enter/exit repeater mode.

**Format:** 68 0E 01 01 CHKSUM 00 01 Mode 10

**Parameters:**

**Mode:** 1 byte

0x01 -> Enter repeater mode

0x02 -> Exit repeater mode

**Example:**

Enter repeater mode.

68 0E 01 01 95 DF 00 01 01 10

**Note:** Only when the TX frequency is not the same as RX frequency, it can enter repeater mode.

### 2.11.2 Response

**Format:** 68 0E 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Enter repeater mode done.

68 0E 00 00 87 F1 00 00 10

## 2.12 SQ setting command 0x12

### 2.12.1 Instruction

**Description:** Set the level of squelch. Improve the SQ level can reduce the noise in the output audio, but it will also reduce the communication distance. The higher the SQ level, the smaller the noise and the shorter distance.

**Format:** 68 12 01 01 CHKSUM 00 01 Level 10

**Parameters:**

**Level:** 1 byte, SQ level(1~9)

**Example:**

Set SQ level to 1.

68 12 01 01 95 DB 00 01 01 10

**Note:** This instruction can only be used on analog channel.

### 2.12.2 Response

**Format:** 68 12 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Setting SQ level done.

68 12 00 00 87 ED 00 00 10

## 2.13 Select CTCSS / CDCSS type command 0x13

### 2.13.1 Instruction

**Description:** Select the type of CXCSS. The TX side and RX side can only communicate when the CXCSS type and code is match. This instruction can only be used on analog channel.

**Format:** 68 13 01 01 CHKSUM 00 02 RX\_CXCSS\_TYPE TX\_CXCSS\_TYPE 10

**Parameters:**

**RX\_CTCSS\_TYPE:** 1 byte, type of RX CXCSS

0x01 -> No CTCSS and no CDCSS (CSQ)

0x02 -> CTCSS (TPL)

0x03 -> CDCSS (DPL)

0x04 -> CDCSS Invert (DPL Invert)

**TX\_CTCSS\_TYPE:** 1 byte, type of TX CXCSS

0x01 -> No CTCSS and no CDCSS (CSQ)

0x02 -> CTCSS (TPL)

0x03 -> CDCSS (DPL)

0x04 -> CDCSS Invert (DPL Invert)

**Example:**

RX\_CTCSS\_TYPE= CDCSS Invert, TX\_CTCSS\_TYPE= CTCSS

68 13 01 01 86 E8 00 02 04 02 10

**Note:** After sending this command, command 14 should be sent to select the code of CXCSS.

**2.13.2 Response**

**Format:** 68 13 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Select the type of CXCSS done.

68 13 00 00 87 EC 00 00 10

**2.14 Select CTCSS / CDCSS code 0x14****2.14.1 Instruction**

**Description:** Select the code of CXCSS. The TX side and RX side can only communicate when the CXCSS type and code is match. This instruction can be only used on analog channel, and the CXCSS type is not CSQ.

**Format:**

68 14 01 01 CHKSUM 00 02 RX\_CXCSS TX\_CXCSS 10

**Parameters:**

**RX\_CXCSS:** 1 byte, index of RX CTCSS/CDCSS code, CTCSS (TPL) 1~50, CDCSS(DPL/ DPL Invert) 0~83

**TX\_CXCSS:** 1 byte, index of TX CTCSS/CDCSS code, CTCSS (TPL) 1~50, CDCSS(DPL/ DPL Invert) 0~83

**Example:**

RX CDCSS=23I, TX CTCSS =67Hz

68 14 01 01 86 E8 00 02 00 01 10

**Note:** Detailed relation between the index and the CXCSS code are in Appendix 1.

Command 0x13 should be sent first before sending this command.

**2.14.2 Response**

**Format:** 68 14 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Select the code of CXCSS done.

68 14 00 00 87 EB 00 00 10



## 2.15 Set TX power 0x17

### 2.15.1 Instruction

**Format:** 68 17 01 01 CHKSUM 00 01 Power 10

**Parameters:**

**Power:** 1 byte, TX power

0x01 -> High power

0xFF -> Low power

**Example:**

TX power = Low power

68 17 01 01 97 D5 00 01 FF 10

### 2.15.2 Response

**Format:** 68 17 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Set TX power done.

68 17 00 00 87 E8 00 00 10

## 2.16 Set contact 0x18

### 2.16.1 Instruction

**Description:** Set the TX contact. After pressing PTT, module will start calling out according to the contact ID. The contact ID set by this command will lost after power off.

**Format:** 68 18 01 01 CHKSUM 00 04 Call\_type Call\_ID 10

**Parameters:**

**Call\_type:** 1byte, Calling type

0x01-> Private call

0x02-> Group call

0x04-> All call

**Call\_ID :**3 bytes, contact ID to be called.

**Example:**

contact ID = 0x000001, type = Group call.

68 18 01 01 85 E1 00 04 01 00 00 01 10

**Note:** If user has never sent this command, after pressing PTT, Module will start calling out based on the current channel's default contact ID. The default contact ID can be changed by PC software and saved when power off.

Also querying the contact ID with the command 0x22 will always return the default contact ID for the current channel, whether or not this command has been sent.

### 2.16.2 Response

**Format:** 68 18 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Set contact ID done.

68 18 00 00 87 E7 00 00 10

## 2.17 Encryption on/off command 0x19

### 2.17.1 Encryption on command

**Description:** Open encryption. Only available on DMR channels.

**Format:** 68 19 01 01 CHKSUM 00 01 SWITCH KEY 10

**Parameters:**

**SWITCH:** 1 byte

0x01-> Encryption on

**KEY:** 8 bytes, encryption key. The key of transmitter and receiver must be the same.

**Example:**

Encryption on, Key = 0x0102030405060708

68 19 01 01 00 00 00 09 01 01 02 03 04 05 06 07 08 10

### 2.17.2 Encryption off command

**Description:** Close encryption. only available at DMR channels.

**Format:** 68 19 01 01 CHKSUM 00 01 SWITCH 10

**Parameters:**

**SWITCH:** 1 byte

0xFF-> Encryption off

**Example:**

Encryption off

68 19 01 01 00 00 00 01 FF 10

### 2.17.3 Response

**Description:** Response of encryption on/off.

**Format:** 68 19 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Encryption done.

68 19 00 00 87 E6 00 00 10

**Note:** After encryption, the voice or SMS data cannot be parsed, but it will still upload the UART message (command 0x06/0x07).

**2.18 Check initialization status 0x1A****2. 18.1 Instruction**

**Description:** Check whether the module has completed initialization.

**Format:** 68 1A 01 01 CHKSUM 00 01 01 10

**Example:**

68 1A 01 01 95 D3 00 01 01 10

**2. 18.2 Response**

**Format:** 68 1A 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x09 -> Checksum error

**Example:**

Module has completed initialization.

68 1A 00 00 87 E5 00 00 10

**2.19 Check the contact ID 0x22****2. 19.1 Instruction**

**Description:** Ask for the contact ID of current channel.

**Format:** 68 22 01 01 CHKSUM 00 01 01 10

**Example:**

68 22 01 01 95 CB 00 01 01 10

**2. 19.2 Response**

**Format:** 68 22 00 S/R CHKSUM 00 0E Call\_name Call\_ID Call\_type 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Call\_name:** 10 bytes, Name of contact

**Call\_ID:** 3 bytes, contact ID

**Call\_type:** 1byte, type of contact

0x01-> Private call

0x02-> Group call

0x04-> All call

**Example:**

Name of contact = "Call1", contact ID = 0x000001, type of contact= Group call

68 22 00 00 A5 FF 00 0E 43 61 6C 6C 31 00 00 00 00 00 00 01 02 10

**Note:** The contact is the default contact ID for the current channel. After setting the contact with the 0x18 command, the return value of the command still does not change.

**2.20 Check the radio ID 0x24****2. 20.1 Instruction**

**Description:** Check the radio ID of this module. Radio ID is used in private call. If other walkie-talkies call to this module on private call, the contact ID should be the same as radio ID of this module.

