

# Bluetooth Module

## Index

### BLUETOOTH

|   |           |
|---|-----------|
| <b>MODULE.....</b>                        | <b>1</b>  |
| <b>OVERVIEW .....</b>                     | <b>2</b>  |
| <b>LICENSE .....</b>                      | <b>2</b>  |
| <b>SPECIFICATIONS .....</b>               | <b>3</b>  |
| HARDWARE FEATURES.....                    | 3         |
| SOFTWARE FEATURES.....                    | 3         |
| <b>SOFTWARE</b>                           |           |
| <b>INSTRUCTION .....</b>                  | <b>4</b>  |
| WORKING SKETCH MAP.....                   | 4         |
| FLOWCHART .....                           | 4         |
| COMMANDS TO CHANGE DEFAULT SETTINGS ..... | 5         |
| COMMANDS FOR NORMAL OPERATION:.....       | 6         |
| <b>PINOUT .....</b>                       | <b>8</b>  |
| <b>PIN FUNCTION.....</b>                  | <b>9</b>  |
| <b>REVISION HISTORY.....</b>              | <b>11</b> |

# Overview



The module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

## License



Source files and documents are licensed under a Creative Commons Attribution 3.0 Unported License.

# Specifications

## Hardware features

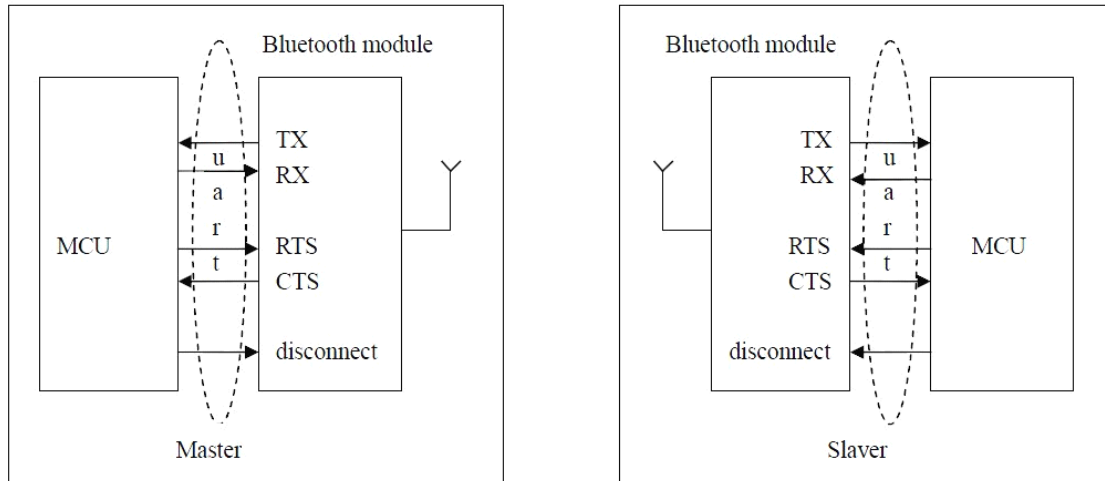
- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector

## Software features

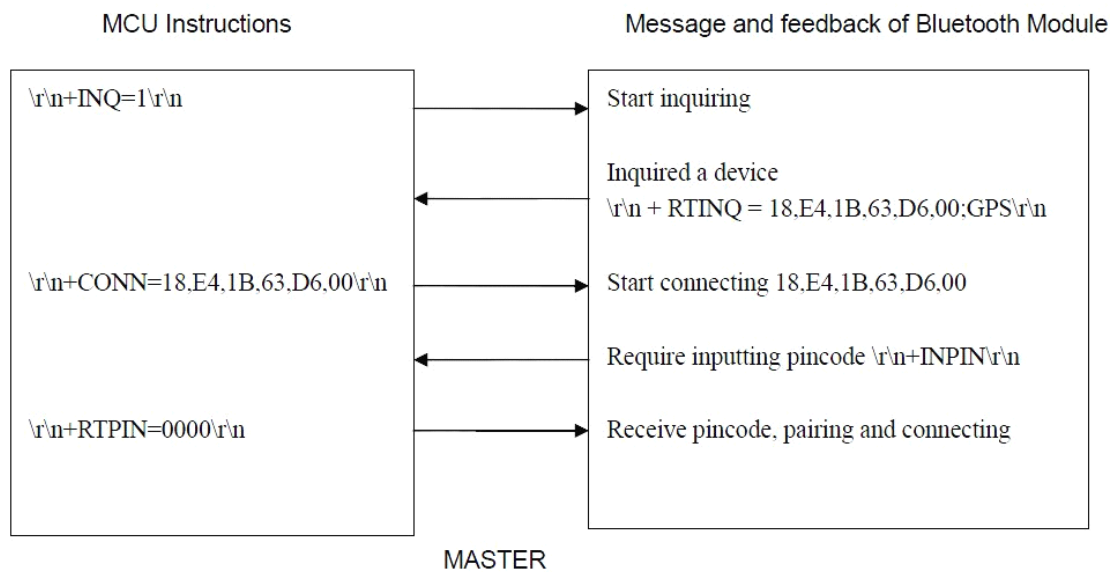
- Default Baud rate: **9600**, Data bits:8, Stop bit:1,Parity:No parity, Data control: has.  
Supported baud rate:  
9600,19200,38400,57600,115200,230400,460800.
- Use CTS and RTS to control data stream.
- Given a rising pulse in PIO0, device will be disconnected.
- Status instruction port PIO1: low-disconnected, high-connected;
- PIO10 and PIO11 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s.
- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing PINCODE:"**0000**" as default
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

