

Operation Guide

CONTENTS

Pin Function	2
Use with BCE-GENTX-X01 Board	3
Use with an Arduino Uno (I ² C interface)	4
Wire Connection	4
Read the Module Version.....	5
Read the Module Status.....	5
Activate the RF Transmission and Send 1-byte Data.....	6
Stop RF Transmission	8

Pin Function

The BCM-2102-X03 module has 9 pins, as below.

Pin#	Name	Function
1	GND	Power Ground
2	VDD	Power Supply, with a voltage range of 2.2V - 3.6V.
3	VDDRF	Power Supply for RF circuit This power supply has a voltage range of 2.2V - 3.6V.
4	KEY3	Input pin, active low. This pin is used to trigger the module to transmit a signal. Note that this pin is connected to pin 8.
5	RX/SCL	UART / I ² C interface signal RX: UART data input if the UART interface is selected. SCL: I ² C clock signal if the I ² C interface is selected.
6	TX/SDA	UART / I ² C interface signal TX: UART data output if the UART interface is selected. SDA: I ² C data signal if the I ² C interface is selected.
7	UART/I ² C	Interface select pin This pin is used to select the communication interface. If this pin is unconnected, the default interface is I ² C interface. Low: UART interface is selected High: I ² C interface is selected
8	KEY3	Input pin, active low. This pin is used to trigger the module to transmit a signal. Note that this pin is connected to pin 4.
9	NC	No connection

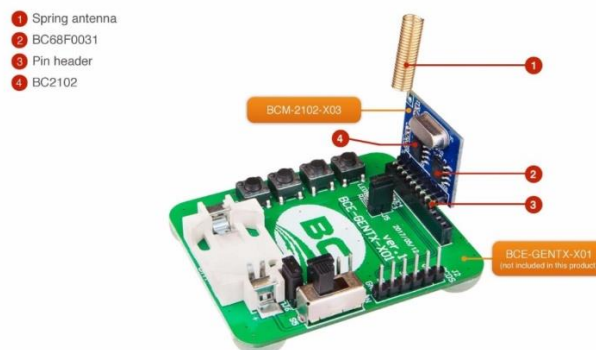
Pin 2(VDD) is the power supply to the module, pin 3(VDDRF) is the power supply to the RF circuit. In general, we usually connect pin 2 and pin 3 together so that they have the same power supply.

Our Arduino example code works on the Arduino Uno R3 and uses I²C interface to control this module. Only pin1, 2, 3, 5, and 6 of this module are used, other pins are left unconnected.

Use with BCE-GENTX-X01 Board

We have a dedicated evaluation board ([BCE-GENTX-X01](#)) for the transmitter modules. With the BCE-GENTX-X01 board, you can operate this module easily.

1. Solder the antenna and pin header of the module. Although the module comes with a 90° pin header, a 180° pin header can be used if you prefer. It is recommended to place the antenna perpendicular to the ground and locate it away from metal objects to achieve the optimal performance.
2. After completing the soldering, the module can be connected with an external MCU board. Ensure to turn off the power before a connection. Note that the power voltage level applied to the module and the MCU board should be the same.
3. This module supports both UART and I²C interfaces. Pin 7 is used to select the interface mode: low for the UART mode and high for the I²C mode. If pin 7 is unconnected, the default selection is I²C mode.
4. The receiver module must execute a pairing process to a transmitter module before it can receive data from the transmitter module. The pairing process is implemented that the transmitter module transmits a signal to the receiver module in the pairing mode. This module can transmit a signal by either executing a transmission command or pulling a specific pin to a low level. The KEY3 pin, which is located on pin 4 or pin 8, is the pin used to transmit a signal when it is pulled low. It is convenient to complete the pairing process by pulling the KEY3 pin low.
5. For ease of use both the transmitter module and the corresponding receiver module have the single-key mode to implement the single-key transmission function. Pulling the transmitter module KEY3 pin low will transmit a signal. At this time a low level signal will be output on the LED2 pin which is located on pin 2 of the corresponding receiver module, the BCM-2401-C03. If both the transmitter and receiver modules are connected to the corresponding evaluation boards, the BCE-GENTX-X01 and the BCE-GENRX-X01, respectively, the LED2 on the receiver module evaluation board will be turned on when the KEY2 pin on the transmitter module evaluation board is pulled low.