**Format:** 68 24 01 01 CHKSUM 00 01 01 10

**Example:**

68 24 01 01 95 C9 00 01 01 10

## 2. 20.2 Response

**Format:** 68 24 00 S/R CHKSUM 00 03 Radio\_ID 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x09 -> Checksum error

**Radio\_ID:** 3 bytes, radio ID of this module

**Example:**

The radio ID of this module is 0x000001.

68 24 00 00 96 C8 00 03 00 00 01 10

## 2.21 Check the firmware version 0x25

### 2. 21.1 Instruction

**Format:** 68 25 01 01 CHKSUM 00 01 01 10

**Example:**

68 25 01 01 95 C8 00 01 01 10

### 2. 21.2 Response

**Format:** 68 25 00 S/R CHKSUM 00 LEN Version 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x09 -> Checksum error

**Len:** 1 byte, the length of version

**Version :** Firmware version.

**Example:**

Firmware version="DMR818S\_V1.0".

68 25 00 00 E8 4F 00 0C 44 4D 52 38 31 38 53 5F 56 31 2E 30 10

## 2.22 Check the encryption status 0x28

### 2.22.1 Instruction

**Description:** Check the encryption is on or off.

**Format:** 68 28 01 01 CHKSUM 00 01 01 10

**Example:**

68 28 01 01 95 C5 00 01 01 10

### 2.22.2 Response

**Format:** 68 28 00 S/R CHKSUM 00 01 SWITCH 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**SWITCH:** 1 byte, encryption status

0x00 -> Encryption off

0x01 -> Encryption on

**Example:**

Encryption on

68 28 00 00 96 C6 00 01 01 10

## 2.23 Add contact ID into RX group list 0x29

### 2.23.1 Instruction

**Description:** Add a new contact ID into a RX group list and associate the group list of current channel to this RX group list. In group call, the TX contact ID should in the related RX group list of receiver.

**Format:** 68 29 01 01 CHKSUM 00 04 INDEX Call\_ID 10

**Parameters:**

**INDEX:** 1 byte, index of RX group list, range 1~32

**Call\_ID:** 3 bytes, contact ID to be added.

**Example:**

Add contact ID 0x000001 into RX group list 2.

68 29 01 01 84 D0 00 04 02 00 00 01 10

### 2.23.2 Response

**Format:** 68 29 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Add contact ID into RX group list done.

68 29 00 00 87 D6 00 00 10

Note: Each DMR channel has a parameter named group list. The group list is an index which used to associate to one of the 32 RX group lists. After associated to the RX group list, this channel can listen to any contact ID in this RX group list. A RX group list can save a maximum of 32 members.

## 2.24 Clear a RX group list 0x30

### 2.24.1 Instruction

**Description:** Delete all the contact in a RX group list

**Format:** 68 30 01 01 CHKSUM 00 01 INDEX 10

**Parameters:**

**INDEX:** 1 byte, the index of RX group list which need to be cleared, range 1~32

**Example:**

Delete all the contact in RX group list 1.

68 30 01 01 95 BD 00 01 01 10

### 2.24.2 Response

**Format:** 68 30 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Clear a RX group list done.

68 30 00 00 87 CF 00 00 10

## 2.25 Set radio ID 0x1B

### 2.25.1 Instruction

**Description:** Set the Radio ID. Radio ID is used in private call. If other walkie-talkies call to this module on private call, the contact ID should be the same as radio ID of this module.

**Format:** 68 1B 01 01 CHKSUM 00 03 Radio\_ID 10

**Parameters:**

**Radio\_ID:** 3 bytes, Radio ID.

**Example:**

Radio ID = 0x000001.

68 1B 01 01 95 D0 00 03 00 00 01 10

### 2.25.2 Response

**Format:** 68 1B 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Set the radio ID done.

68 1B 00 00 87 E4 00 00 10

**Note:** The radio ID of two walkie-talkies can be the same when only use group call. If need to use private call, every walkie-talkie should have a unique radio ID.

## 2.26 Set the color code 0x31

### 2. 26.1 Instruction

**Description:** Set the color code. A color code is used to identify a system. Walkie-talkies can't communicate with different color code.

**Format:** 68 31 01 01 CHKSUM 00 01 ColorCode 10

**Parameters:**

**ColorCode:** 1 byte, Color code, range 0~15

**Example:**

Color code=1

68 31 01 01 95 BC 00 01 01 10

**2. 26.2 Response**

**Format:** 68 31 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Set the color code done.

68 31 00 00 87 CE 00 00 10

**2.27 Set bandwidth 0x32****2.27.1 Instruction**

**Description:** Set the bandwidth of current channel. This instruction can only be used in analog channels. Walkie-talkies with different bandwidth can communication with each other, and the communication effect will be poor.

**Format:** 68 32 01 01 CHKSUM 00 01 BW 10

**Parameters:**

**BW:** 1 byte, Bandwidth

0x00 ->12.5K

0x01 ->25K

**Example:**

Bandwidth = 12.5k

68 32 01 01 96 BB 00 01 00 10

**2. 27.2 Response**

**Format:** 68 32 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Set the bandwidth done.

68 32 00 00 87 CD 00 00 10

**2.28 Set time slot 0x33****2. 28.1 Instruction**

**Format:** 68 33 01 01 CHKSUM 00 01 TIMESLOT 10

**Parameters:**

**TIMESLOT:** 1 byte, Time slot

0x01 -> Slot 1

0x02 -> Slot 2

**Example:**

Time slot=slot 1

68 33 01 01 95 BA 00 01 01 10

## 2. 28.2 Response

**Format:** 68 33 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x02 -> Channel errors

0x09 -> Checksum error

**Example:**

Set time slot done.

68 33 00 00 87 CC 00 00 10

## 2.29 On/off tone 0x1C

### 2. 29.1 Instruction

**Description:** Turn off/on the boot tone, call out tone, etc. Tone is on by default.

**Format:** 68 1C 01 01 CHKSUM 00 01 TONE 10

**Parameters:**

**TONE:** 1 byte

0x00 -> Tone is on

0x01 -> Tone is off

**Example:**

Tone is on.

68 1C 01 01 95 D1 00 01 01 10

### 2. 29.2 Response

**Format:** 68 1C 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x09 -> Checksum error

**Example:**

On/off tone done.

68 1C 00 00 87 E3 00 00 10

## 2.30 Check the parameters of current channel 0x1D

### 2. 30.1 Instruction

**Format:** 68 1D 01 01 CHKSUM 00 01 01 10



**Example:**

68 1D 01 01 00 00 00 01 01 10

**2. 30.2 Response**

**Format 1(DMR channels):** 68 1D 00 S/R CHKSUM 00 LEN CH\_TYPE TX\_FREQ  
RX\_FREQ POWER CC SLOT ENCY CALL\_TYPE CALL\_NUM RXLIST RX\_NUM 10

**Parameters:**

**CH\_TYPE:** 1 byte, Type of channel

0x01-> Analog channel

0x02 -> DMR channel

**TX\_FREQ:** 4 bytes, TX frequency

**RX\_FREQ:** 4 bytes, RX frequency

**POWER:**1 byte, TX power

0x00-> Low power

0x01-> High power

**CC:** 1 byte, Color code, 0~15

**SLOT:** 1 byte, Time slot

0x01-> Slot 1

0x02-> Slot 2

**ENCY:** 1 byte, Encryption

0x00-> Encryption off

0x01-> Encryption on

**CALL\_TYPE:** 1 byte, type of contact

0x01-> Private call

0x02-> Group call

0x04-> All call

**CALL\_NUM:** 3 bytes, contact ID

**RXLIST:** 1 byte, index of group list

**RX\_NUM:** 3\*n bytes, the contact ID in current group list.

**Example:**

DMR channel, TX frequency = RX frequency = 418.125MHz, TX power = High power, Color code = 1, Time slot= Slot 1, Encryption = OFF, Contact= Group call 0x000001, Group list = RX group list1(Include contact ID 0x000001)

68 1D 00 00 37 50 00 15 02 C8 14 EC 18 C8 14 EC 18 01 01 01 00 02 00 00 01 01 00 00 01 10

**Format 2(Analog channels):** 68 1D 00 S/R CHKSUM 00 LEN CH\_TYPE TX\_FREQ  
RX\_FREQ POWER BW TX\_CXCSS\_TYPE TX\_CXCSS RX\_CXCSS\_TYPE RX\_CXCSS 10