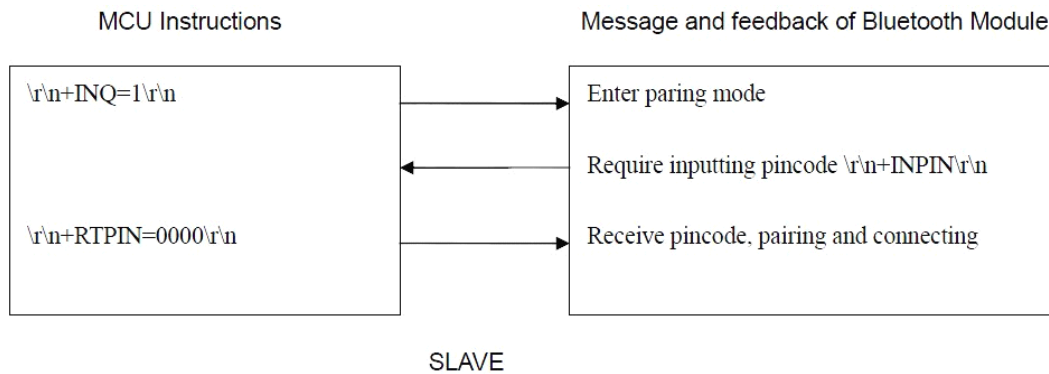
# Software Instruction

## Working Sketch Map



## Flowchart





Attention: To use Uart port, you may need to change voltage level by RS232 or other IC.

Attention: To use Uart port, you may need to change voltage level by RS232 or other IC.

## Commands to change default settings

### 1. Set working MODE

\r\n+STWMOD=0\r\n Set device working as client (slave), Save and Rest

\r\n+STWMOD=1\r\n Set device working as server (master), Save and Rest

Note: \r\n is needed, and the value of which is 0x0D 0x0A in Hex, meaning return and next row,

### 2. Set BAUDRATE

\r\n+STBD=115200\r\n Set baudrate 115200, Save and Rest

Supported baudrate: 9600, 19200,38400,57600,115200,230400,460800.

### 3. Set Device NAME

\r\n+STNA=abcdefg Set device name "abcdefg", Save and Rest

### 4. Auto-connect the last paired device on power

\r\n+STAUTO=0\r\n Forbidden, Save and Rest

\r\n+STAUTO=1\r\n Permit, Save and Rest

## **5. Permit Paired device to connect me**

\r\n+STOAUT=0\r\n Forbidden, Save and Rest

\r\n+STOAUT=1\r\n Permit, Save and Rest

## **6. Set PINCODE**

\r\n +STPIN=222\r\n Set pincode “2222”, Save and Rest

## **7. Delete PINCODE (input PINCODE by MCU)**

\r\n+DLPIN\r\n Delete pincode, Save and Rest

## **8. Read local ADDRESS CODE**

\r\n+RTADDR\r\n Return address of the device

## **9. Auto-reconnecting when master device is beyond the valid range (slave device will auto-reconnect in 30 min as it is beyond the valid range)**

\r\n+LOSSRECONN=0\r\n Forbidden auto-reconnecting

\r\n+LOSSRECONN=1\r\n Permit auto-reconnecting

## **Commands for Normal Operation:**

### **1. Inquire**

a) Master

\r\n+INQ=0\r\n Stop Inquiring

\r\n+INQ=1\r\n Begin/Restart Inquiring

b) Slave

\r\n+INQ=0\r\n Disable been inquired

\r\n+INQ=1\r\n Enable been inquired

## 2. Bluetooth module returns inquiring result

\r\n+RTINQ=aa,bb,cc,dd,ee,ff;name\r\n A serial Bluetooth device with the address "aa,bb,cc,dd,ee,ff" and the name "name" is inquired

