



# RT830i-B USER GUIDE

(VERSION 1.0)

2022.07

# catalog

Catalog .....	0
1.Product introduction .....	3
1.1 Brief introduction .....	3
1.2 Features: .....	3
1.3 Range .....	4
1.4 Characters .....	5
1.5 Engine appearance .....	6
1.5.1 Front view .....	6
1.5.2 Top view .....	6
1.5.3 Bottom view .....	错误! 未定义书签。
1.6 Interface .....	7
1.6.1 Connector .....	7
1.6.2 Ribbon Cable .....	7
1.6.3 Pin definition .....	7
1.7 Design notice .....	8
1.7.1 Components construction .....	8
1.7.2 Temperature .....	8
1.7.3 Window range .....	9
2、 Configuration protocol .....	9
2.1 Brief introduction .....	9
2.2 Using configuration command .....	10
2.2.1 Configuration (For serial port) .....	10
2.2.2 Communication protocol .....	10
2.2.3 Configuration and response protocol .....	11
2.2.4 Handshake .....	11
2.2.5 Get information .....	12
2.2.6 Start or stop decoding .....	17
2.2.7 Set operation mode .....	17

---

2.2.8 Set configuration mode .....	18
2.2.9 Set message-control mode .....	20
2.2.10 Settings operation .....	21
3. CRC source code .....	22
Technical support .....	23

# 1. Product introduction

## 1.1 Brief introduction

RT830i-B is designed for near-distance barcode reading. This module has a large field of view, which is outstanding at near reading on-screen or paper bar codes even when the screen is lower brightness or poor quality printed. For example, the Min. readable distance is 1.5cm for the 5mm DM code.

It supports reading most types of common 1d&2d bar codes, whatever is read from paper or from the phone screen. Builds in USB and TTL-232 interfaces, and the RS232 interface is optional. So you can connect it with your terminals via USB directly (no need for any other kit board) or 12pins TTL interface.

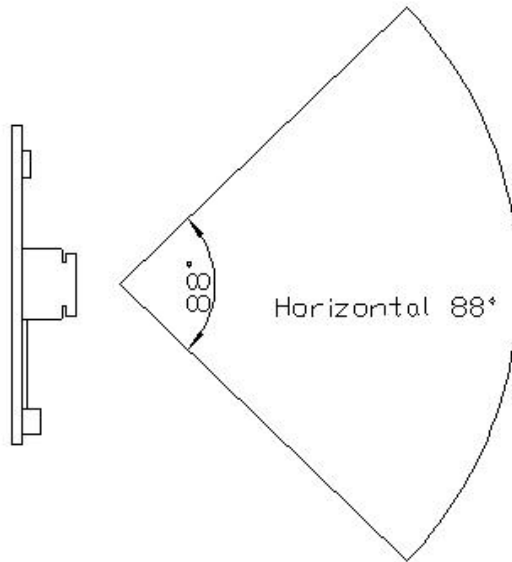
RT830i-B is a fixed-mount module for embedded use. With a thin design, it can be embedded in ATMs, POS terminals, turnstiles, lockers, laptops, self-service terminals, notebooks, and also medical devices/ instruments, etc.

RTscan provides a complete and detailed user guide and SDK documents for your integration work.

## 1.2 Features:

- Suitable for near-distance barcodes reading
- Ultra-thin design for embedded use
- Support reading most types of 1D&2D bar codes from screen and paper
- Reading well even in a dim environment
- Windows, Linux, Android, Arduino, Raspberry PI
- USB/ TTL-232 interface, and the RS232 is optional
- Application: ATMs, POS terminals, turnstiles, lockers, laptops, notebooks, and also medical device, etc.

### 1.3 Range



Code39 60-160mm (2\*4cm)  
Code128 50-160mm(2\*4cm)  
QR code 30-70mm (5mm)  
PDF417 40-190mm (3\*5cm)  
PDF417 30-170mm(2\*4cm)  
Data Matrix 15-90mm (5mm)

