

# Table of Contents

<b>1 Serial Port Communication Protocol.....</b>	<b>1</b>
1.01 Set Digital Group Command (0x22).....	3
1.02 Set Analog Group Command (0x23).....	5
1.03 Set Transmission Command (0x26).....	6
1.04 Check if the Analog Circuit Initialization is Complete (0x27).....	7
1.05 Set Enhanced Function (0x28).....	8
1.06 Set Encryption Function (0x29).....	10
1.07 Set MIC Gain (0x2A).....	11
1.08 Retrieve Digital Voice Reception Information (0x2B).....	11
1.09 Send SMS (0x2C).....	13
1.10 Retrieve SMS (0x2D).....	14
1.11 Set Volume (0x2E).....	15
1.12 Set Monitor (0x2F).....	16
1.13 Set Squelch (0x30).....	16
1.14 Set Power Saving (0x31).....	17
1.15 Retrieve Signal Strength (0x32).....	18
1.16 Set Repeater Decoupling Function (0x33).....	19
1.17 Retrieve Version Information (0x34).....	20
1.18 Test Cutting Function in Progress (0x35).....	21
1.19 Module Power-On Status Information (0x36).....	22
1.20 Strategy (0x37).....	22
1.21 Set Digital-Analog Synchronization Check Command (0x38).....	23
1.22 Set Short Message Transmission Concatenation Format (0x3A).....	25
1.23 Set TOT (Time-Out Timer) (0x3B).....	26
1.24 Set spk_en Mode Type (0x3C).....	27
<b>2 DMR Module Basic Communication Usage Explanation.....</b>	<b>28</b>
2.01 Digital Intercom.....	28
2.02 Analog Intercom.....	28
2.03 Digital Intercom Contacts.....	29
2.04 Digital Intercom Parameter Settings.....	29
2.05 Examples of Module Contact Parameter Settings.....	29
2.06 Time Slot Mode Settings.....	30
2.07 SMS.....	31

# Real-time Slot

## DMR Product Protocol Document

### Attention:

1. After the analog circuit is powered on (PD is pulled high), delay for 5 seconds or wait for the analog circuit's *elevation machine*<sup>1</sup> to complete initialization (represented by the hexadecimal code: BF AA BB FA B3 F5 CA BC BB AF CD EA B3 C9).
2. After sending a command to the analog circuit, wait for the analog circuit to respond before sending the next command.

## 1 Serial Port Communication Protocol

The analog circuit supports configuration of functions like voice and SMS through the serial port. The serial port communication packet format is shown in Figure 1, and the field definitions are as described in Table 1.

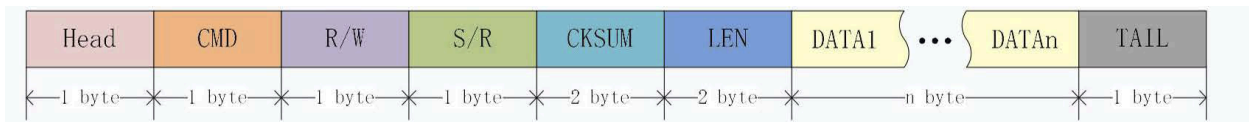


Figure 1: Serial Port Communication Packet Format

Table 1: Serial Port Communication Protocol Field Definitions<sup>2</sup>

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Start	0x68
1	CMD	1	Command	0x22: Reset digital channel 0x23: Reset analog channel 0x24: Check digital channel information 0x25: Check analog channel information 0x26: Reset transmission information 0x27: Check if the analog circuit is initialized 0x28: Set enhanced features 0x29: Set encryption function 0x2A: Set MIC gain

<sup>1</sup> The reference to "elevation machine" seems to be a literal translation and might refer to a specific process or component in the context of the protocol.

<sup>2</sup> This table appears to summarize all of the command/response packets, which are detailed in subsequent sections.

