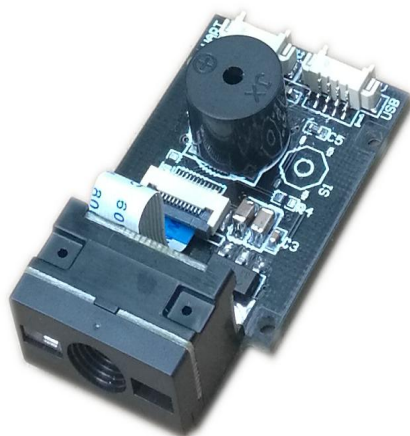




# GM65 Bar Code Reader Module User Manual



Hangzhou Grow Technology Co., Ltd.  
V1.3 Sept. 2016

---

# Catalog

1.	Introduction of Module.....	3
1.1	Introduction.....	3
1.2	Technical Specification.....	3
1.3	Dimension(mm).....	2
	Control Panel.....	2
	Scan Area(testing in office (250 lux).....	3
1.4	Setup Code.....	4
1.5	Appendix E: Save or Cancel.....	4
1.6	Reset.....	4
2	Communication interface.....	5
2.1	Series Communication Interface.....	5
2.2	USB Interface.....	6
2.3	USB Virtual Serial Port.....	6
3	Read Mode.....	7
3.1	Continuous Mode.....	7
3.2	Induction Mode.....	8
3.3	Manual Mode.....	9
3.4	Command Triggered Mode.....	10
4	Lighting and Collimate.....	11
4.1	Lighting.....	11
4.2	Collimation.....	11
5	Prompts.....	12
5.1	Prompts Tone.....	12
5.2	Decode successfully tone.....	12
5.3	Decoding Prompt.....	13
5.4	Data code format.....	13
5.5	Keyboard Settlement.....	13
5.6	Image Flip.....	14
6	Data Edition.....	15
6.1	Prefix.....	15
6.2	Suffix.....	16
6.3	CODE ID.....	16
6.4	Tail.....	19
6.5	Cut out Data.....	19
6.6	RF Information.....	21
7	Bar code type enables/disable configuration.....	22
7.1	All types of bar code can be decoded.....	22
7.2	Bar code rotate.....	22
7.3	EAN13.....	22
7.4	EAN8.....	23

---

7.5	UPCA.....	23
7.6	UPCE0.....	23
7.7	UPCE1.....	23
7.8	Code128.....	23
7.9	Code39.....	24
7.10	Code93.....	24
7.11	CodeBar.....	25
7.12	QR.....	26
7.13	Interleaved 2 of 5.....	26
7.14	Industrial 25.....	26
7.15	Matrix 2 of 5.....	27
7.16	Code11.....	27
7.17	MSI.....	28
7.18	RS.....	29
7.19	DM.....	29
7.20	PDF417.....	30
8	Serial Port Instruction.....	31
	8.1CRC Algorithm.....	31
	8.2 Read Zone Bit.....	32
	8.3 Write Zone Bit.....	33
	8.4 Save zone bit to EEPROM Command.....	34
	8.5 List of zone bit.....	34
9	Appendix A: Common serial port instruction.....	48
10	Appendix B: Code ID List.....	49
11	Appendix C: ASCII.....	50
12	Appendix D: data code.....	55

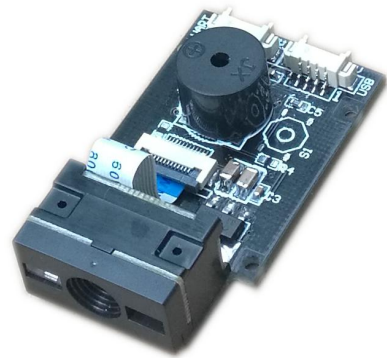
# 1. Introduction of Module

## 1.1 Introduction

MG65 Bar code reader module is a high performance scanner, can read 1D bar code easily and read 2D bar code with high speed. It also wins high scan speed for linear code, even for bar code on paper or screen.

MG65 bar code reader module is an advanced bar code decoding algorithm which developed on image recognition algorithm, can easily and accurately read bar code, simplify secondary development.

MG65 works stable in dark and large temperature range.



## 1.2 Technical Specification

Default scan mode		Continuous scan	
Read code time for once		3s	Parameter: 0.1-25.5s; step-size: 0.1s; 0 means no time limited
Reading interval		1S	Parameter: 0.1-25.5s; step-size: 0.1s; 0 means no time limited
Output		GBK	GBK、UNICODE、BIG5
Interface		USB	USB 、UART、USB VCom
Interface ( TTL-232)	Serial Baud Rate	9600	adjustable, details at 2.1
	Verification	N	
	Data bit	8	
	Stop bit	1	
	CTSRTS	No	
serial mode	Read code time for once	5s	Parameter: 0.1-25.5s; step-size: 0.1s; 0 means no time limited

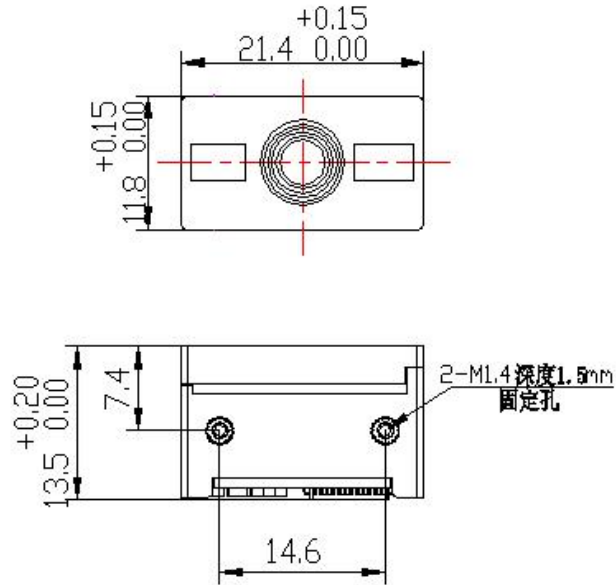
## Electrical specification:

1	Operating Voltage	DC 4.2 - 6.0V
2	Standby Current	30mA
3	Operating Current	160mA
4	Sleep Current	3mA

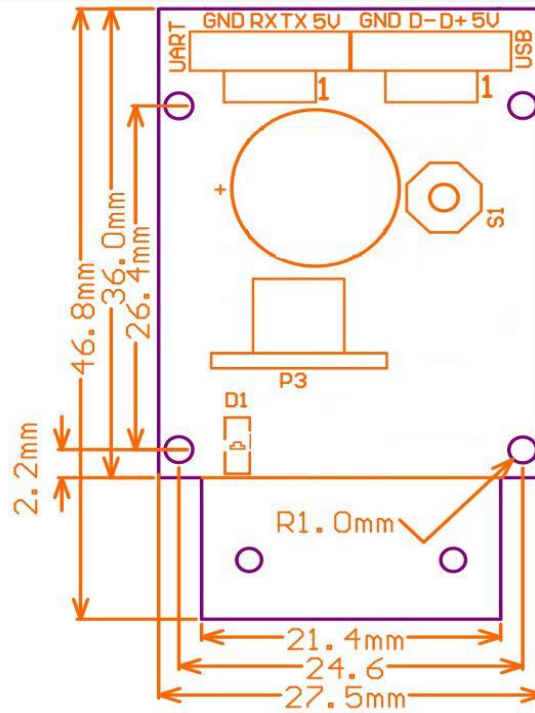
## Running characteristic

1	Light	White light
2	Capture light	Red
3	Scan Angle	Roll:0-360° , Pitch:±65° , Yaw:±60°
4	Resolution	648x 488
5	Scanning angle	35° (Inclination) , 28° (Elevation)

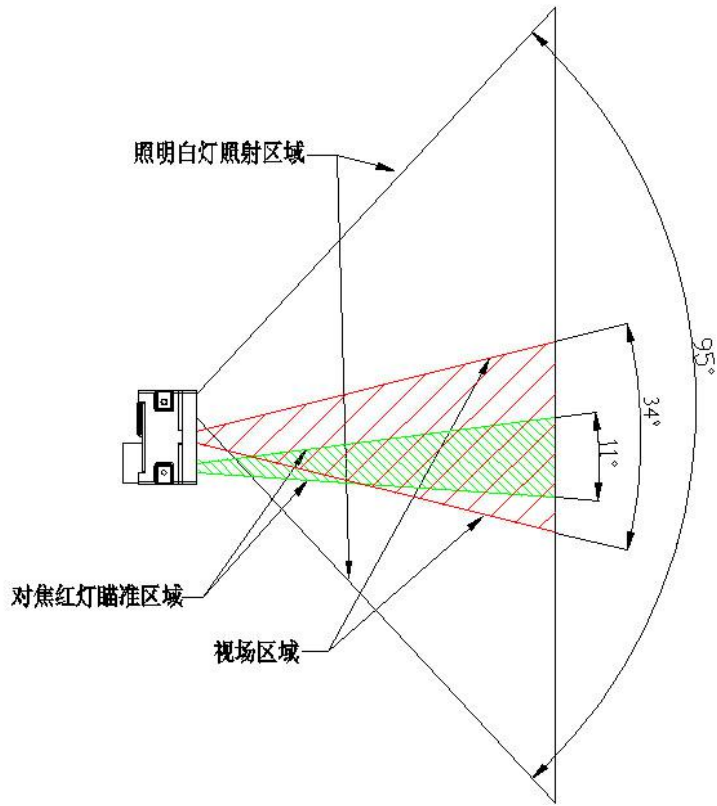
### 1.3 Dimension(mm)



### Control Panel



**Scan Area (testing in office (250 lux))**



Type of Bar Code	Density	Min. distance	Max. distance
Code 39	0.125 mm (5 mils)	4.0 cm	9.0 cm
	0.375 mm (15 mils)	4.0 cm	25.0cm
UPC/EAN	0.375 mm (15 mils)	4.0 cm	25.0cm
Code93	0.254 mm (10 mils)	4.0 cm	21.0cm

## 1.4 Setup Code

Customer can set module by scan setup code.



**Default: setup code on**



**Off**

Output details in setup code



**Default: Not output**



**Output**

## 1.5 Save and Cancel

Scan cancel code to cancel wrong read date.



**Save**



**Cancel last byte**



**Cancel all read bytes**



**Cancel changes**

## 1.6 Reset

Back to Factory Setting by scan follow code.



**Reset**



## 2 Communication interface

GM65 can receive database, control module and set functional parameter by TTL - 232.

### 2.1 Series Communication Interface

It's default and common to connect module and mainframe(such as PC, POS) by series communication interface. Make sure communication parameter for module and mainframe are same, then it will communicate smooth and correctly.



**Series Output**

TTL-232 is used for series interface which suitable for most system. Required change-over circuit for RS-232.

Default Parameter as formForm 2- 1. Only Baud Rate can be changed.