**Parameters:**

**CH\_TYPE:** 1 byte, Type of channel

0x01-> Analog channel

0x02 -> DMR channel

**TX\_FREQ:** 4 bytes, TX frequency

**RX\_FREQ:** 4 bytes, RX frequency

**POWER:**1 byte, TX power

0x00-> Low power

0x01-> High power

**BW:** 1 byte, Bandwidth

0x01 ->12.5K

0x02 ->25K

**TX\_CTCSS\_TYPE:** 1 byte, type of TX CXCSS

0x00 -> No CTCSS and no CDCSS (CSQ)

0x01 -> CTCSS (TPL)

0x02 -> CDCSS (DPL)

0x03 -> CDCSS Invert (DPL Invert)

**TX\_CXCSS:** 1 byte, TX CTCSS/CDCSS code, CTCSS (TPL) 1~50, CDCSS(DPL/ DPL Invert) 0~83

**RX\_CTCSS\_TYPE:** 1 byte, type of RX CXCSS

0x00 -> No CTCSS and no CDCSS (CSQ)

0x01 -> CTCSS (TPL)

0x02 -> CDCSS (DPL)

0x03 -> CDCSS Invert (DPL Invert)

**RX\_CXCSS:** 1 byte, RX CTCSS/CDCSS code, CTCSS (TPL) 1~50, CDCSS(DPL/ DPL Invert) 0~83

**Example:**

Analog channel, TX frequency = RX frequency = 418.125MHz, TX power = High power,  
Bandwidth=12.5k, TX CXCSS=RX CXCSS=CSQ

68 1D 00 00 3A 5A 00 0F 01 C8 14 EC 18 C8 14 EC 18 01 01 00 00 00 00 10

**Note:** Frequency is by LSB. C8 14 EC 18->0x18EC14C8->418125000->418MHz

## 2.31 Reset to default parameters 0xF0

### 2. 31.1 Instruction

**Description:** Restore module parameters to default values

**Format:** 68 F0 01 01 CHKSUM 00 01 01 10

**Example:**

68 F0 01 01 94 FD 00 01 01 10

### 2. 31.2 Response

**Format:** 68 F0 00 S/R CHKSUM 00 00 10

**Parameters:**

**S/R:** 1 byte, Flag of responding

0x00 -> Done

0x01 -> Busy or fail

0x09 -> Checksum error

**Example:**

68 F0 00 00 87 0F 00 00 10

**Note:** After response, module will reset to make the parameters take effect.

## 2.32 Software reset 0xF2

### 2. 32.1 Instruction

**Format:** 68 F2 01 01 CHKSUM 00 01 01 10

**Example:**

68 F2 01 01 94 FD 00 01 01 10

### 2. 32.2 Response

**Description:** After response, module will reset automatically.

**Format:** 68 F2 00 S/R CHKSUM 00 00 10

**Parameters:****S/R:** 1 byte, Flag of responding

0x00 -&gt; Done

0x01 -&gt; Busy or fail

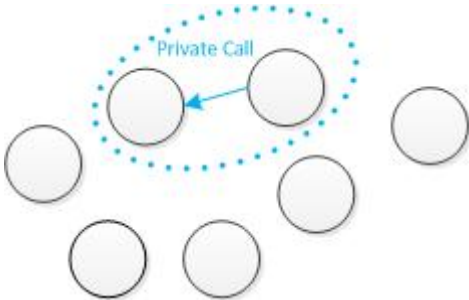
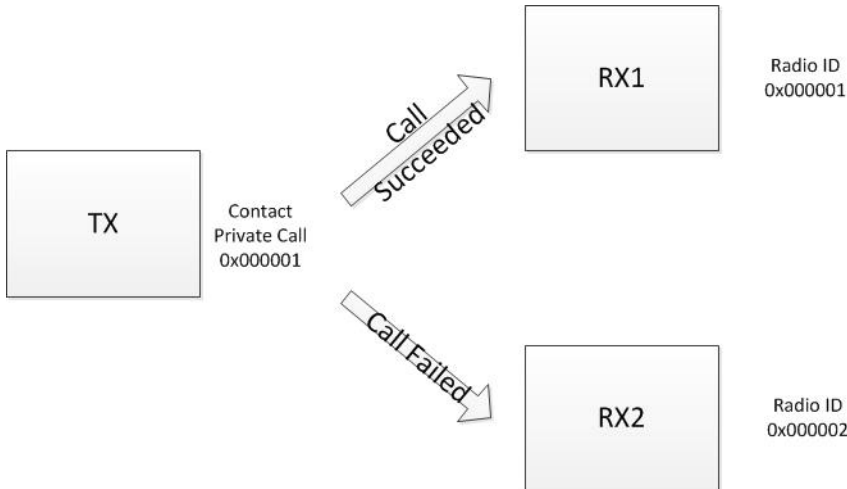
0x09 -&gt; Checksum error

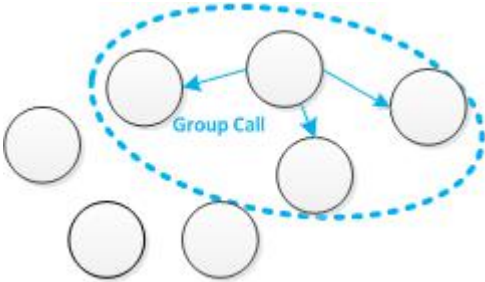
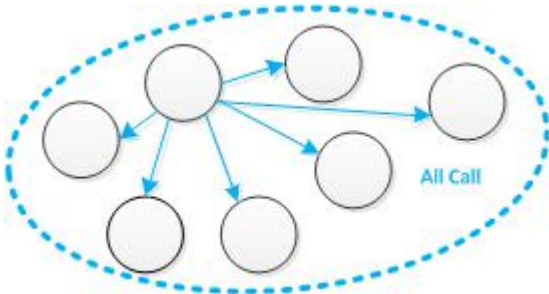
**Example:**

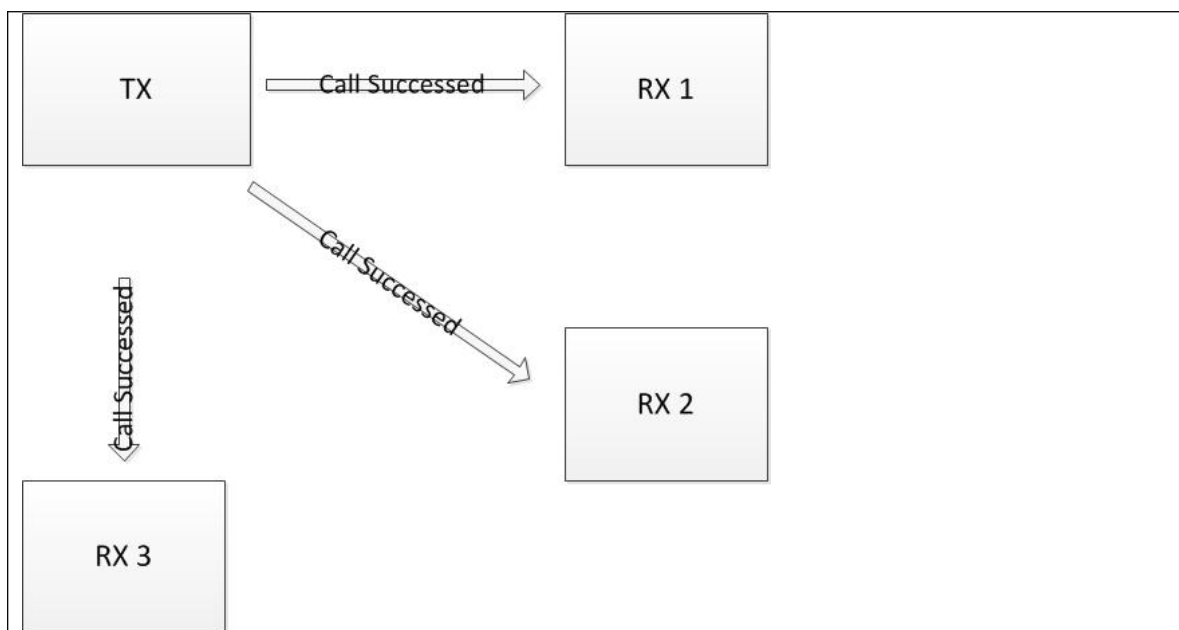
68 F2 00 00 87 0D 00 00 10

**Note:** After reset, the configuration that does not support power-off saving will be lost. See Appendix 2 for whether parameters can be saved after power-off.