## 1.4 Characters

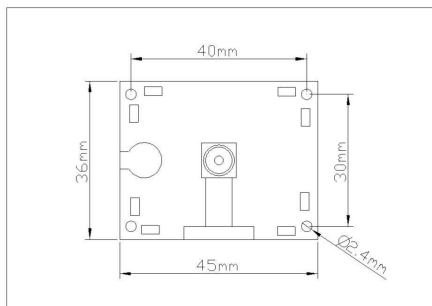
<b>Performance</b>	Image Sensor	640 pixels (H)x 480 pixels (V)		
	Illumination	Illumination: 5000K LED		
	Symbologies	2D	PDF417, Data Matrix, QR Code, etc.	
		1D	UPC-A,UPC-C, EAN-8,EAN-13, Code 128, Code 39, Code 93, Codabar, Interleaved 2 of 5, etc.	
	Resolution	≥7mil		
	Typical Depth of Field	Code39	60-160mm (2*4cm)	
		Code128	50-160mm(2*4cm)	
		QR code	30-70mm (5mm)	
		PDF417	40-190mm (3*5cm)	
		PDF417	30-170mm(2*4cm)	
	Data Matri x	15-90mm (5mm)		
	Scan Mode	Sense mode, continuous mode, command mode		
	Scan Angle	Roll: 360°, Pitch: ±60°, Skew:±40°		
	Field of View	Horizontal 88°, Vertical 57°		
Motion Tolerances	>0.3m/s			
Print Contrast	30% minimum reflective difference			
FPS	30FPS			
<b>Mechanical/ Electric</b>	Interface	TTL-232, USB-HID, USB-VCOM, RS232 (optional)		
	Dimensions	45mm x 36mm x 12mm		
	Weight	13g		
	Notification	Beep		
	Operating Voltage	FPC PORT 3.3V DC MX1.25 PORT 5.0V DC		
	Current	Operating: 140mA. Standby: 40mA. Peak: 240mA.		
<b>Environmental</b>	Operating Temperature	-20°C to 60°C		
	Storage Temperature	-40°C to 80°C		
	Humidity	5% to 95% (non-condensing)		
	Drop/Shock	Withstands 10Gs 0.06"double amp		
	Ambient Light Immunity	0-100,000 Lux.		

## 1.5 Engine appearance

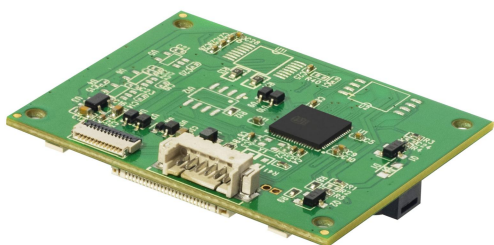
### 1.5.1 Front view



### 1.5.2 Top view



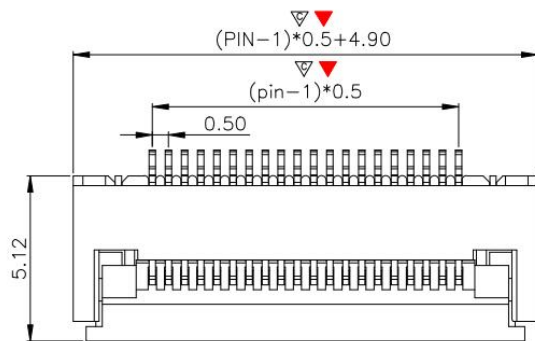
### 1.5.3 Bottom view



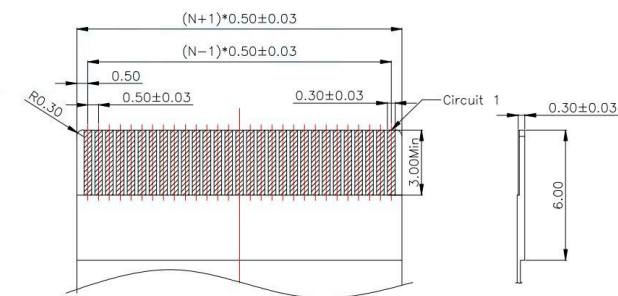
## 1.6 Interface

Connector and PIN out definition, as follows:

### 1.6.1 Connector



### 1.6.2 Ribbon Cable



### 1.6.3 Pin definition

PIN#	Signal Name	I/O	Function
PIN 1	—	—	—
PIN 2	VCC_5V	—	Power supply
PIN 3	GND	—	GND & Signal ground
PIN 4	UART1_RX	Input	RS232 receiving



PIN 5	UART1_TX	Output	RS232 transmission
PIN 6	USB-	—	USB- signal
PIN 7	USB+	—	USB+ signal
PIN 8	—	—	—
PIN 9	BEEP_CTL	Output	Passive buzzer output signal, Output PWM pulse waveform when working. idle output low level.
PIN10	DECODE_LED	Output	Indicator output signal, idle output high level. Working output low level
PIN11	—	—	—
PIN12	TRIG	Input	Trigger key input signal, keep the low level for more than 5ms to trigger code reading

Note: please remember your device's TXD should be connected to RXD of RT830i-B, and RXD to TXD.