Form 2-1 Default Parameters

Parameters	Default
Series communication interface	Standard TTL-232
Baud rate	9600
Verification	N
Data bit	8
Stop bit	1
CTSRTS	N

#### Baud Rate Settlement



**1200bps**



**4800bps**



**9600bps(Default)**



**14400bps**



**19200bps**



38400bps



57600bps



115200bps

## 2.2 USB Interface

Scan the following code to become standard keyboard input pattern when module connects PC by USB.



USB PC Keyboard

## 2.3 USB Virtual Serial Port

Scan the following code to become virtual serial port output pattern when module connects PC by USB.



USB Virtual serial port

### 3 Read Mode

#### 3.1 Continuous Mode

On this mode, reading module read code continuous and automatic.

Break after reading one code, break time is changeable.

Click the toggle key to pause. Then click to continuous cyclic read code.



Default Continuous Mode

##### Time settlement for single read

The longest time before first successful reading. After this time, module will be into no read time.

Single Read time: 0.1~25.5 s, step-size: 0.1s;

0 means infinite time interval.

Default time: 5s



1000ms



5000ms



default 3000ms



infinite time interval

##### Break time settlement

Time between two read. Can be settled from 1 to 25.5 s, step-size: 0.1s; default 1.0s



No break



500ms



default 1000ms



1500ms



2000ms

### 3.2 Induction Mode

After setting, module begins to monitor brightness immediately. When scene changed, module will begin to read until time of image stabilization over.

After first successful reading or single reading time out, module will monitor brightness again after some time (changeable)

Module will cycle working as above when follow happen: module can't find code between single read time, then it will stop reading and jump to monitor brightness.

On induction mode, module can begin reading code by click, and it will begin to monitor brightness when release toggle key or successfully output information.



Induction Mode

#### Time settlement for single read

The longest time read before first successful reading. After this time, module will be into no read time.

Single Read time: 0.1~25.5 s, step-size: 0.1s;

0 means infinite time interval.

Default time: 5s



1000ms



3000ms



Default 5000ms



infinite time interval

### Break time settlement

After one successful output or time out for single read. Module will be into monitor after some time.

Time from 0 to 25.5 s, step-size: 0.1s; default 1.0s



No Break



500ms



Default 1000ms



1500ms



2000ms

### Image stabilization time

Image stabilization time: the time cost after module find scene change then waiting for the scene stable. Time from 0s to 25.5s, step size 0.1s. Default 0.4s.



100ms



Default 400ms



1000ms



2000ms

## 3.3 Manual Mode

Manual mode is default mode. Click toggle key begin to read, stop when output or release toggle key



Manual Mode

Into sleep mode after not work for a while, can be settled by following code.



**Sleep mode on**



**Sleep mode off**

Default: come into sleep mode after 30s free. Wake up by key. Module will restart after dropping out sleep mode.

### 3.4 Command Triggered Mode

Module begins to read when receive scan command from mainframe( bit0 of zone bit 0x0002 writes"1" ) , and stop at output or read timeout.



**Command triggered mode**

Under command triggered mode, command for serial port trigger is 7E 00 08 01 00 02 01 AB CD;

After receiving command, model will output " 02 00 00 01 00 33 31" and start scan.

#### **Time settlement for single read**

Read and Scan time before output. From 0.1s to 25.5s, step size 0.1s. 0 : infinite time interval;  
Default: 5s



**1000ms**



**3000ms**



**Default 5000ms**



**infinite time interval**

## 4 Lighting and Collimate

### 4.1 Lighting

Head lamp is used to additional lighting when read.

Normal(default): Head lamp will be on when read, others off.

Normally on: always on after boot.

OFF: head lamp is always off



**Normal**



**Normally on**



**Off**

### 4.2 Collimation

There will be a pointing light beam which can help user to find best distance.

Normal(default): pointing light beam shows when read

Normally on: pointing light beam shows after power on until power off

No Collimation: no pointing light beam



**Normal(default)**



**Normally on**



**No Collimation**

## 5 Prompts

### 5.1 Prompts Tone

Sound of buzzer



UP



Down

Silence: Close all prompt tones



Silence on



Silence off

### 5.2 Read code successfully tone



Default on



Close read code successfully tone

Duration time setting. Default: 60ms



30ms



60ms



90ms



120ms



### 5.3 Decoding Prompt

Output Decoding Prompt: F- read unsuccessfully; S- successful



Default no decoding prompt output



Output

### 5.4 Data code format

Read following to print Chinese data according to specified code forma.

GBK format: notepad, UNICODE format: WORD and input box for chat tools,

BIG5 format: Chinese Traditional



GBK



UNICODE



BIG5

### 5.5 Keyboard Settlement

Scan following code to change keyboard.



Default Keyboard- US



Czech



France



Germany



Hungary



Italy



Japan



Spain

## 5.6 Image Flip

When meet flipped or mirror image, like:



原始图像

镜像翻转

Scan code into or exit image flip mode



image flip mode



Exit

PS: in the image flip mode can only scan flipped image. For normal, should exit.

## 6 Data Edition

Sometimes we need to edit the data before output to make data separation and processing more easily.

Data edition include:

- Add Prefix
- Add Suffix
- Cut data
- Output CodeID
- Output “RF” when fail to decode
- Add End words”Tail”

Output sequence after data edition:

【Prefix】 【CodeID】 【Data】 【Suffix】 【Tail】

### 6.1 Prefix

#### Add prefix

Prefix is on the head of encoding Information , and can be self-defined.

Scan the code to add prefix.