## II Call Type Details

Call Type	Requirements	Features	Related commands	
			TX	RX
Private call	The type of contact is private call. The ID of contact ID is the same as the radio ID of receiver.	One-to-one communication, one walkie-talkie calls to designated walkie-talkie. 	Set contact ID 0x18	Set radio ID 0x1B
				
Group call	The type of contact is group call. The contact ID can be	One-to-many communication, one walkie-talkie calls to several specific walkie-talkies at the same time	Set contact ID	Add contact ID into

	found in the group list for the current channel of receiver.		0x18	RX group list 0x29
<div style="display: flex; flex-direction: column; align-items: center;"> <div style="display: flex; justify-content: space-around; width: 100%;"> <div style="border: 1px solid black; padding: 10px; text-align: center;">TX</div> <div style="text-align: center;">             Call Succeeded →           </div> <div style="border: 1px solid black; padding: 10px; text-align: center;">RX1</div> </div> <div style="margin-top: 20px;">             Group list  <span style="border: 1px solid green; padding: 2px;">0x000001</span>              0x000002              0x000003           </div> <div style="display: flex; justify-content: space-around; width: 100%; margin-top: 20px;"> <div style="text-align: center;">             Contact Group Call              0x000001              ↓              Call Succeeded           </div> <div style="text-align: center;">             Call Fail ↘           </div> </div> <div style="display: flex; justify-content: space-around; width: 100%; margin-top: 20px;"> <div style="border: 1px solid black; padding: 10px; text-align: center;">RX3</div> <div style="text-align: center;">             Group list  <span style="border: 1px solid green; padding: 2px;">0x000001</span> </div> </div> <div style="border: 1px solid black; padding: 10px; text-align: center; margin-top: 20px; width: 200px;">RX2</div> <div style="margin-top: 20px;">             Group list              0x000002              0x000003           </div> </div>				
All call	The type of contact is all call, and the ID is in the range of 0xFFFFCE0~0xFFFFFFF, No requirement for receiver	Broadcast mode, one walkie-talkie calls all walkie-talkies 	Set contact ID 0x18	None



### 1. Communication Parameter Requirements

Before calling, it needs to confirm that the following parameters of the communication module are the same, otherwise the calling will be unsuccessful. The parameters of each channel have been configured to communicate by default. It only needs to ensure the modules share the same channel.

Analog Channel	Frequency, CXCSS of modules are the same
DMR Channel	Frequency, contact ID, color code and encryption of modules are the same.

The communication-related parameters in the channel default configuration are as follows.

Channel	Type	TX Frequency(MHz)	RX Frequency(MHz)
1	DMR	418.125	418.125
2	DMR	419.125	419.125
3	DMR	420.125	420.125
4	DMR	421.125	421.125
5	DMR	422.125	422.125
6	DMR	423.125	423.125
7	DMR	424.125	424.125
8	DMR	425.125	425.125
9	Analog	418.125	418.125
10	Analog	419.125	419.125
11	Analog	420.125	420.125
12	Analog	421.125	421.125
13	Analog	422.125	422.125
14	Analog	423.125	423.125
15	Analog	424.125	424.125
16	Analog	425.125	425.125

The following parameters are the same for all analog channels.

Analog channel
----------------

CXCSS	No CTCSS and no CDCSS (CSQ)
-------	-----------------------------

The following parameters are the same for all DMR channels.

DMR channel	
Contact	Group call 0x000001
Group List	Rx group list1(including ID 0x000001)
Color Code	1
Encryption	OFF

## 2. Relevant Command Description

Analog Channel	
Set TX/RX Frequency 0x0D	Only when the TX frequency of TX side is the same as the RX frequency of RX side, the TX side can communicate with RX side.
Select CTCSS / CDCSS type command 0x13	CXCSS: On an analog channel, a specific frequency below the audio frequency that is loaded into the wireless signal. When the specific frequency match the code of receiver, the receiver can output the voice.  Users can use different CXCSS to avoid interference from other radios on the same frequency. When the RX CXCSS of receiver is No CTCSS and no CDCSS (CSQ), receiver can parse all the signal of different CXCSS.  The types of CXCSS are CTCSS (TPL), CDCSS (DPL), CDCSS Invert (DPL Invert).  CTCSS (TPL) is a series of frequency below the audio frequency, range from 67Hz to 254.1Hz.  CDCSS (DPL), CDCSS Invert (DPL Invert) are a series of 134.4 bit/s codes, see Appendix 1 for codes.
Select CTCSS / CDCSS code 0x14	
DMR Channel	
Set TX/RX Frequency 0x0D	Only when the TX frequency of TX side is the same as the RX frequency of RX side, the TX side can communicate with RX side.
Set contact ID 0x18	Contact ID: After pressing PTT, module will start calling out according to the contact ID. Contact consists of type and ID. The types of contact include private call, group call and all call. For the introduction of each kind of call, see the introduction of call type detail.
Add contact ID into RX group list 0x29	Used in group call.  Add a new RX group call contact ID. It is done by added the new contact ID into the associated RX group list of current channel.  Group list: A parameter of DMR channels. It is an index pointed to an RX group list.  RX group list: Lists which include some contact IDs of group call.  The group list is used to associate to one of the 32 RX group lists. After associated to the RX group list, this channel can listen to any contact ID in this RX group list.
Set radio ID 0x1B	Used in private call.

	The contact ID of transmitter should be the same as radio ID of receiver.
Set the color code 0x31	Walkie-talkies which need to communicate with each other must have the same color code
Encryption on/off command 0x19	The encryption selection and key of transmitter and receiver must be the same.

### III Function Description

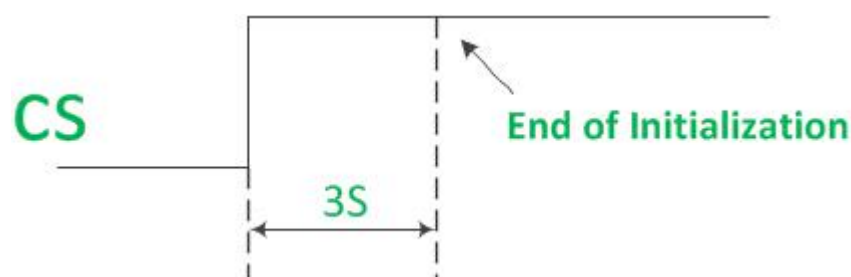
#### 1. Power on and Sleep

##### 1.1 Power on

(1) Power supply 3V~5V

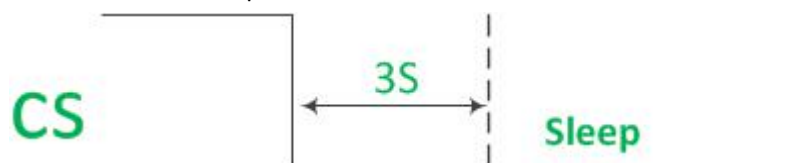
(2) CS pin to high level

After power up, the SPK\_EN pin and T/R pin will output a high level for a period of time and play the boot tone.



##### 1.2 Sleep

Pull the CS pin low, the module will enter the sleep state after 3 seconds. In the sleep state, the module can't call or respond to serial commands.



Note:

The PTT should keep floating or high level during initialization.