Offset	Flag	Length	Comment	Detail
				0x2B: Check digital voice reception information 0x2C: Send SMS 0x2D: Receive SMS 0x2E: Set volume 0x2F: Set gain 0x30: Set squelch 0x31: Set power saving mode 0x32: Check signal strength 0x33: Set repeater detachment mode 0x34: Check version information 0x35: Test input during cutting function 0x36: Print analog circuit status information (including voice reception, SMS reception) 0x37: Set polite strategy 0x38: Set analog/digital synchronization information 0x39: Check analog/digital synchronization information
2	R/W	1	Operation Mode	0x00: Read; 0x01: Write; (External CPU writes, external CPU read for display) 0x02: Main machine sends command
3	S/R	1	Set/Reply Command	Set: 0x01: Indicates beginning of setting Response: 0x00: Setting successful 0x01: Setting failed 0x02: Checksum error
4, 5	CKSUM	2	Checksum	-
6, 7	LEN	2	Data Segment Length	DATA: Data segment length, if there is no data segment information, then LEN value is 0x00
8	DATA	len	Data Segment Information	-
9	TAIL	1	Packet End	0x10

**Checksum Algorithm:**

```
C/C++
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);
}

```

Note: All data uses little-endian mode.

## 1.01 Set Digital Group Command (0x22)

```

C/C++
typedef struct {
    uint32 rx_freq;    // Receive frequency range: 400000000-480000000Hz (Default:
401.0250MHz)
    uint32 tx_freq;    // Transmit frequency range: 400000000-480000000Hz (Default:
401.0250MHz)
    uint32 localID;    // Local ID range: 1-16776415 (Default: 888)
    uint32 GroupList[32]; // Receive group list (Default: 1)
    uint32 tx_contact; // Contact code for transmission: 1-16776415 (All call
code: 0xFFFFF0-0xFFFFF) (Default: 1)
    uint8 ContactType; // Contact type: 0 individual call, 1 group call, 2 all
call (Default: 1)
    uint8 power;      // Power level: 0 low power, 1 high power (Default: 1)
    uint8 cc;         // Color code range: 0-15 (Default: 1)
    uint8 InboundSlot; // Inbound time slot: 0 or 1 (Default: 0)
    uint8 OutboundSlot; // Outbound time slot: 0 or 1 (Default: 0)
    uint8 ChannelMode; // Channel mode: 0 direct mode, 4 real-time slot (Default:
0)
    uint8 EncryptSw;  // Encryption switch: 1 enable, 2 disable (Default: 2)
    uint8 EncryptKey[8]; // Encryption key

```

```

uint8 pwrsave;    // Power saving mode: 2 disable, 1 enable (Default: 1)
uint8 volume;    // Volume level: 1-9 (Default: 8)
uint8 mic;       // Microphone gain: 0-5 (Default: 0)
uint8 relay;     // Repeater detachment: 2 disable, 1 enable (Default: 2)
} DB_DIGITAL_INFO;
    
```

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Start	0x68
1	CMD	1	Command	0x22
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	C 16-bit checksum value
6, 7	LEN	2	Data Segment Length	Length of the digital structure
8	DATA	0xA3	Data Segment Information	Fill in the digital structure as needed, the notes after the structure indicate the variable's range
9	TAIL	1	Packet End	0x10

**Response Packet:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Start	0x68
1	CMD	1	Command	0x22
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	C 16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00

Offset	Flag	Length	Comment	Detail
8	TAIL	1	Packet End	0x10

## 1.02 Set Analog Group Command (0x23)

C/C++

```
typedef struct {
    uint32 rx_freq;    // Receive frequency 400000000-480000000Hz
    uint32 tx_freq;    // Transmit frequency 400000000-480000000Hz
    uint8 band;        // 0: Narrowband, 1: Wideband
    uint8 power;       // 0: Low power, 1: High power
    uint8 sq;          // SQ level 0-9
    uint8 rx_type;     // 0: off, 1: CTCSS, 2: Forward DCS, 3: Reverse DCS
    uint8 rx_subcode; // 0-50, 0-82, 0-82 (depending on rx_type)
    uint8 tx_type;     // 0: off, 1: CTCSS, 2: Forward DCS, 3: Reverse DCS
    uint8 tx_subcode; // 0-50, 0-82, 0-82 (depending on tx_type)
    uint8 pwrsave;     // Power saving mode: 2: disable, 1: enable
    uint8 volume;      // Volume 1-9
    uint8 monitor;     // Monitor mode: 2: disable, 1: enable
    uint8 relay;       // Repeater detachment: 2: disable, 1: enable
} DB_ANALOG_INFO;
```

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x23
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	C 16-bit checksum value
6, 7	LEN	2	Data Segment Length	Length of the analog structure
8	DATA	0x13	Data Segment Information	Fill in the analog structure as needed, notes after the structure indicate the range of the variable
-	TAIL	1	Packet End	0x10

**Response Packet:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Start	0x68
1	CMD	1	Command	0x23
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	C 16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