Allow to add prefix



Default no prefix

#### Change prefix

Scan “change prefix” and “setup code” code to change prefix.

Use 2 base 16 to express each character.

Max 15 characters.

ASCII on appendix C.



change prefix

**E.G. Change prefix to “DATA”**

1. "DTAT" in base 16: "44", "41", "54", "41"
2. Confirm open the " setup code", if not, find on 1.4
3. Scan "change prefix" code
4. Successively scan "Code ID": "4", "4", "4", "1", "5", "4", "4", "1"

**6.2 Suffix**

**Add Suffix**

Suffix on the end of encoding Information, and can be self-defined.



**Allow add suffix**



**Default no suffix**

**Change suffix**

Scan " change suffix" and "setup code" code to change prefix.

Use base 16 to express each character.

Max 15 characters. ASCII on appendix C.



**Change Suffix**

**E.G.: Change suffix to “DATA”**

1. "DTAT" in base 16: "44", "41", "54", "41"
2. Confirm opening the "setup code", if not, find on 1.4
3. Scan "change suffix" code
4. Successively scan "Code ID": "4", "4", "4", "1", "5", "4", "4", "1"
5. Scan "save" code

**6.3 CODE ID**

**Add CODE ID**

Users can identify different types of bar code by CODE ID. CODE ID use one character to identify and can be self- defined.



**Allow add CODE ID**



**Default close CODE ID**

**Default of CODE ID**

Scan “ Default of CODE ID” to back default ID, default ID on appendix B



**all bar code back to default ID**

**Change CODE ID**

Users can change CODE ID of any bar code by scan the setup code (as follow) and data edition code.

Base 16 is used to express each CODE ID.

ASCII on appendix C.

**E.G.: change CODE ID of CODE 128 to “A”**

1. Find “A”=”41” in base 16
2. Confirm opening the “setup code”, if not, find on 1.4
3. Scan “ change CODE 128”
4. Successively scan “Code ID”: “4”, “1”
5. Scan “save” code

**Change CODE ID LIST**



**Change CODE ID of EAN13**



**Change CODE ID of EAN8**



**Change CODE ID of UPCA**



**Change CODE ID of UPCE0**



Change CODE ID of UPCE1



Change CODE ID of CODE 128



Change CODE ID of CODE 39



Change CODE ID of CODE 93



Change CODE ID of CODE BAR



Change CODE ID of Interleaved 2 of 5



Change CODE ID of Industrial 25



Change CODE ID of Matrix 2 of 5



Change CODE ID of CODE 11



Change CODE ID of MSI



Change CODE ID of RSS



Change CODE ID of limited RSS



Change CODE ID of expended RSS



Change CODE ID of QR CODE



Change CODE ID of Data Matrix



Change CODE ID of limited PDF417

## 6.4 Tail

Open this function to help system quickly distinguish current decoding results.

Scan “Add tail” to open this function, if read success, there will be tail on the end of decode data.



Close tail



Add tail “CR”



Add tail “TAB”



Add tail “CRLF”

## 6.5 Cut out Data

Open to output part of data.

[Data] is composed of [Start] + [Center] + [End]

Character length of “start” and “end” can be changed



Output whole data



Output Start part



Output End Part



Output Center part

**Change length of [Start]-M**

Scan “ Change M” code and “data edition” code to change length of [Start], max 255 characters

Base 16 is used to express length. ASCII on appendix C.



**Change M**

**Change length of [End]-N**

Scan “ Change N” code and “data edition” code to change length of [Start], max 255 characters

Base 16 is used to express length. ASCII on appendix C.



**Change N**

**Output Start part**

**E.G. Output “1234567890123” of whole decode information “ 1234567890123ABC”**

1. “13” =”0D” in base 16
2. Confirm opening the “setup code”, if not, find on 1.4
3. Scan” change length M”
4. Successively scan “Code ID”: “0”, “D”
5. Scan “save” code
6. Scan” Output Start part”

**Output End Part**

**E.G. Output “ABC” of whole decode information “ 1234567890123ABC”**

1. “3” = “03” in base 16
2. Confirm opening the “setup code” , if not, find on 1.4
3. Scan ”change length N”
4. Successively scan “Code ID”:.”0”, “3”
5. Scan “save” code
6. Scan” Output Start part”



### Output Center part

**E.G.: Output “0123” of whole decode information “1234567890123ABC”**

1. “10” = “0A”; “3” = “03” in base16
2. Confirm opening the “setup code”, if not, find on 1.4
3. Scan “change length N”
4. Successively scan “Code ID”: “0”, “3”
5. Scan “save” code
6. Scan “change length M”
7. Successively scan “Code ID”: “0”, “A”
8. Scan “save” code
9. Scan “Output Center part”

## 6.6 RF Information

RF(Read Fail): Users can self- define output information when read fail.



Output RF information



Default not output

### Change RF information

Scan “change RF information” and “data edition code” to change RF information.

Base 16 is used to express, max at 15 character. ASCII on appendix C.



Change RF information

**E.G.: change RF to “FAIL”**

1. Find “FAIL” in base-16: “46”, “41”, “49”, “4C”
2. Confirm opening the “setup code”, if not, find on 1.4
3. Scan “change RF information”
4. Successively scan “4”、“6”、“4”、“1”、“4”、“9”、“4”、“C”
5. Scan “save” code

## 7 Bar code type enables/disable configuration

### 7.1 All types of bar code can be decoded

After scan “ Forbid read all bar code” , module will only support to scan setup code.