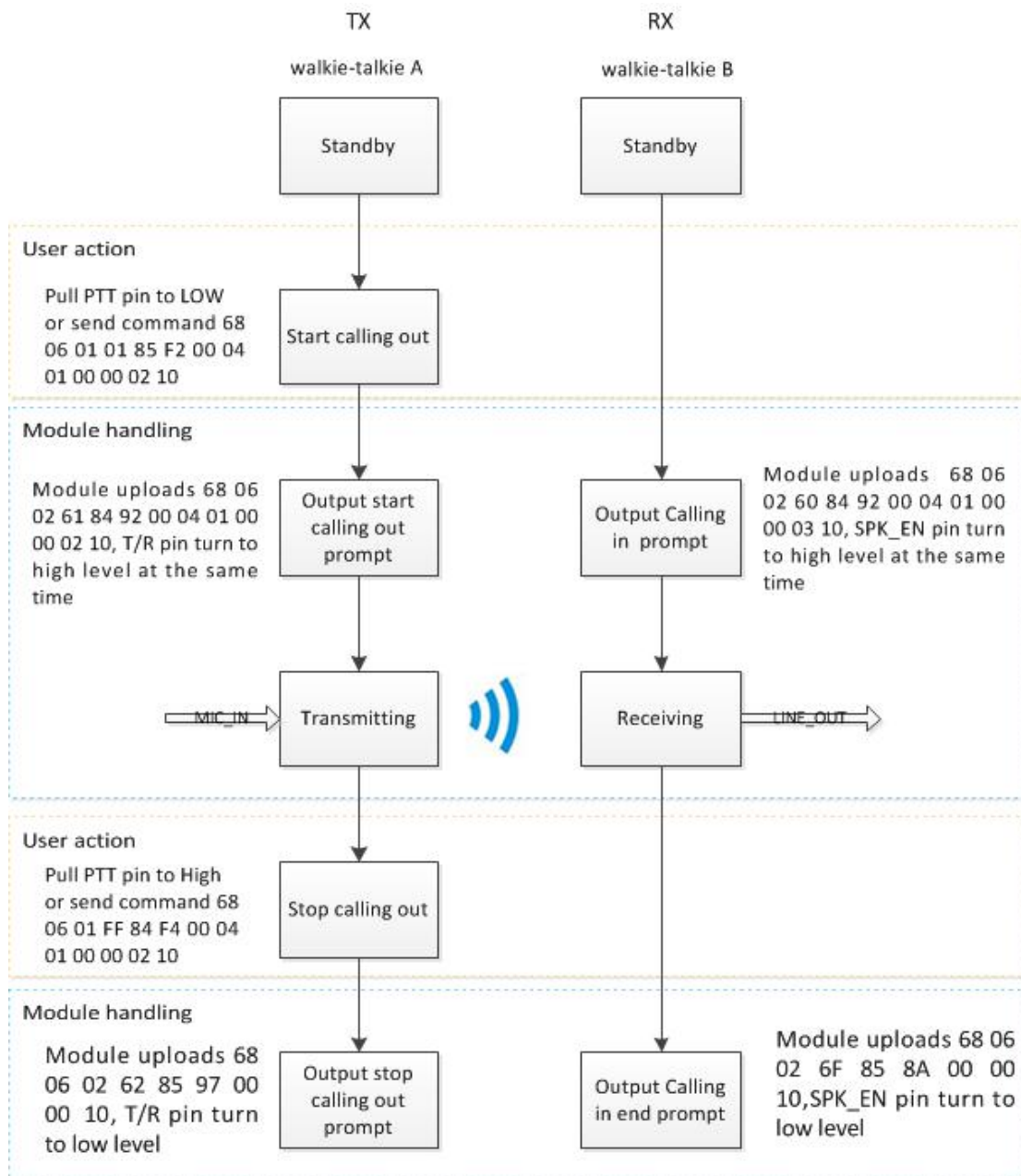
The power supply voltage should preferably be above 4V. When the voltage is too low, it may transmit fail.

#### 2. Calling

When the parameters are match, the transmitter can communicate with the receiver by pressing the PTT key or sending command 0x06.

##### 2.1 Private call

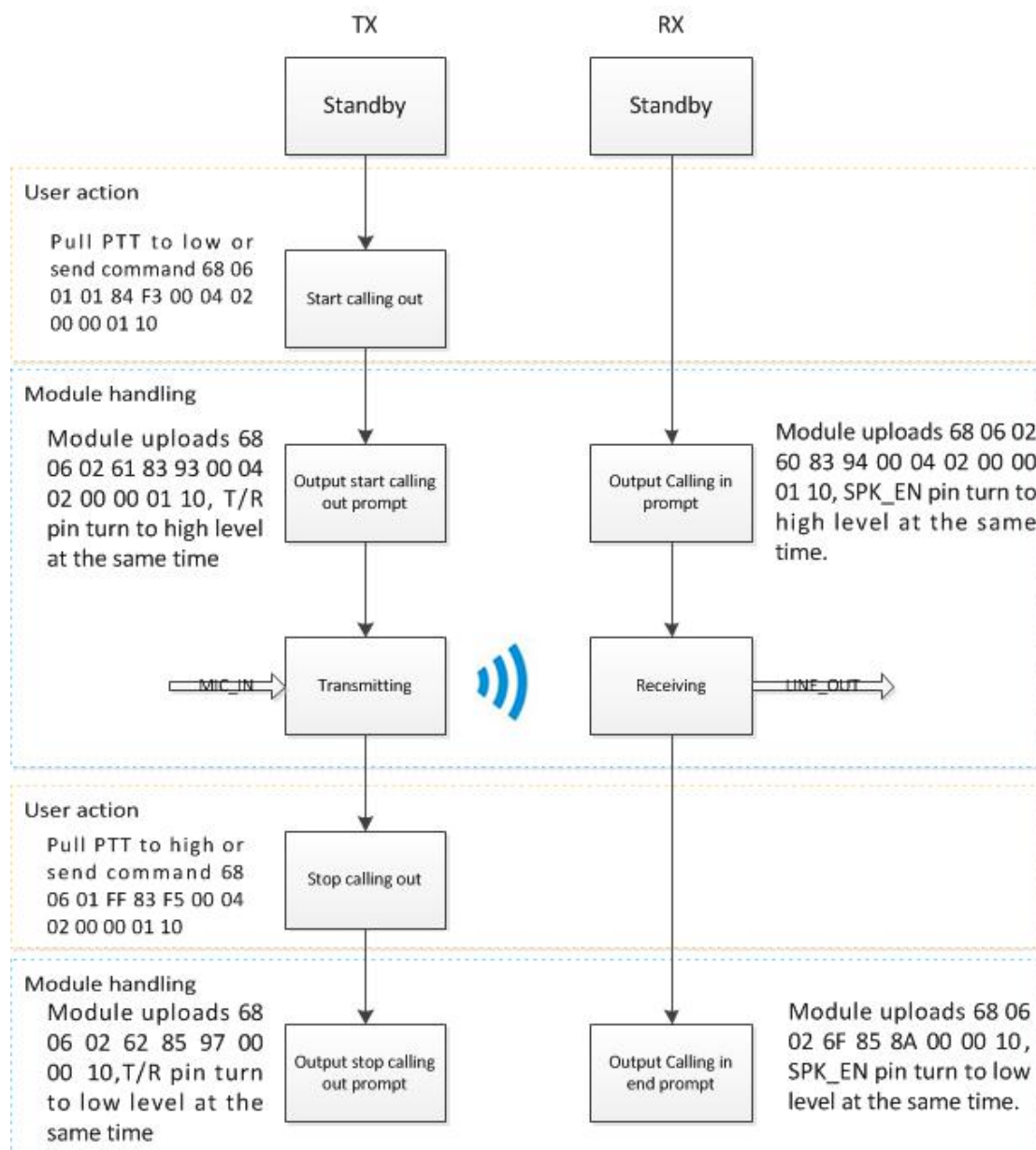
The process for the walkie-talkie A(Radio ID 0x000003) to call walkie-talkie B(Radio ID 0x000002) is as follow.