## 3. Connect device

\r\n+CONN=aa,bb,cc,dd,ee,ff\r\n Connect to a device with address of "aa,bb,cc,dd,ee,ff"

## 4. Bluetooth module requests inputting PINCODE

\r\n+INPIN\r\n

## 5. Input PINCODE

\r\n+RTPIN=code\r\n

Example: RTPIN=0000 Input PINCODE which is four zero

## 6. Disconnect device

Pulling PIO0 high will disconnect current working Bluetooth device.

## 7. Return status

\r\n+RTSTA:xx\r\n

xx status:

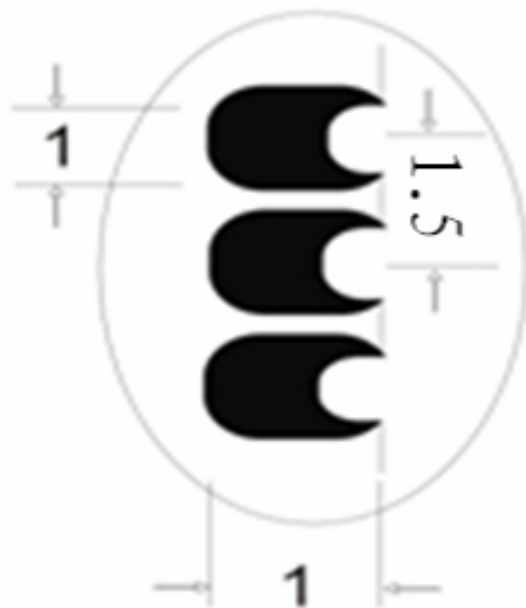
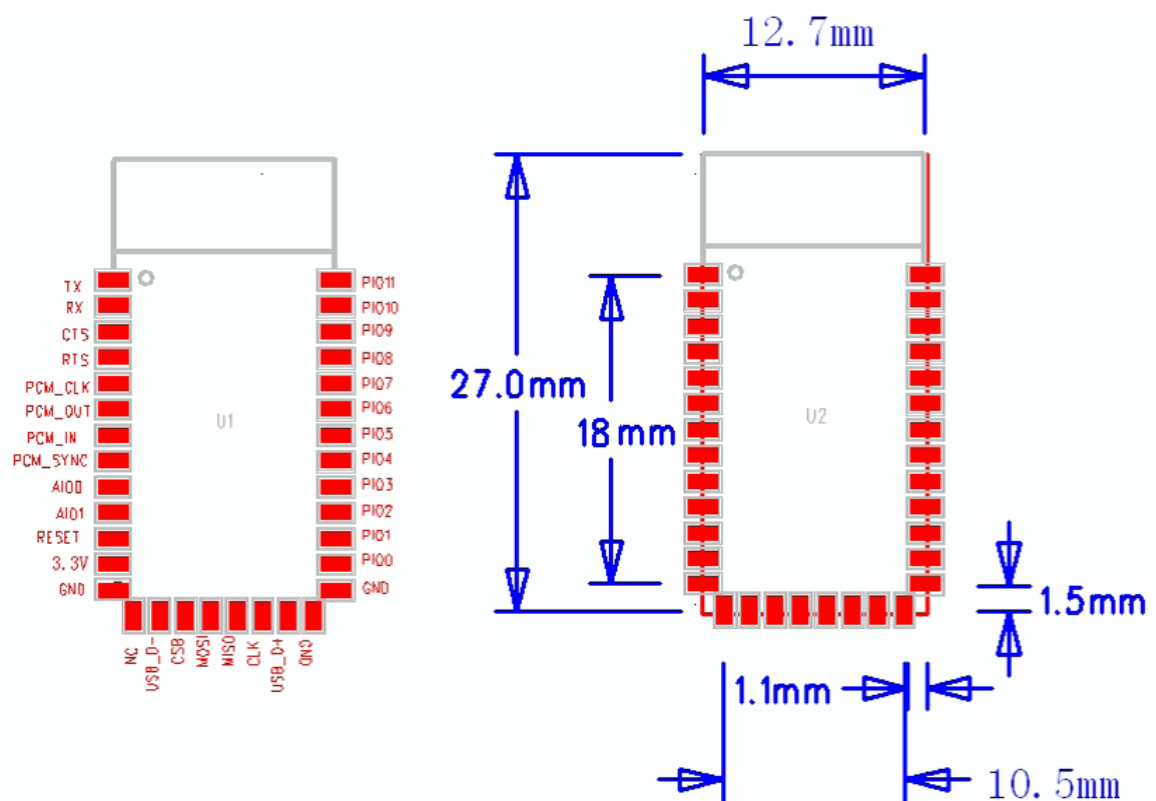
0: Initializing; 1: Ready; 2: Inquiring; 3: Connecting; 4: Connected;