Support all



Forbid read all bar code



Open default support types

### 7.2 Bar code rotate



Default allow 360° rotate read



Forbid 360° rotate read

### 7.3 EAN13



Allow reading EAN13



Forbid reading EAN13

### 7.4 EAN8



Allow reading EAN8



Forbid reading EAN8

### 7.5 UPCA



Allow reading UPCA



Forbid reading UPCA

### 7.6 UPCE0



Allow reading UPCE0

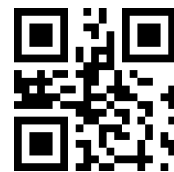


Forbid reading UPCE0

### 7.7 UPCE1



Allow reading UPCE1



Forbid reading UPCE1

### 7.8 Code128



Allow reading Code128



Forbid reading Code128

Scan following code to change min length of code 128



**Code128 min length at 0**



**Code128 min length at 4**

Scan following code to change max length of code 128



**Code128 max length at 32**



**Code128 max length at 255**

## 7.9 Code39



**Allow reading Code39**



**Forbid reading Code39**

Scan following code to change min length of code39



**Code39 min length at 0**



**\*Code39 min length at 4**

Scan following code to change max length of code39



**Code39 max length at 32**



**Code39 max length at 255**

## 7.10 Code93



**Allow reading Code93**



**Forbid reading Code93**

Scan following code to change min length of code93



**Code93 min length at 0**



**Code93 min length at 4**

Scan following code to change max length of code93



**Code93 max length at 32**



**Code93 max length at 255**

### 7.11 CodeBar



**Allow reading CodeBar**



**Forbid reading CodeBar**

Scan following code to allow/forbid to send Start-stop operator



**Send CodeBar Start-stop operator**



**Not send CodeBar Start-stop operator**

Scan following code to change min length of CodeBar



**CodeBar min length at 0**



**CodeBar min length at 4**

Scan following code to change min length of CodeBar



**CodeBar max length at 32**



**CodeBar max length at 255**

### 7.12 QR



Allow reading QR



Forbid reading QR

### 7.13 Interleaved 2 of 5



Allow reading Interleaved 2 of 5



Default Forbid reading Interleaved 2 of 5

Scan following code to set min length of Interleaved 2 of 5



Interleaved 2 of 5 min length at 0



Default Interleaved 2 of 5 min length at 4

Scan following code to set max length of Interleaved 2 of 5



Interleaved 2 of 5 max length at 32



Interleaved 2 of 5 max length at 255

### 7.14 Industrial 25



Allow reading Industrial 25



Forbid reading Industrial 25

Scan following code to set min length of Interleaved 25



**Industrial 25 min length at 0**



**Industrial 25 min length at 4**

Scan following code to set max length of Interleaved 25



**Industrial 25 max length at 32**



**Industrial 25 max length at 255**

### 7.15 Matrix 2 of 5



**Allow reading Matrix 2 of 5**



**Forbid reading Matrix 2 of 5**

Scan following code to set min length of Matrix 2 of 5



**Matrix 2 of 5 min length at 0**



**Matrix 2 of 5 min length at 4**

Scan following code to set max length of Matrix 2 of 5



**Matrix 2 of 5 max length at 32**



**Matrix 2 of 5 max length at 255**

### 7.16 Code11



**Allow reading Code11**

Scan following code to set min length of Code11



**Code11 min length at 0**

Scan following code to set max length of Code11



**Code11 max length at 32**

**Forbid reading Code11**



**Code11 min length at 4**



**Code11 max length at 255**

**7.17 MSI**



**Allow reading MSI**

Scan following code to set min length of MSI



**MSI min length at 0**

Scan following code to set max length of MSI



**MSI max length at 32**



**Forbid reading MSI**



**\*MSI min length at 4**



**MSI max length at 255**



## 7.18RS



Allow reading RSS-14



Forbid reading RSS-14



Allow reading limited RSS



Forbid reading limited RSS



Allow reading expended RSS



Forbid reading expended RSS

Scan following code to set min length of RSS



RSS min length at 0



RSS min length at 4

Scan following code to set max length of RSS



\*RSS max length at 32



RSS max length at 255

## 7.19DM



Allow reading DM



Forbid reading DM

## 7.20 PDF417



**Allow reading PDF417**



**Forbid reading PDF417**

## 8 Serial Port Instruction

Users can settle the module by sending instruction from mainframe.

Please make sure communicate parameter complete matching between module and mainframe.

Module default serial communicate parameter: Baud rate 9600bps; No check; 8 bit data; 1 bit stop bit; No flow control.

### 8.1CRC Algorithm

CRC: CRC\_CCITT check value (2 bytes).

Suitable for Types、Lens、 Address、 Datas;

Characteristic polynomial :  $X^{16}+X^{12}+X^5+1$ , multinomial coefficient: 0x1021, original value:0 ;

For single byte, the highest bit will be calculated at first, output will be without negation.

Reference code of C:

```

unsigned int crc_cal_by_bit(unsigned char* ptr, unsigned int len)
{
    unsigned int crc = 0;
    while(len-- != 0)
    {
        for(unsigned char i = 0x80; i != 0; i /= 2)
        {
            crc *= 2;
            if((crc&0x10000) !=0)
                crc ^= 0x11021;
            if((*ptr&i) != 0)
                crc ^= 0x1021;
        }
        ptr++;
    }
    return crc;
}

```

PS: when no need for checking CRC, CRC bite can be filled in 0xAB 0xCD

## 8.2 Read Zone Bit

Max 256 bytes/time for zone bit reading.