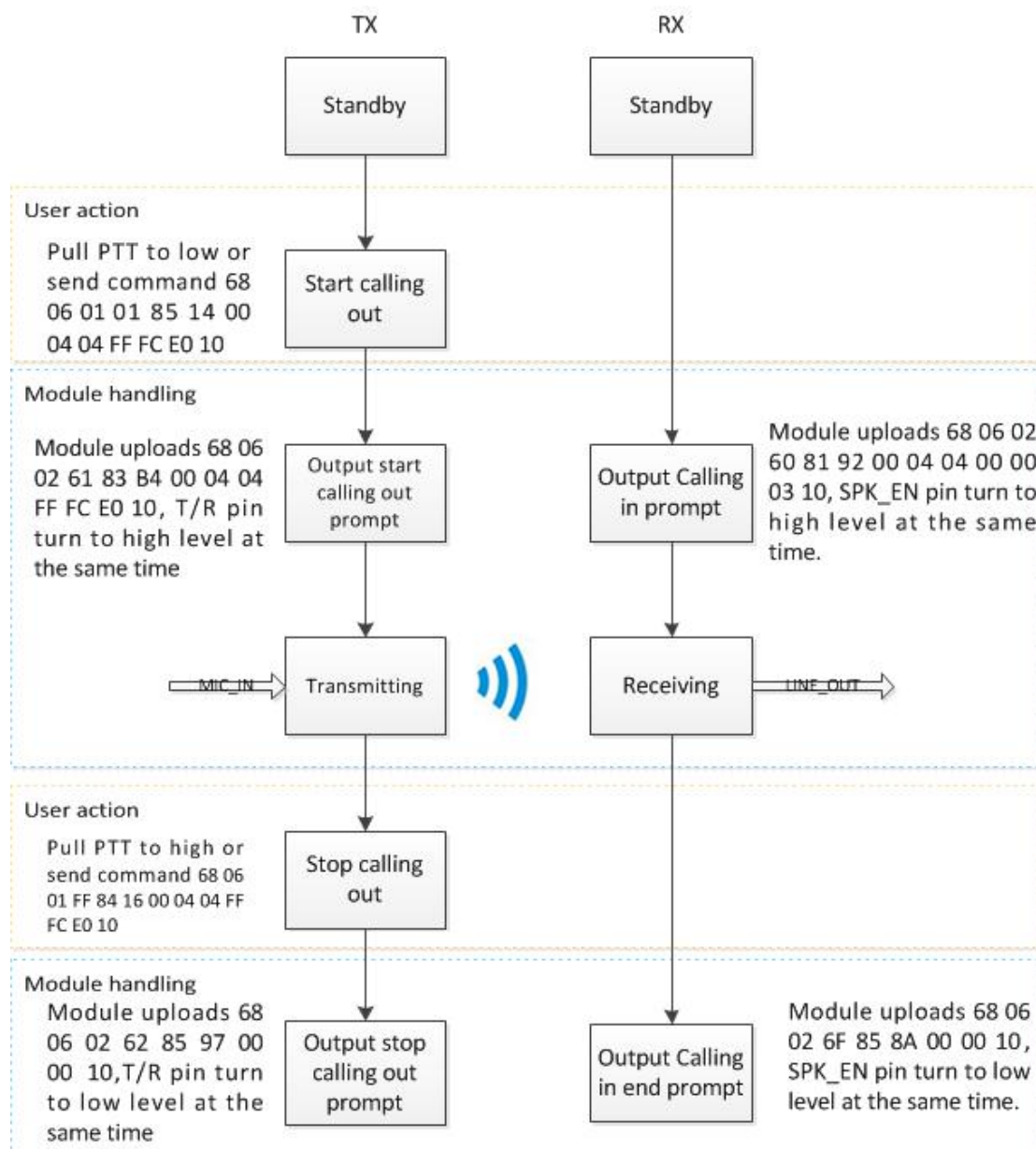
## 2.2 Group Call

The process for the group calling with contact ID 0x000001 is as follow.





## 2.3 All Call

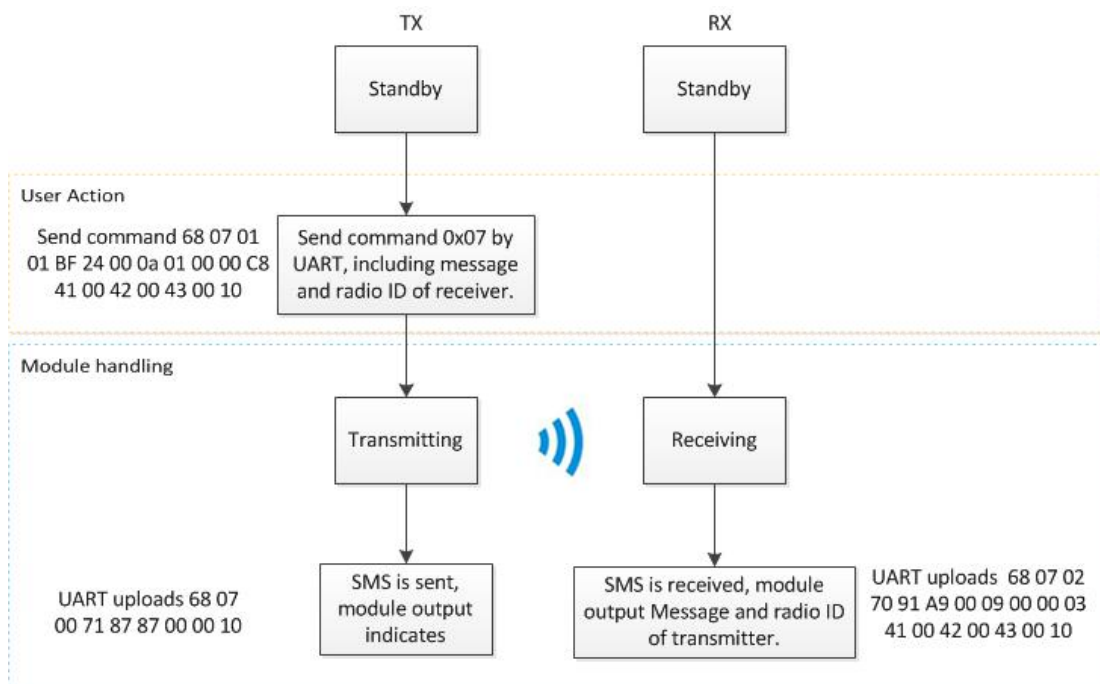


## 3. SMS

Transmitter can send SMS to receiver by command 0x07 under the same frequency, color code and encryption.

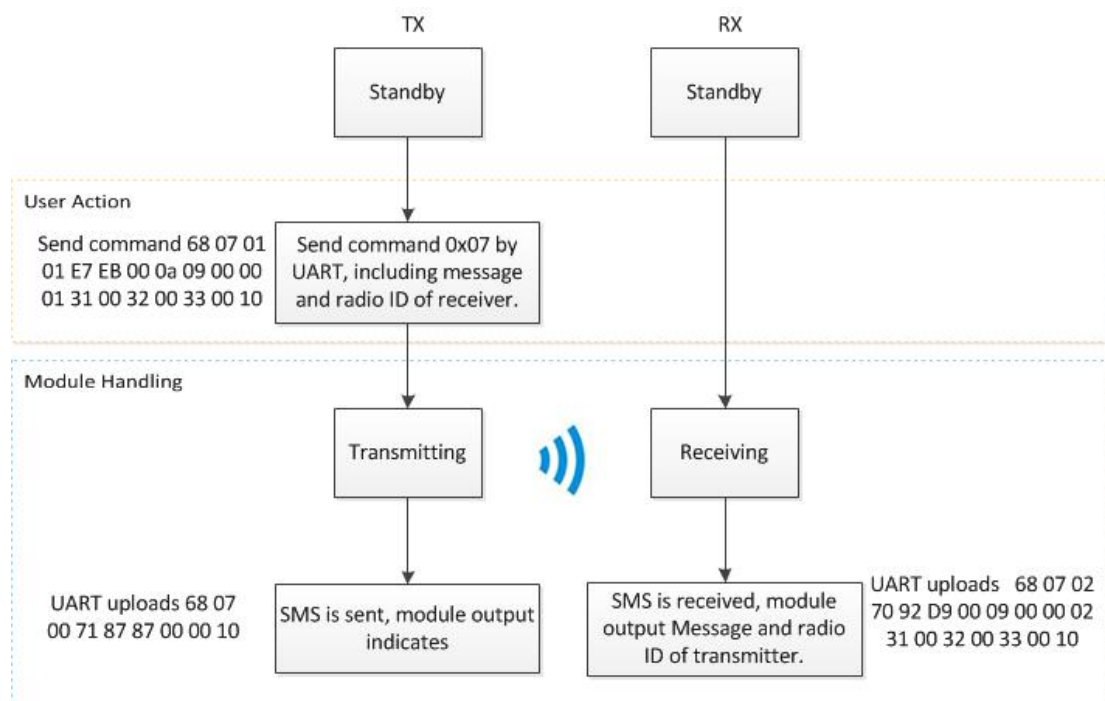
### 3.1 Private SMS

Walkie-talkie A(Radio ID 0x000003) send private message 'ABC' to walkie-talkie B(本机号码 0x00000C8) is as follow.