You may also refer to the ***Interface and Integration of RT830i-B.pdf*** to get some tips for integration via the TTL interface or via USB /RS232 with the emulation kit.

## 1.7 Design notice

### 1.7.1 Components construction

Leave enough space for RT830i-B and the cable, which make RT830i-B and the cable cannot be pressed or destroyed by any other components.

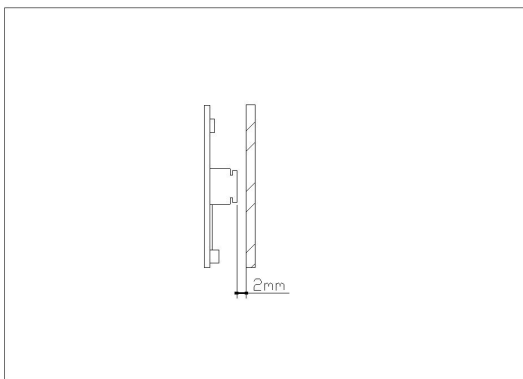
### 1.7.2 Temperature

Under the continuous working, RT830i-B's temperature will get higher, additional cooling measure is recommended.

### 1.7.3 Window range

Some rules should be followed when using the engine:

- ✓ The engine's camera and LED should not be blocked.
- ✓ High transparent material should be used for better picture caption.
- ✓ window and the camera should be parallel, the distance between the window and the engine should be less than 2mm.



No exceed 2mm

## 2、 Configuration protocol

### 2.1 Brief introduction

Two method can be used to configure the engine.

- ✓ Using configuration command, below 2.2 with introduction, and you can also refer to the [RT830i-B API Instruction.pdf](#), [RT830i-B-Protocol-Instruction.pdf](#) and [Protocol Map.xls](#) for more information.
- ✓ Using setting codes which provided by : [Configuration Tool.exe](#) , to learn how to use the Configuration Tool.exe, please refer to the: [RT830i-B-Configuration-tool-user-guide.pdf](#)

## 2.2 Using configuration command

### 2.2.1 Configuration (For serial port)

The engine uses the serial port as the communication interface, the default parameters is: 115200 baud rate, 8 data bits, no parity and stop bits.

### 2.2.2 Communication protocol

Host send command, and the engine response. According to the spec, developers can set or read the parameters of the engine. Note: the command and response follows the little Endian format.

#### 2.2.2.1 Connect and Disconnect

Only for testing, to check whether engine can work properly.

#### 2.2.2.2 Connect

This command tests the communication between host and engine.

Type	data	length	remark
Command	0x52(R)	1	Test if engine ready
response	0x59(Y)	1	OK

#### 2.2.2.3 Disconnect

Type	data	length	remark
Command	0x4443(DC)	2	disconnect
response	No data		

## 2.2.3 Configuration and response protocol

Host can set or query the engine's parameter. host send command, and the engine response. According to the different command, host and engine complete different operation.

### 2.2.3.1 Command and response format

**Command format:** HEAD + LENTH +DATA +CRC

HEAD	LENTH	DATA	CRC
3bytes	2 bytes	Variable length	2bytes

HEAD: three fixed ASCII char 'C', 'M', 'D'

LENTH: the length of the command,2 bytes;

DATA: Command body, including instruction code and corresponding data. The instruction is a five-char string. Command can be made by many sequential instructions. Only sending one instruction code is suggested.

CRC: CRC data of command.

**Response format:** HEAD + LENTH +DATA +CRC

HEAD	LENTH	DATA	CRC
3bytes	2 bytes	Variable length	2bytes

HEAD: three fixed ASCII char 'R', 'S', 'P'

LENTH: response length, 2 bytes;

DATA: response body, including instruction code and corresponding data.

CRC: CRC of the command.

## 2.2.4 Handshake

Before sending command, host should send handshake first, in order to test the engine whether works normally. When received the right response, the host can then

send instruction.

handshake	data	length(bytes)	remark
command	0x59(Y)	1	
response	0xAABB0AABB	4	

## 2.2.5 Get information

### 1) description:

host send this instruction to engine to get all kinds of engine's information ,including software and hardware information, operation mode, 1D/2D status.