Send Command Format:

Numbers of bytes	2bytes	1byte	1byte	2bytes	1byte	2bytes
Name	Head	Types	Lens	Address	Datas	CRC
Details	0x7E00	07H	01H	xxH	xxH	xxH

PS: Address : 0x0000~0x00FF (2 bytes) , address to start reading zone bit

Datas : 0x00~0xFF (1 byte) , Numbers of zone bit for Sequential read, 0x00= 256bytes

Receive Command Format:

Numbers of bytes	2bytes	1byte	1byte	1byte	2bytes
Name	Head	Types	Lens	Datas	CRC
Details	0x0200	00H	xxH	xxH	xxH

PS: Types: 0x00= read succeed

Lens : numbers of upload bytes; 0x00= 256bytes

Datas : 0x00~0xFF means read data.

If CRC wrong, no return data

If format or data wrong, no return data

### E.G.:

Read address 0x000A of Zone bit

1) Read successfully and return data is 0x3E.

Input: 0x7E 0x00 0x07 0x01 0x00 0x0A 0x01 0xEE 0x8A

Output: 0x02 0x00 0x00 0x01 0x3E 0xE4 0xAC

2) CRC wrong

Input: 0x7E 0x00 0x07 0x01 0x00 0x0A 0x01 0x11 0x22

Output : None

3) When length of command to short or more than 400ms after 0x7e 0x00, treat as unknown command.

Input: 0x7E 0x00 0x07 0x01 0x00 0x0A 0x01

Output: None

### 8.3 Write Zone Bit

Max 245 bytes for one time write

Send Command Format:

Numbers of bytes	2bytes	1byte	1byte	2bytes	1byte	2bytes
Name	Head	Types	Lens	Address	Datas	CRC
Detail	0x7E00	08H	01H	xxH	xxH	xxH

PS: Lens: 0x00~0xFF ( 1 byte ) , means numbers of bytes of this datas, times of continuous writing.

0x00 means 256bytes

Address : 0x0000~0xFFFF ( 2 bytes ) Start location of write

Datas : 0x00~0xFF ( 1~256 bytes ) , dates wrote in zone bit

Receive Command Format:

Numbers of bytes	2bytes	1byte	1byte	1byte	2bytes
Name	Head	Types	Lens	Datas	CRC
Detail	0x0200	00H	01H	00H	0x3331

PS: Types: 0x00 write succeed

If CRC wrong, no return data

If format or data wrong, no return data

#### E.G.:

Write 0x3E in 0x000A of zone bit

1) successfully set

Input : 0x7E 0x00 0x08 0x01 0x00 0x0A 0x3E 0x4C 0xCF

Output: 0x02 0x00 0x00 0x01 0x00 0x33 0x31

2) CRC wrong

Input: 0x7E 0x00 0x08 0x01 0x00 0x0A 0x3E 0x11 0x22

Output: None

3) When length of command to short or more than 400ms after 0x7e 0x00, treat as unknown command.

Input: 0x7E 0x00 0x08 0x01 0x00 0x0A 0x3E

Output: None

## 8.4 Save zone bit to EEPROM Command

Send Command Format:

Numbers of bytes	2bytes	1byte	1byte	2bytes	1byte	2bytes
Name	Head	Types	Lens	Address	Datas	CRC
Detail	0x7E00	09H	01H	0x0000	00H	0xDEC8

Datas Receive Format:

Numbers of bytes	2bytes	1byte	1byte	1byte	2bytes
Name	Head	Types	Lens	Datas	CRC
Detail	0x0200	00H	01H	00H	0x3331

PS: Types : 0x00 = succeed

If CRC wrong, no return data

If format or data wrong, no return data

## 8.5 List of zone bit

Zone bit	0x0000		
Data bit	Function		
Bit 7	1: Open LED when successfully read 0: Close		
Bit 6	1: Mute off	0: Mute on	
Bit 5-4	00: No aim	01: Standard	10/11: always on
Bit 3-2	00: No light	01: Standard	10/11: Always on
Bit 1-0	00: Manual mode 11: Sensor mode	01: Command mode	10: Continuous mode
Zone bit	0x0002		
Data bit	Function		

Bit 7	Keep
Bit 6	1: Output successfully decode prompt                      0: Not output
Bit5-1	Keep
Bit0	Keep
Zone bit	0x0003
Data bit	Function
Bit 7-2	Keep
Bit 1	1: Close Settlement Code                      0: Open
Bit0	1: Output content of settlement code                      0: Not output
Zone bit	0x0004
Data bit	Function
Bit 7-0	Time for image stabilization 0x00-0xFF: 0.0-25.5s
Zone bit	0x0005
Data bit	Function
Bit 7-0	Read interval 0x00-0xFF: 0.0-25.5s
Zone bit	0x0006
Data bit	Function
Bit 7-0	Time for single read 0x00-0xFF: 0.0-25.5s
Zone bit	0x0007
Data bit	Function
Bit 7	Auto sleep 1: On                      0: Off
Bit6-0	Free time (high order Bit14-8) Unit: 100ms
Zone bit	0x0008