### 3.2 Group SMS

The process for sending group message with contact ID 0x000001 is as follow.

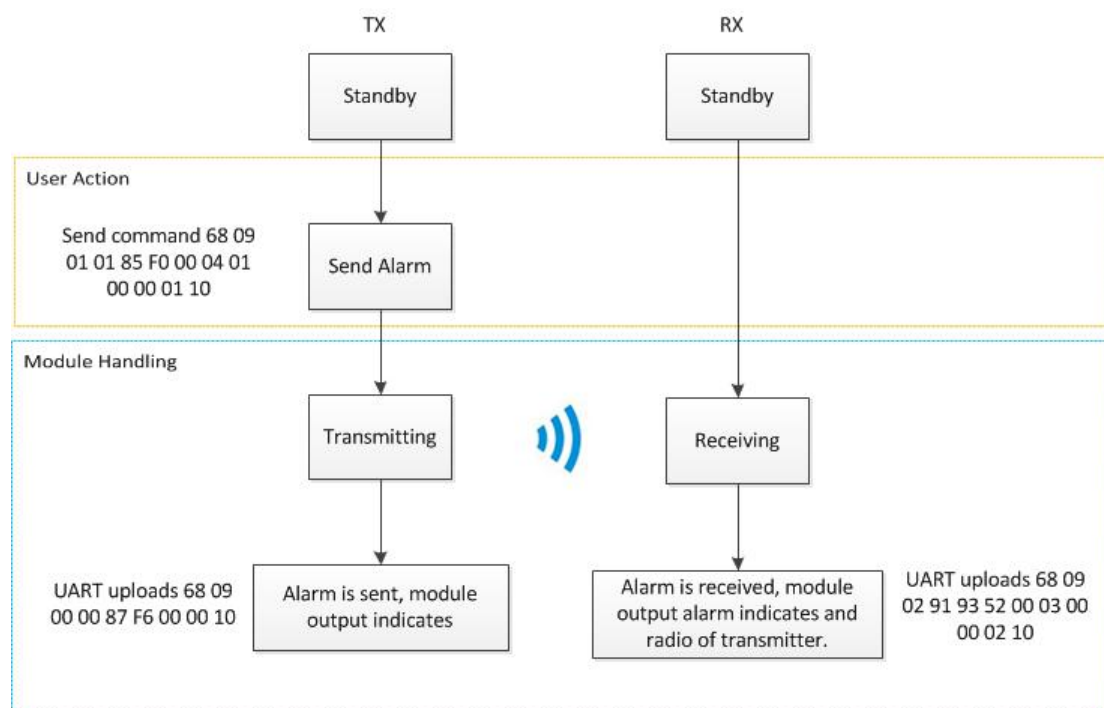


The contact in command 0x07 should correspond to the contact of receiver, otherwise the receiver can't receive this message.

Group SMS	The contact ID of message should be in the related RX group list of receiver.
Private SMS	The contact ID of message should be the same with radio ID of receiver.

#### 4. Emergency Alarm

Emergency alarm is a non-voice signaling of DMR, which is mostly used to send alarms. The module simplifies the emergency alarm process. the receiver only outputs the radio ID of transmitter. After receiving the alarm information, the user can perform follow-up operations as needed.



The contact type of emergency is group call. So that only when the contact ID in command 09 is in the released RX group list of receiver, the receiver can receive alarm. If there is no module that can respond to the alarm, transmitter will return the alarm failure.

#### 5. Encryption

DMR channels support encryption in calling and SMS. For calls and SMS under the encryption function, ensure that the keys at both ends are the same.

Encryption of walkie-talkie A	Key of walkie-talkie A	Encryption of walkie-talkie B	Key of walkie-talkie B	Communication between A and B
Open	0x0102030405060708	Close		Fail
Open	0x0102030405060708	Open	0x0202030405060708	Fail
Open	0x0102030405060708	Open	0x0202030405060708	Success

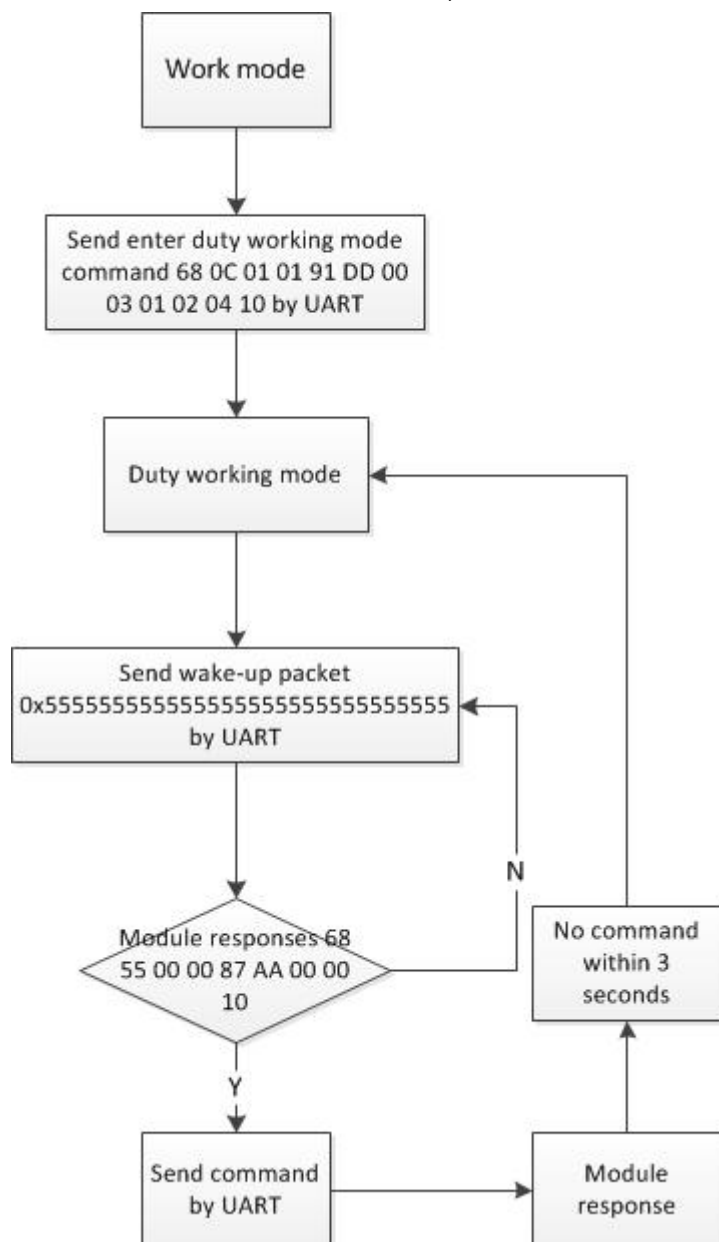
Note: This function is used to encrypt voice data and message data. After encryption, module will also output the prompt if received calling or message, but it can't decode the voice or message.

#### 6. Duty Working Mode

The duty working mode switches between sleep and work modes automatically to reduce the current. Module can call out or receive signal. However, the response time will be longer than work mode due to the need to wake up the module.

In duty working mode, MCU is in sleep state and will not be able to respond to UART commands. It needs to send UART packets to wake up until it receives a wake-up reply. After it wakes up, if

there is no command within 3 seconds, the module will re-enter duty working mode.



The duty working mode is different from sleep mode. The sleep mode can't call or respond to any UART command, and the duty working mode can call in and out.