**1.03 Set Transmission Command (0x26)**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x26
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	C 16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00 0x01
8	DATA	1	Data Segment Information	1: Start Transmission, 2: End Transmission
-	TAIL	1	Packet End	0x10

**Response Packet:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Start	0x68

1	CMD	1	Command	0x26
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	C 16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

### 1.04 Check if the Analog Circuit Initialization is Complete (0x27)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Start	0x68
1	CMD	1	Command	0x27
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	(16-bit checksum value)
6, 7	LEN	2	Data Segment Length	0x00 0x01
8	DATA	1	Data Segment Information	01
9	TAIL	1	Packet End	0x10

#### Response Packet:

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x27
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value



Offset	Flag	Length	Comment	Detail
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

### 1.05 Set Enhanced Function (0x28)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x28
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x05
8	FUN	1	Enhanced Function	0x01: Walkie-Talkie Detection, 0x02: Call Alert, 0x03: Remote Monitoring, 0x04: Walkie-Talkie Inhibition, 0x05: Walkie-Talkie Activation
9, 10, 11, 12	CallNum	4	Contact Code	1-16776415
12	TAIL	1	Packet End	0x10

#### Response Packet:

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x28
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error

4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

**Received Enhanced Function Response Packet 1:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x28
2	R/W	1	Operation Mode	0x02 (Main machine sends)
3	S/R	1	Set/Reply Command	0x00: Indicates success
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x05
8, 9, 10, 11, 12	DATA	5	Data Segment Information	First byte is the type of enhanced function: 0x02: Call Alert; Bytes 1–4 are the transmitter's local ID
13	TAIL	1	Packet End	0x10

**Received Enhanced Function Response Packet 2:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x0A
2	R/W	1	Operation Mode	0x02 (Main machine sends)
3	S/R	1	Set/Reply Command	0x00: Indicates success
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01

Offset	Flag	Length	Comment	Detail
8	DATA	1	Data Segment Information	0x04: Walkie-Talkie Inhibition, 0x05: Walkie-Talkie Activation
9	TAIL	1	Packet End	0x10

## 1.06 Set Encryption Function (0x29)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x29
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x09
8	SWITCH	1	Encryption Switch	1: enable, 2: disable
9-16	DATA	8	Encryption Key	Bytes 0-7 of the encryption key
17	TAIL	1	Packet End	0x10

### Response Packet:

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x29
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 1.07 Set MIC Gain (0x2A)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2A
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	GAIN	1	Gain Value	Range 0–5: 0: 0dB; 1: 4dB; 2: 8dB; 3: 12dB; 4: 16dB; 5: 20dB; default: 0
9	TAIL	1	Packet End	0x10

**Response Packet:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2A
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 1.08 Retrieve Digital Voice Reception Information (0x2B)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68

Offset	Flag	Length	Comment	Detail
1	CMD	1	Command	0x2B
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	-	1	-	01
9	TAIL	1	Packet End	0x10

**Response Packet:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2B
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x05
8-12	DATA		Data Segment Information	The 0th byte represents the call type: 0x00: Individual call 0x01: Group call 0x02: All call and broadcast  Bytes 1-4: Transmitter's local ID Bytes 5-8: Incoming call code
13	TAIL	1	Packet End	0x10

## 1.09 Send SMS (0x2C)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2C
2	R/W	1	Operation Mode	0x01
3	S/R	1	Set/Reply Command	0x01
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	
8	-	1	-	0x01: IP Confirmed 0x02: IP Not Confirmed 0x03: Group Call
8	Msg Type	1	Message Type	Contact ID: 4 Bytes
9–12	CallNum	4	Contact Code	Bytes 1–4: Transmitter's local ID, Bytes 5–8: Incoming call code
13	DATA	LEN-5	Data Segment Information	SMS format is Unicode encoded
13	TAIL	1	Packet End	0x10

## Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2B
2	R/W	1	Operation Mode	1
3	S/R	1	Set/Reply Command	1
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, LEN
8	Msg Type	1	Message Type	-

9-12	CallNum	4	Contact Code	-
13	DATA	LEN-5	Data Segment Information	-
14	TAIL	1	Packet End	0x10

## 1.10 Retrieve SMS (0x2D)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2D
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01
4, 5	CKSUM	2	Checksum	-
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Query Content	0x01
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2D
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x01
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	-
8	TYPE	1	Contact Type	0x00: Individual call, 0x01: Group call
9-12	CallID	4	SMS Sender Code	-
13	MsgData	LEN-5	SMS Content	SMS format in Unicode encoding