Data bit	Function
Bit 7-0	Free time (low order Bit7-0) Unit: 100ms
Zone bit	0x000A
Data bit	Function
Bit 7-0	Successfully read suond 0x00: Piezo Buzzer 0x01-0xFF: sfn, Value*20;
Zone bit	0x000B
Data bit	Function
Bit 7-0	Time duration for successfully read sound 0x00-0xFF; 0-255ms
Zone bit	0x000C
Data bit	Function
Bit 7-1	Keep
Bit0	Piezo Buzzer 0: high level when free, low level when busy      1: low level when free, high level when busy
Zone bit	0x000D
Data bit	Function
Bit 7-4	Keep
Bit 3-2	Data encoding format 00: GBK                      01: UNICODE                      10: BIG5                      11: Keep
Bit1-0	00: serial port output                      01: USB PC Keyboard                      10: Keep 11: USB virtual serial port
Zone bit	0x000E
Data bit	Function
Bit 7-3	keep
Bit2	1: Decode successfully sound on                      0: Decode successfully sound off
Bit1-0	Keep





Bit 7-1	Keep
Bit0	Read UPCA 0: Forbid      1: Allow
Zone bit	<i>0x0031</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read UPCE0 0: Forbid      1: Allow
Zone bit	<i>0x0032</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read UPCE1 0: Forbid      1: Allow
Zone bit	<i>0x0033</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read Code128 0: Forbid      1: Allow
Zone bit	<i>0x0034</i>
Data bit	Function
Bit 7-0	Min length of Code128 0x00-0xFF: 0-255Byte
Zone bit	<i>0x0035</i>
Data bit	Function
Bit 7-0	Max length of Code128 0x00-0xFF: 0-255Byte
Zone bit	<i>0x0036</i>
Data bit	Function

Bit 7-1	Keep
Bit0	Read Code39 0: Forbid      1: Allow
Zone bit	0x0037
Data bit	Function
Bit 7-0	Min length of Code39 0x00-0xFF: 0-255Byte
Zone bit	0x0038
Data bit	Function
Bit 7-0	Max length of Code39 0x00-0xFF: 0-255Byte
Zone bit	0x0039
Data bit	Function
Bit 7-1	Keep
Bit0	Read Code93 0: Forbid      1: Allow
Zone bit	0x003A
Data bit	Function
Bit 7-0	Min length of Code93 0x00-0xFF: 0-255Byte
Zone bit	0x003B
Data bit	Function
Bit 7-0	Max length of Code93 0x00-0xFF: 0-255Byte
Zone bit	0x003C
Data bit	Function
Bit 7-2	Keep
Bit1	Send CodeBar Start-stop operator 0: Forbid   1: Allow

Bit0	Read CodeBar 0: Forbid      1: Allow
Zone bit	<i>0x003D</i>
Data bit	Function
Bit 7-0	Min length of CodeBar 0x00-0xFF: 0-255Byte
Zone bit	<i>0x003E</i>
Data bit	Function
Bit 7-0	Max length of CodeBar 0x00-0xFF: 0-255Byte
Zone bit	<i>0x003F</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read QR 0: forbid      1: allow
Zone bit	<i>0x0040</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read Interleaved 2 of 5 0: Forbid      1: Allow
Zone bit	<i>0x0041</i>
Data bit	Function
Bit 7-0	Min length of Interleaved 2 of 5 0x00-0xFF: 0-255Byte
Zone bit	<i>0x0042</i>
Data bit	Function
Bit 7-0	Max length of Interleaved 2 of 5 0x00-0xFF: 0-255Byte
Zone bit	<i>0x0043</i>
Data bit	Function

Bit 7-1	Keep
Bit0	Read Industrial 25 0: forbid      1: Allow
Zone bit	0x0044
Data bit	Function
Bit 7-0	Min length of Industrial 25 0x00-0xFF: 0-255Byte
Zone bit	0x0045
Data bit	Function
Bit 7-0	Max length of Industrial 25 0x00-0xFF: 0-255Byte
Zone bit	0x0046
Data bit	Function
Bit 7-1	Keep
Bit0	Read Matrix 2 of 5 0: forbid      1: Allow
Zone bit	0x0047
Data bit	Function
Bit 7-0	Min length of Matrix 2 of 5 0x00-0xFF: 0-255Byte
Zone bit	0x0048
Data bit	Function
Bit 7-0	Max length of Matrix 2 of 5 0x00-0xFF: 0-255Byte
Zone bit	0x0049
Data bit	Function
Bit 7-1	Keep
Bit0	Read Code11 0: Forbid      1: Allow
Zone bit	0x004A

Data bit	Function
Bit 7-0	Min length of Code11 0x00-0xFF: 0-255Byte
Zone bit	<i>0x004B</i>
Data bit	Function
Bit 7-0	Max length of Code11 0x00-0xFF: 0-255Byte
Zone bit	<i>0x004C</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read MSI 0: Forbid      1: Allow
Zone bit	<i>0x004D</i>
Data bit	Function
Bit 7-0	Min length of MSI 0x00-0xFF: 0-255Byte
Zone bit	<i>0x004E</i>
Data bit	Function
Bit 7-0	Max length of MSI 0x00-0xFF: 0-255Byte
Zone bit	<i>0x004F</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read RSS-14 0: Forbid      1: Allow
Zone bit	<i>0x0050</i>
Data bit	Function
Bit 7-1	Keep

Bit0	Read limited RSS 0: Forbid      1: Allow
Zone bit	<i>0x0051</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read Expended RSS 0: Forbid      1: Allow
Zone bit	<i>0x0052</i>
Data bit	Function
Bit 7-0	Min length of RSS 0x00-0xFF: 0-255Byte
Zone bit	<i>0x0053</i>
Data bit	Function
Bit 7-0	Max length of RSS 0x00-0xFF: 0-255Byte
Zone bit	<i>0x0054</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read DM 0: Forbid      1: Allow
Zone bit	<i>0x0055</i>
Data bit	Function
Bit 7-1	Keep
Bit0	Read PDF417 0: Forbid      1: Allow
Zone bit	<i>0x0060</i>
Data bit	Function
Bit 7	Keep