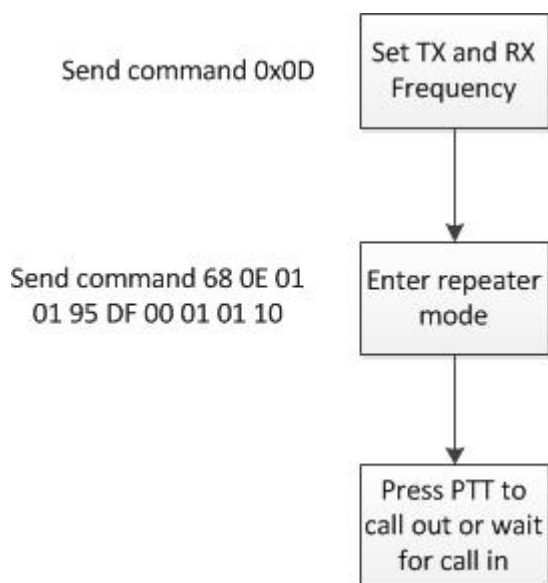
## 7. Repeater

The module can communicate with repeater as a node, but the module cannot be used as a repeater.

To communicate with repeater, module needs to send command 0x0E to enter repeater mode.

And it also needs to set the frequency and time slot according to the repeater.

After entered repeater mode, modules can only communicate through the repeater. If there is no repeater, it will response fail when pressing PTT.



Note: Module will not enter repeater mode when the TX frequency and RX frequency are the same.

## Appendix 1 CXCSS Code

Index of CTCSS	CTCSS		Index of Inver CDCSS	Inver CDCSS Code		Index of CDCSS	CDCSS Code
1	67		0	023I		0	023N
2	69.3		1	025I		1	025N
3	71.9		2	026I		2	026N
4	74.4		3	031I		3	031N
5	77		4	032I		4	032N
6	79.7		5	043I		5	043N
7	82.5		6	047I		6	047N
8	85.4		7	051I		7	051N
9	88.5		8	054I		8	054N
10	91.5		9	065I		9	065N
11	94.8		10	071I		10	071N
12	97.4		11	072I		11	072N
13	100		12	073I		12	073N
14	103.5		13	074I		13	074N
15	107.2		14	114I		14	114N

16	110.9		15	115I		15	115N
17	114.8		16	116I		16	116N
18	118.8		17	125I		17	125N
19	123		18	131I		18	131N
20	127.3		19	132I		19	132N
21	131.8		20	134I		20	134N
22	136.5		21	143I		21	143N
23	141.3		22	152I		22	152N
24	146.2		23	155I		23	155N
25	151.4		24	156I		24	156N
26	156.7		25	162I		25	162N
27	159.8		26	165I		26	165N
28	162.2		27	172I		27	172N
29	165.5		28	174I		28	174N
30	167.9		29	205I		29	205N
31	171.3		30	223I		30	223N
32	173.8		31	226I		31	226N
33	177.3		32	243I		32	243N
34	179.9		33	244I		33	244N
35	183.5		34	245I		34	245N
36	186.2		35	251I		35	251N
37	189.9		36	261I		36	261N
38	192.8		37	263I		37	263N
39	196.6		38	265I		38	265N
40	199.5		39	271I		39	271N
41	203.5		40	306I		40	306N
42	206.5		41	311I		41	311N
43	210.7		42	315I		42	315N
44	218.1		43	331I		43	331N

45	225.7		44	343I		44	343N
46	229.1		45	346I		45	346N
47	233.6		46	351I		46	351N
48	241.8		47	364I		47	364N
49	250.3		48	365I		48	365N
50	254.1		49	371I		49	371N
			50	411I		50	411N
			51	412I		51	412N
			52	413I		52	413N
			53	423I		53	423N
			54	431I		54	431N
			55	432I		55	432N
			56	445I		56	445N
			57	464I		57	464N
			58	465I		58	465N
			59	466I		59	466N
			60	503I		60	503N
			61	506I		61	506N
			62	516I		62	516N
			63	532I		63	532N
			64	546I		64	546N
			65	565I		65	565N
			66	606I		66	606N
			67	612I		67	612N
			68	624I		68	624N
			69	627I		69	627N
			70	631I		70	631N
			71	632I		71	632N
			72	654I		72	654N



			73	<b>662I</b>		73	<b>662N</b>
			74	<b>664I</b>		74	<b>664N</b>
			75	<b>703I</b>		75	<b>703N</b>
			76	<b>712I</b>		76	<b>712N</b>
			77	<b>723I</b>		77	<b>723N</b>
			78	<b>731I</b>		78	<b>731N</b>
			79	<b>732I</b>		79	<b>732N</b>
			80	<b>734I</b>		80	<b>734N</b>
			81	<b>743I</b>		81	<b>743N</b>
			82	<b>754I</b>		82	<b>754N</b>

## Appendix 2: Command Properties

Command	Scope	Saved After Power Down	Para of Analog	Para of DMR	RESP during Calling Out	RESP during Calling In
Channel changing command 0x01		✓	✓	✓	✓	✓
Volume setting command 0x02	<b>All</b>	✓	✓	✓		✓
Check module status 0x04	Current Channel		✓	✓	✓	✓
RSSI 0x05	Current Channel		✓	✓	✓	✓
Call out/Call In 0x06	Current Channel		✓	✓	✓	✓
SMS TX or RX command 0x07	Current Channel			✓	✓	✓
Emergency alarm 0x09	Current Channel			✓		
Setting MIC gain command 0x0B	<b>All</b>	✓	✓	✓		
Enter/exit duty working mode command 0x0C	<b>All</b>	✓	✓	✓		
Set TX/RX Frequency 0x0D	Current Channel	✓	✓	✓		
Enter/exit repeater mode 0x0E	Current Channel	✓		✓		

Query call in contact 0x10	Current Channel			✓		✓
Query SMS content 0x11	Current Channel			✓		
SQ setting command 0x12	Current Channel	✓	✓			
Select CTCSS / CDCSS type command 0x13	Current Channel	✓	✓			
Select CTCSS / CDCSS code 0x14	Current Channel	✓	✓			
Set TX power 0x17	Current Channel	✓	✓	✓		
Set contact 0x18	Current Channel			✓		
Encryption on/off command 0x19	Current Channel	✓		✓		
Check initialization status 0x1A			✓	✓		✓
Check the contact ID 0x22	Current Channel			✓		
Check the radio ID 0x24	All			✓		
Check the firmware version 0x25	All		✓	✓		
Check the encryption status 0x28	Current Channel			✓		
Add contact ID into RX group list 0x29	Current Channel	✓		✓		
Clear a RX group list 0x30	Current Channel	✓		✓		
Set radio ID 0x1B	All	✓		✓		
Set the color code 0x31	Current Channel	✓		✓		
Set bandwidth 0x32	Current Channel	✓	✓			
Set time slot 0x33	Current Channel	✓		✓		
On/off tone 0x1C	All	✓	✓	✓		
Check the parameters of current channel 0x1D	Current Channel		✓	✓		
Reset to default parameters 0xF0	All	✓	✓	✓		

Software reset 0xF2	All		✓	✓		
---------------------	-----	--	---	---	--	--

### Appendix 3: Default Parameters

Channel	Type	TX Frequency(MHz)	RX Frequency(MHz)	TX Power
1	DMR	418.125	418.125	High
2	DMR	419.125	419.125	High
3	DMR	420.125	420.125	High
4	DMR	421.125	421.125	High
5	DMR	422.125	422.125	High
6	DMR	423.125	423.125	High
7	DMR	424.125	424.125	High
8	DMR	425.125	425.125	High
9	Analog	418.125	418.125	High
10	Analog	419.125	419.125	High
11	Analog	420.125	420.125	High
12	Analog	421.125	421.125	High
13	Analog	422.125	422.125	High
14	Analog	423.125	423.125	High
15	Analog	424.125	424.125	High
16	Analog	425.125	425.125	High

The following parameters are the same for all analog channels.

Analog Channel	
Bandwidth	12.5KHz
CXCSS	No CTCSS and no CDCSS (CSQ)

The following parameters are the same for all DMR channels.

DMR Channel	
Contact	Group Call 0x000001
Group List	Rx group list1(including ID 0x000001)
Color Code	1
Encryption	OFF
Time Slot	1