Offset	Flag	Length	Comment	Detail
14	TAIL	1	Packet End	0x10

## 1.11 Set Volume (0x2E)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2E
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	VOL	1	Volume Value	Range 1-9, where 1 is the lowest volume, 9 is the highest, default 8
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2E
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10



## 1.12 Set Monitor (0x2F)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2F
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	VOL	1		1: Enable, 2: Disable
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x2F
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 1.13 Set Squelch (0x30)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x30

Offset	Flag	Length	Comment	Detail
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	VOL	1	SQL	0-9
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x30
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

### 1.14 Set Power Saving (0x31)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x31
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value

6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	VOL	1		1: Enable, 2: Disable
9	TAIL	1	Packet End	0x10

**Response Packet:**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x31
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

**1.15 Retrieve Signal Strength (0x32)**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x32
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	Data	1	Data Segment Information	0x01
9	TAIL	1	Packet End	0x10

**Response Packet**

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x32
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Successful, 0x01: Failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	-
8	RSSI	1	Data Segment Information	Signal strength value
9	TAIL	1	Packet End	0x10

### 1.16 Set Repeater Decoupling Function (0x33)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x33
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	VOL	1		1: Enable, 2: Disable
9	TAIL	1	Packet End	0x10

#### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x33

2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

### 1.17 Retrieve Version Information (0x34)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x34
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Data	0x01
9	TAIL	1	Packet End	0x10

#### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x34
2	R/W	1	Operation Mode	0x00 (Interrogation Mode)
3	S/R	1	Set/Reply Command	0x00: Successful, 0x01: Failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value

6, 7	LEN	2	Data Segment Length	-
8	DATA	-	Software Version	Software version
9	TAIL	1	Packet End	0x10

### 1.18 Test Cutting Function in Progress (0x35)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x35
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Data	1: Increase, 2: Decrease, 3: Transmit
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x35
2	R/W	1	Operation Mode	0x00 (Interrogation Mode)
3	S/R	1	Set/Reply Command	0x00: Successful, 0x01: Failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 1.19 Module Power-On Status Information (0x36)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x36
2	R/W	1	Operation Mode	0x02: Main machine sends
3	S/R	1	Set/Reply Command	0x00: Setting successful
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Data	0x01: Reception start, 0x02: Reception end, 0x03: Transmission start, 0x04: Transmission end, 0x05: SMS received, 0x06: Repeater activation timeout, 0x07: Channel busy, transmission prohibited, 0x08: Confirmed SMS transmission successful, 0x09: Confirmed SMS transmission failed, 0x0A: Digital-analog synchronization digital reception start, 0x0B: Digital-analog synchronization digital reception end, 0x0C: Digital-analog synchronization analog reception start, 0x0D: Digital-analog synchronization analog reception start
9	TAIL	1	Packet End	0x10

Note: The module requires a phone-side ACK (68 36 01 01 95 87 00 01 01 10), otherwise it will retry 5 times.

## 1.20 Strategy (0x37)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x37
2	R/W	1	Operation Mode	0x01 (Write Mode)

3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Data	0: Impolite (non-courteous), 1: Polite to All (courteous to all terminals)
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x37
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 1.21 Set Digital-Analog Synchronization Check Command (0x38)

C/C++

```
typedef struct {
    uint32 rx_freq;    // Receive frequency range 400000000-480000000Hz
    uint32 tx_freq;    // Transmit frequency range 400000000-480000000Hz
    uint32 localID;    // Local ID range 1-16776415
    uint32 GroupList[32]; // Receive group list
    uint32 tx_contact; // Contact code 1-16776415, All call 0xFFFFF0-0xFFFFF
    uint8 ContactType; // Contact type 0: Individual call, 1: Group call, 2: All
    call, 3: Operational
    uint8 power;      // 0: Low power, 1: High power
    uint8 cc;         // Color code 0-15
}
```



```

uint8 InboundSlot; // Inbound slot, default 1
uint8 OutboundSlot; // Outbound slot, default 1
uint8 ChannelMode; // 0: Direct mode, 4: Real-time slot
uint8 EncryptSw; // Encryption switch 1: enable, 2: disable
uint8 EncryptKey[8]; // Encryption key
uint8 pwrsave; // Power saving mode 2: disable, 1: enable
uint8 volume; // Volume level 1-9
uint8 mic; // Microphone gain 0-5
uint8 relay; // Repeater decoupling 2: disable, 1: enable
uint8 sq; // SQ level 0-9
uint8 rx_type; // 1: CTCSS, 2: Forward DCS, 3: Reverse DCS
uint8 rx_subcode; // 0-50, 0-82, 0-82 based on rx_type
uint8 tx_type; // 1: CTCSS, 2: Forward DCS, 3: Reverse DCS
uint8 tx_subcode; // 0-50, 0-82, 0-82 based on tx_type
uint8 rxdly; // Receive delay 1-10s
} DB_MIX_CHECK_INFO;