	0x00: write 0 to wake up module
Zone bit	0x00E0 (Only read Zone bit)
Data bit	Function
Bit 7-0	Product model 0x05: GM65 Module
Zone bit	0x00E1 (Only read Zone bit)
Data bit	Function
Bit 7-0	Hardware Version 0x64: V1.00 0x6E: V1.10 0x78: V1.20 0x82: V1.30 0x8C: V1.40 .....
Zone bit	0x00E2 (Only read Zone bit)
Data bit	Function
Bit 7-0	Software Version 0x64: V1.00 0x6E: V1.10 0x78: V1.20 0x82: V1.30 0x8C: V1.40 .....
Zone bit	0x00E3 (Only read Zone bit)
Data bit	Function
Bit 7-0	Year of software (Add 2000) 0x0F: 2015 0x10: 2016 0x11: 2017 .....
Zone bit	0x00E4 (Only read Zone bit)
Data bit	Function
Bit 7-0	Software month 0x09: September 0x0A: Oct.

	0x0B: Nov. .....
Zone bit	0x00E5 (Only read Zone bit)
Data bit	Function
Bit 7-0	Software date 0x09: 9 0x0A: 10 0x0B: 11 .....

## 9 Appendix A: Common serial port instruction

Function	Instruction
Baud rate to 9600	7E 00 08 01 00 D9 D3 20 38
Save settlements to EEPROM	7E 00 09 01 00 00 DE C8
Find baud rate	7E 00 07 01 00 2A 02 D8 0F

Module will return following info after mainframe send serial port instruction - find baud rate

Return information	Baud rate
02 00 00 02 C4 09 SS SS	1200
02 00 00 02 71 02 SS SS	4800
02 00 00 02 39 01 SS SS	9600
02 00 00 02 D0 00 SS SS	14400
02 00 00 02 9C 00 SS SS	19200
02 00 00 02 4E 00 SS SS	38400
02 00 00 02 34 00 SS SS	57600

PS: SS SS= check value。

## 10 Appendix B: Code ID List

Type of Bar Code	Corresponding character	Zone bit address
EAN-13	d	0x91
EAN-8	d	0x92
UPC-A	c	0x93
UPC-E0	c	0x94
UPC-E1	c	0x95
Code 128	j	0x96
Code 39	b	0x97
Code 93	i	0x98
Codabar	a	0x99
Interleaved 2 of 5	e	0x9A
Industrial 2 of 5	D	0x9B
Matrix 2 of 5	v	0x9C
Code 11	H	0x9D
MSI-Plessey	m	0x9E
GS1 Databar(RSS-14)	R	0x9F
GS1 Databar Limited(RSS)	R	0xA0
GS1 Databar Expanded(RSS)	R	0xA1
QR Code	Q	0xA2
Data Matrix	u	0xA3
PDF 417	r	0xA4

## 11 Appendix C: ASCII

Base 16 Hexadecimal	Decimalism	Character
00	0	NUL
01	1	SOH
02	2	STX
03	3	ETX
04	4	EOT
05	5	ENQ
06	6	ACK
07	7	BEL
08	8	BS
09	9	HT
0a	10	LF
0b	11	VT
0c	12	FF
0d	13	CR
0e	14	SO
0f	15	SI
10	16	DLE
11	17	DC1
12	18	DC2
13	19	DC3
14	20	DC4
15	21	NAK
16	22	SYN
17	23	ETB
18	24	CAN
19	25	EM
1a	26	SUB
1b	27	ESC

Base 16	Decimalism	Character
1c	28	FS
1d	29	GS
1e	30	RS
1f	31	US
20	32	SP
21	33	!
22	34	"
23	35	#
24	36	\$
25	37	%
26	38	&
27	39	`
28	40	(
29	41	)
2a	42	*
2b	43	+
2c	44	,
2d	45	-
2e	46	.
2f	47	/
30	48	0
31	49	1
32	50	2
33	51	3
34	52	4
35	53	5
36	54	6
37	55	7
38	56	8
39	57	9
3a	58	:

Base 16	Decimalism	Character
3b	59	;
3c	60	<
3d	61	=
3e	62	>
3f	63	?
40	64	@
41	65	A
42	66	B
43	67	C
44	68	D
45	69	E
46	70	F
47	71	G
48	72	H
49	73	I
4a	74	J
4b	75	K
4c	76	L
4d	77	M
4e	78	N
4f	79	O
50	80	P
51	81	Q
52	82	R
53	83	S
54	84	T
55	85	U
56	86	V
57	87	W
58	88	X
59	89	Y



Base 16	Decimalism	Character
5a	90	Z
5b	91	[
5c	92	\
5d	93	]
5e	94	^
5f	95	_
60	96	'
61	97	a
62	98	b
63	99	c
64	100	d
65	101	e
66	102	f
67	103	g
68	104	h
69	105	i
6a	106	j
6b	107	k
6c	108	l
6d	109	m
6e	110	n
6f	111	o
70	112	p
71	113	q
72	114	r
73	115	s
74	116	t
75	117	u
76	118	v
77	119	w
78	120	x

---

Base 16	Decimalism	Character
79	121	y
7a	122	z
7b	123	{
7c	124	
7d	125	}
7e	126	~
7f	127	DEL

## 12 Appendix D: data code

0 ~ 9



0



1



2



3



4



5



6



7



8



9

A - F



A



B



C



D



E



F