### 2) format:

**command:** head+length+instruction code +CRC

**response:** "RSP" + length + data(information )+ CRC

### instruction code:

instruction code (ASCII)	description
85505	Get hardware information
88306	Get software information
88501	Get operation mode
88603	Get 1D symbology status
88705	Get 2D symbology status
88807	Get the image stored in engine's memory

### 2.2.5.1 Get hardware information

Hardware information including hardware version、type、serial number、product date can be acquired through this command.

**CMD:** "CMD"+ 0x00 + 0x05 + "85505"+ CRC

**RSP:** "RSP"+ Length of hardware information (2bytes) (0x00 + 0x37)+ hardware information (55 ↑ bytes) + CRC

The structure body of hardware information is a 55-byte data, as follows:

```
Type def struct Hardware Info {
    unsigned char hardware Ver[10]; //hardware version
    unsigned char hardware Type[10]; //hardware type
    unsigned char specs[10]; //hardware specification
    unsigned char serial Num[10]; //serial number
    unsigned char product Date[15]; //product date
} Hardware Info;
```

### 2.2.5.2 get software information

Each symbology has its own version, there is no connection between these version. The version format is:xx.xx.xx with 21 bytes,the first 13 byte is the symbology, zero will be filled if less than 13 bytes. The following 8 bytes demonstrate the symbology version. Between each version, there will be two-byte data "\r\n".

**CMD:**"CMD"+ 0x00 + 0x05 + "88306"+ CRC

**RSP:** "RSP"+ Length of software information (2bytes) + software information (variable length) + CRC

**example:** "QR Code: 01.02.24\r\nPDF417: 01.01.36\r\nDataMatrix:01.03.18 \r\nCS Code: 01.01.10" , means that QR version is 01.02.24,PDF417 version01.01.36、 Data Matrix version is01.03.18、 CS 码 version is 1.01.10。

### 2.2.5.3 Get operation mode

**CMD:** "CMD"+0x00+0x05+"88501"+CRC

**RSP:** "RSP"+0x00+0x0F+current operation mode (15byte, each mode has 5 bytes)  
+CRC

Operation mode includes decoding mode, supplementing light mode and location light mode. Each mode has its own command, and the sequence is not fixed, the operation mode can be determined by the command.

The commands are as follows:

Instruction code (ASCII)	description
Trigger mode	
20301	Key pressed
20302	Auto
20303	continuous (key pressed trigger first)
20304	Single auto
supplementing light (Red Led)	
20101	flash
20102	Always on
20103	Always off
Location light (Green led)	
20201	flash
20202	Always on
20203	Always off

**example:** "203012010120201", "20301" means the current decoding mode is

key-press triggered, "20101" means supplementing light mode is flash mode, "20201" means the location light mode is flash mode.

#### 2.2.5.4 Get 1D switch status

**CMD:** "CMD"+ 0x00 + 0x05 + "88603"+ CRC

**RSP:** "RSP"+ 0x00 + 0x14 + 1D switch status (20bytes) + CRC

1D barcode includes 4 symbology: Code128、Code39、EAN-8、EAN-13, Each status has 5 bytes, so there are 20 bytes. The status sequence is not fixed, the 1D switch mode can be determined by the command.

The status are as follows:

table 0- 3 instruction code

Instruction code (ASCII)	description
1D	
10111	Enable code128
10110	Disable code128
10411	Enable code39
10410	Disable code39
10211	Enable EAN8
10210	Disable EAN8
10311	Enable EAN13
10310	Disable EAN13

**example:** "10111102111031110411", "10111" means enable code128, "10211" means enable EAN8, "10311" means enable EAN13, "10411" means enable code39.



### 2.2.5.5 : Get 2D switch status

**CMD:** "CMD"+ 0x00 + 0x05 + "88705"+ CRC

**RSP:** "RSP"+ 0x00 + 0x14 + 1D switch status (20bytes) +CRC

1D barcode includes QR ,CS,PDF417,Data Matrix, Each status has 5 bytes, so there are 20 bytes. The status sequence is not fixed, the 2D switch mode can be determined by the command.

The status are as follows:

2D	
10611	Enable QR
10610	Disable QR
10811	Enable CS
10810	Disable CS
10511	Enable PDF417
10510	Disable PDF417
10711	Enable Data Matrix
10710	Disable Data Matrix

**example:** "10611107111081110511", "10611"means enable QR, "10711" means enable Data Matrix,"10811" means enable CS,10511"means enable PDF417.