```

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x38
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	Length of the digital structure
8	DATA	0xA9	Data Segment Information	Fill in the digital structure as needed, notes after the structure indicate the variable's range
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68

Offset	Flag	Length	Comment	Detail
1	CMD	1	Command	0x38
2	R/W	1	Operation Mode	0x00 (Interrogation Mode)
3	S/R	1	Set/Reply Command	0x00: Setting successful, 0x01: Setting failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 1.22 Set Short Message Transmission Concatenation Format (0x3A)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x3A
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Data	0x00: IP (default: 0), 0x01: Defined, 0x02: Compressed TCP/IP
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x3A

Offset	Flag	Length	Comment	Detail
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Successful, 0x01: Failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

### 1.23 Set TOT (Time-Out Timer) (0x3B)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x3B
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Data	Range 0–255, where 0 is disable TOT, 1–255 represents time in seconds, default: 120
9	TAIL	1	Packet End	0x10

#### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x3B
2	R/W	1	Operation Mode	0x00 (Read Mode)

Offset	Flag	Length	Comment	Detail
3	S/R	1	Set/Reply Command	0x00: Successful, 0x01: Failed, 0x02: Checksum error
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 1.24 Set spk\_en Mode Type (0x3C)

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x3C
2	R/W	1	Operation Mode	0x01 (Write Mode)
3	S/R	1	Set/Reply Command	0x01: Set
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x01
8	DATA	1	Data	Range 0–255, where 0 indicates no SMS notification, 1–255 represents time in milliseconds, default: 0
9	TAIL	1	Packet End	0x10

### Response Packet

Offset	Flag	Length	Comment	Detail
0	Head	1	Packet Header	0x68
1	CMD	1	Command	0x3C
2	R/W	1	Operation Mode	0x00 (Read Mode)
3	S/R	1	Set/Reply Command	0x00: Successful, 0x01: Failed, 0x02: Checksum error

Offset	Flag	Length	Comment	Detail
4, 5	CKSUM	2	Checksum	16-bit checksum value
6, 7	LEN	2	Data Segment Length	0x00, 0x00
8	TAIL	1	Packet End	0x10

## 2 DMR Module Basic Communication Usage Explanation

### 2.01 Digital Intercom<sup>3</sup>

#### Transmitter:

1. After the module is powered on, first send the "Set Digital Group Command" to the module.
2. Directly lower PTT or send "Set Transmission Command" to initiate voice transmission communication.

#### Receiver:

1. After the module is powered on, first send the "Set Digital Group Command" to the module.
2. When receiving voice signals from the transmitter, the module's SPK\_EN line will have a high-level output, and AUDIO\_OUT H will have an audio frequency output. When sending, increase the audio frequency amplifier to play the sound.

### 2.02 Analog Intercom

#### [Transmitter:]

1. After the module is powered on, first send the "Set Analog Group Command" to the module.
2. Directly lower PTT or send "Set Transmission Command" to initiate voice transmission communication.

#### Receiver:

1. After the module is powered on, first send the "Set Analog Group Command" to the module.
2. When receiving voice signals from the transmitter, the module's SPK\_EN line will have a high-level output and AUDIO\_OUT H will have an audio frequency output. When sending, increase the audio frequency amplifier to play the sound.

<sup>3</sup> A better translation for "intercom" here might be "communications."

## 2.03 Digital Intercom Contacts

### Contact Types:

1. Private call - 1 to 1 intercom.
2. Group call - 1 to N intercom.
3. All call - 1 to N intercom, similar to broadcast. Different from group call.

### Contact ID Range:

1. Private call - 1... 16776415.
2. Group call - 1... 16776415.
3. All call - 16777200... 16777215 (0xFFFFF0-0xFFFFF).

### Contact Correspondence:

1. Private call - Contact code = other party's local ID.
2. Group call - Contact code = any order ID in the other party's group list.
3. All call - Contact code = any order ID among the fixed 15 order IDs.