**(Note: This is not a command, but the information returning from the module)**

**More information about how to setup connections between Bluetooth in emartee forum:**

<http://www.emartee.com/category/Wireless/b>

# Pinout





## Pin Function

| PIN Name | PIN #          | Pad type                | Description  | Note |
|----------|----------------|-------------------------|--|------|
| GND      | 13<br>21<br>22 | VSS                     | Ground pot   |      |
| 3.3 VCC  | 12             | 3.3V                    | Integrated 3.3V (+) supply with On-chip linear regulator output within 3.15-3.3V |      |
| AIO0     | 9              | Bi-Directional          | Programmable input/output line   |      |
| AIO1     | 10             | Bi-Directional          | Programmable input/output line   |      |
| PIO0     | 23             | Bi-Directional<br>RX EN | Programmable input/output line, control output for LNA(if fitted)                |      |
| PIO1     | 24             | Bi-Directional<br>TX EN | Programmable input/output line, control output for PA(if fitted)                 |      |

|       |    |                |                                |  |
|-------|----|----------------|--------------------------------|--|
| PIO2  | 25 | Bi-Directional | Programmable input/output line |  |
| PIO3  | 26 | Bi-Directional | Programmable input/output line |  |
| PIO4  | 27 | Bi-Directional | Programmable input/output line |  |
| PIO5  | 28 | Bi-Directional | Programmable input/output line |  |
| PIO6  | 29 | Bi-Directional | Programmable input/output line |  |
| PIO7  | 30 | Bi-Directional | Programmable input/output line |  |
| PIO8  | 31 | Bi-Directional | Programmable input/output line |  |
| PIO9  | 32 | Bi-Directional | Programmable input/output line |  |
| PIO10 | 33 | Bi-Directional | Programmable input/output line |  |
| PIO11 | 34 | Bi-Directional | Programmable input/output line |  |

|                 |           |  |   |  |
|-----------------|-----------|--|---|--|
| <b>RESETB</b>   | <b>11</b> | CMOS input with weak internal pull-up              | Reset if low.input debounced so must be low for >5MS to cause a reset |  |
| <b>UART_RTS</b> | <b>4</b>  | CMOS output, tri-stable with weak internal pull-up | UART request to send, active low                                      |  |
| <b>UART_CTS</b> | <b>3</b>  | CMOS input with weak internal pull-down            | UART clear to send, active low  |  |
| <b>UART_RX</b>  | <b>2</b>  | CMOS input with weak internal pull-down            | UART Data input   |  |
| <b>UART_TX</b>  | <b>1</b>  | CMOS output, Tri-stable with weak internal pull-up | UART Data output  |  |
| <b>SPI_MOSI</b> | <b>17</b> | CMOS input with weak internal pull-down            | Serial peripheral interface data input                                |  |
| <b>SPI_CSB</b>  | <b>16</b> | CMOS input with weak internal pull-up              | Chip select for serial peripheral interface, active low               |  |
| <b>SPI_CLK</b>  | <b>19</b> | CMOS input with weak internal pull-down            | Serial peripheral interface clock                                     |  |
| <b>SPI_MISO</b> | <b>18</b> | CMOS input with weak internal pull-down            | Serial peripheral interface data Output                               |  |
| <b>USB_-</b>    | <b>15</b> | Bi-Directional                                     |   |  |

|          |    |                |                             |  |
|----------|----|----------------|-----------------------------|--|
| USB_+    | 20 | Bi-Directional |                             |  |
| NC       | 14 |                |                             |  |
| PCM_CLK  | 5  | Bi-Directional | Synchronous PCM data clock  |  |
| PCM_OUT  | 6  | CMOS output    | Synchronous PCM data output |  |
| PCM_IN   | 7  | CMOS Input     | Synchronous PCM data input  |  |
| PCM_SYNC | 8  | Bi-Directional | Synchronous PCM data strobe |  |

## Revision History

| Rev. | Descriptions   | Release date |
|------|--|--------------|
| V1.0 | Initial version  | 2010/01/08   |
| V1.1 | Modify some command, hardware and software features                | 2010/04/07   |
| V1.2 | Update the profile, add the return status,<br>delete ECHO command. | 2010/04/21   |
|      |  |              |