### 2.2.5.6: Get image stored in engine

**CMD:** "CMD"+0x00+0x05+"88807"+CRC

**RSP:** image data (length is 752\*480)

## 2.2.6 Start or stop decoding

- 1) **description:** host send this instruction to engine, which make the engine to start or stop decoding
- 2) **format:**  
 command: head+length+instruction code +CRC  
 response: 0x59(Y)
- 3) **instruction code:**

instruction code (ASCII)	description
89910	Stop decoding
89811	Begin decoding

- 4) **example:** stop decoding  
 Step1: host send command: 0x59  
 Step2: received response: 0xAABB0AABB  
 Step3: host send command: "CMD"+0x0005+"89910"+(CRC)  
 Step4: received response: 0x59

## 2.2.7 Set operation mode

- 1) **description:** host send instruction code to set the operation mode.
- 2) **format:**  
 command: head +length+ instruction code +CRC  
 response: 0x59(Y)
- 3) **instruction code:**

Instruction code (ASCII)	description
<b>Trigger mode</b>	
20301	Key pressed
20302	Auto
20303	continuous (key pressed trigger first)
20304	Single auto
<b>Red Led</b>	
20101	flash
20102	Always on
20103	Always off
<b>Green led</b>	
20201	flash
20202	Always on
20203	Always off

4) example: set green led off

Step1: host send command: 0x59

Step2: received response: 0xAABB0AABB

Step3: host send command: "CMD"+0x0005+"20103"+(CRC)

Step4: received response: 0x59

## 2.2.8 Set configuration mode

1) **description:** host send instruction code to set the configuration mode.

2) **format:**

command: head +length+ instruction code +CRC

response: 0x59(Y)

**3) instruction code:**

Instruction code (ASCII)	description
<b>1D</b>	
10111	Enable code128
10110	Disable code128
10411	Enable code39
10410	Disable code39
10211	Enable EAN8
10210	Disable EAN8
10311	Enable EAN13
10310	Disable EAN13
<b>2D</b>	
10611	Enable QR
10610	Disable QR
10620	QR single code
10621	QR double code
10622	QR single or double code
10811	Enable CS
10810	Disable CS
10820	CS single code
10821	CS double code

10822	CS single or double code
10511	Enable PDF417
10510	Disable PDF417
10520	PDF417 single code
10521	PDF417 double code
10522	PDF417 single or double code
10711	Enable Data Matrix
10710	Disable Data Matrix
10720	Data Matrix single code
10721	Data Matrix double code
10722	Data Matrix single or double code

4) **example:** Disable QR

Step1: host send command: 0x59

Step2: received response: 0xAABBOAABB

Step3: host send command: "CMD"+0x0005+" 10610"+(CRC)

Step4: received response: 0x59

## 2.2.9 Set message-control mode

1) **description:** host send instruction code to set message-control mode.

2) **format:**

command: head +length+ instruction code +CRC

response: 0x59(Y) or head + length +message-control code +CRC

3) **instruction code:**

instruction code (ASCII)	description

86101	enter message control mode
86110	exit message control mode
86201	Begin decoding
86202	response, format:  result flag(1byte)+decoding data  result flag: 1success 2timeout 3fail

4) **example:** exit message control mode

Step1: host send command: 0x59

Step2: received response: 0xAABB0AABB

Step3: host send command: "CMD"+0x0005+" 20302"+(CRC)

Step4: received response: 0x59

## 2.2.10 Settings operation

1) **description:** host send instruction code to save or restore engine's settings.

2) **format:**

command: head +length(0x05,2bytes)+ instruction code +CRC

response: 0x59(Y)

3) **instruction code:**

instruction code (ASCII)	description
89101	save user setting
89203	load user default settings
89305	restore default settings

### 3. CRC source code

```
unsigned short cal_crc2(unsigned char *ptr, unsigned long len)
{
    unsigned char i;
    unsigned int crc=0;
    while(len--!=0)
    {
        for(i=0x80; i!=0; i/=2)
        {
            if((crc&0x8000)!=0)
            {
                crc*=2;
                crc^=0x18005;
            }
            else
            {
                crc*=2;
            }
            if((*ptr&i)!=0) crc^=0x18005;
        }
        ptr++;
    }
    return crc; }
```

## Technical support

If need any technical support, feel free to contact our technical support department:

Support Department

.....



RTSCAN TECHNOLOGY LIMITED

Add: 553-555, Dongming Building, Minzhi Street, Longhua District, Shenzhen 518109, China

Tel: +86 755-23762089

E-mail: [support@rtscan.net](mailto:support@rtscan.net)