**Receiving Group List:** The receiving group list indicates the group call IDs that can be received in standby. Generally use the last order.

## 2.04 Digital Intercom Parameter Settings

### Essential Settings for Digital Intercom:

1. Receive/Transmit Frequency - must match exactly. (In relay intercom, it's non-specific, but receive/transmit must match, handheld TX = relay RX, handheld RX = relay TX)
2. Local ID - do not set specifically. (Must be non-specific in relay and when receiving SMS)
3. Color code - must be specific.
4. Contacts - depends on the type of use, must match.
5. Time slot - set according to the mode, generally set specifically.
6. Encryption key - must be specific. Or encryption can be disabled.

## 2.05 Examples of Module Contact Parameter Settings

The setup is as follows: Device A is the local device, and Device B is the other party. The following 9 sets of parameters allow A and B to communicate with each other:

1. A's contact: Private call 1, Local ID: 8, Group list ID: 6; B's contact: Private call 8, Local ID: 1, Group list ID is empty.
2. A's contact: Private call 1, Local ID: 8, Group list ID: 6; B's contact: Group call 6, Local ID: 1, Group list ID is empty.
3. A's contact: Private call 1, Local ID: 8, Group list ID: 6; B's contact: All call 16777215, Local ID: 1, Group list ID is empty.
4. A's contact: Group call 1, Local ID: 8, Group list ID: 6; B's contact: Private call 8, Local ID: 1, Group list ID: 1.

5. A's contact: Group call 1, Local ID: 8, Group list ID: 6; B's contact: Group call 6, Local ID: 1, Group list ID: 1.
6. A's contact: Group call 1, Local ID: 8, Group list ID: 6; B's contact: All call 16777215, Local ID: 1, Group list ID: 1.
7. A's contact: All call 16777215, Local ID: 8, Group list ID: 6; B's contact: Private call 8, Local ID: 1, Group list ID is empty.
8. A's contact: All call 16777215, Local ID: 8, Group list ID: 6; B's contact: Group call 6, Local ID: 1, Group list ID is empty.
9. A's contact: All call 16777215, Local ID: 8, Group list ID: 6; B's contact: All call 16777215, Local ID: 1, Group list ID is empty.

The main consideration when transmitting is the contact setting. For example, in examples 1, 2, and 3, if A's contact is set to Private call 1, as long as B's local ID is 1, regardless of B's contact and group list settings, A can communicate with B. However, if B's local ID is any other number, regardless of B's contact and group list settings, A cannot communicate with B.

In examples 4, 5, and 6, if A's contact is set to Group call 1, as long as there's any ID in B's group list set to 1, regardless of B's contact settings, A can communicate with B. However, if there's no ID in B's group list set to 1, regardless of B's contact settings and other IDs in the group list, A cannot communicate with B.

In examples 7, 8, and 9, if A's contact is set to All call 16777215, regardless of B's contact and group list settings, A can communicate with B.

For the transmitting device, as long as the contact ID meets the requirements of the contact type, it can transmit. For example, for Private call, the contact code is the other party's local ID. For Group call, the contact code is any ID in the other party's group list. For All Call, the contact code is any ID among the 15 orders of All Call.

For the receiving device, as long as one of the three conditions is met, it can receive. For the local ID condition, as long as the transmitting device's contact uses Private call for transmission and the contact code matches (=) the local ID, it can receive. For the receiving group ID condition, as long as the transmitting device's contact uses Group call for transmission and the contact code matches any ID in the group, it can receive. For the All call condition, as long as the transmitting device's contact uses All call for transmission, any contact code from the 15 orders of All call can be received.

## 2.06 Time Slot Mode Settings

During direct intercom: direct intercom between handheld devices.

In direct mode, time slots 1 and 2 can communicate with each other.<sup>4</sup>

---

<sup>4</sup> This implies the unit operates simultaneously on both slots, which would play havoc with a repeater. Perhaps they mean "simplex" mode here, HT to HT?

In real-time slot mode, time slots 1 and 2 cannot communicate with each other; they must be matched one-to-one for communication.

During relay intercom: communication between handheld devices through a relay station.  
In direct mode and real-time slot mode, time slots 1 and 2 cannot communicate with each other; they must be matched one-to-one for communication.

## 2.07 SMS

Requirements for Sending SMS:

1. Local ID should not be set specifically.
2. When sending SMS in 2 time slots, the interval should not be less than 1.5 seconds.