Use with an Arduino Uno (I²C interface)

This module supports 6 commands to communicate with the host MCU. In the below sections, the Arduino Uno R3 is used as the MCU board to control this module.

Wire Connection

When using the Arduino example code, which can be downloaded from the product page of this module, the wire connection between this module and the Arduino Uno is described below.

module pin#	Function	Wire connection
1	GND, power ground	Connect to GND pin of Arduino Uno
2, 3	VDD, VDDRF, power supply	Connect to 3.3V pin of Arduino Uno
5	SCL, I ² C clock signal	Connect to A5 pin of Arduino Uno
6	SDA, I ² C data signal	Connect to A4 pin of Arduino Uno

In practice, we solder together pin 2 and pin 3 of the module; therefore only 4 wires are used.

The SDA and SCL pins of different Arduino boards may be different. If you are not using Arduino Uno, please check the corresponding pins first.

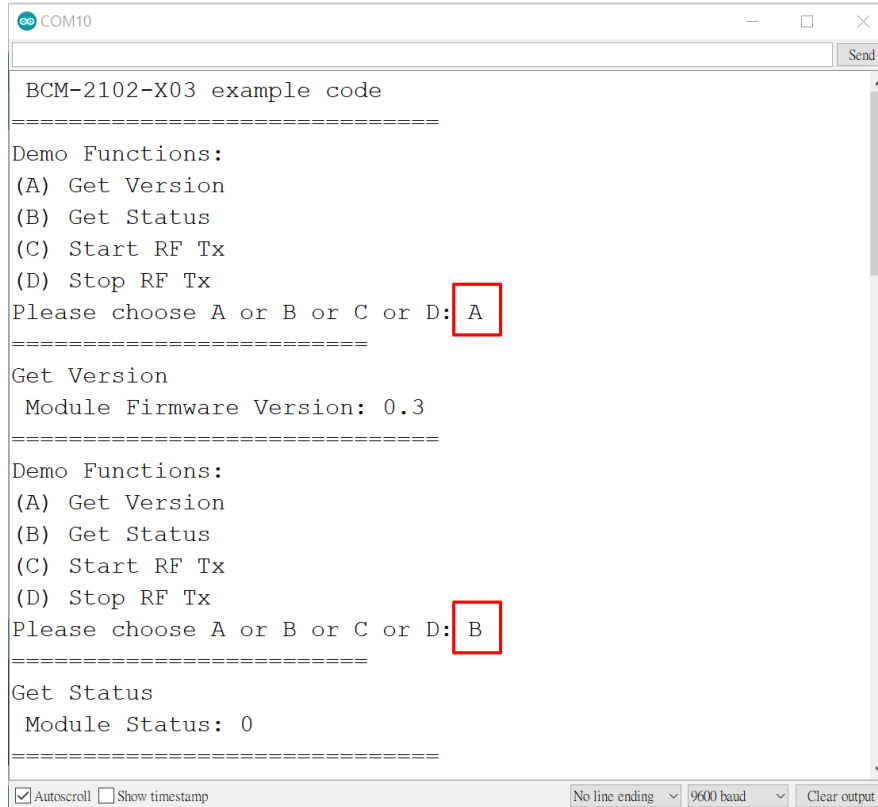
Download and unzip the example code to your sketchbook location, compile and upload the example code to Arduino Uno. The example code demonstrates 4 commands of the module:

- A. GET_VER: read the module version
- B. GET_STATUS: read the module status
- C. START_RF_TX: activate the RF transmission and send 1-byte data
- D. STOP_RF_TX: stop RF transmission

The other two commands are “select RF frequency band” and “set the transmission power”. See the communication protocol document for command details.

Read the Module Version

After uploading the example code, open the serial monitor, the initial display is as below. Input A to execute the Get Version function. The module sends back the version as 0.3.



Read the Module Status

Input B to execute the Read Status function, the module responses the status as 0, which means it is not transmitting data.

Only bit 3 and bit 4 of the module status byte is used. Other bits are not used and read as 0.

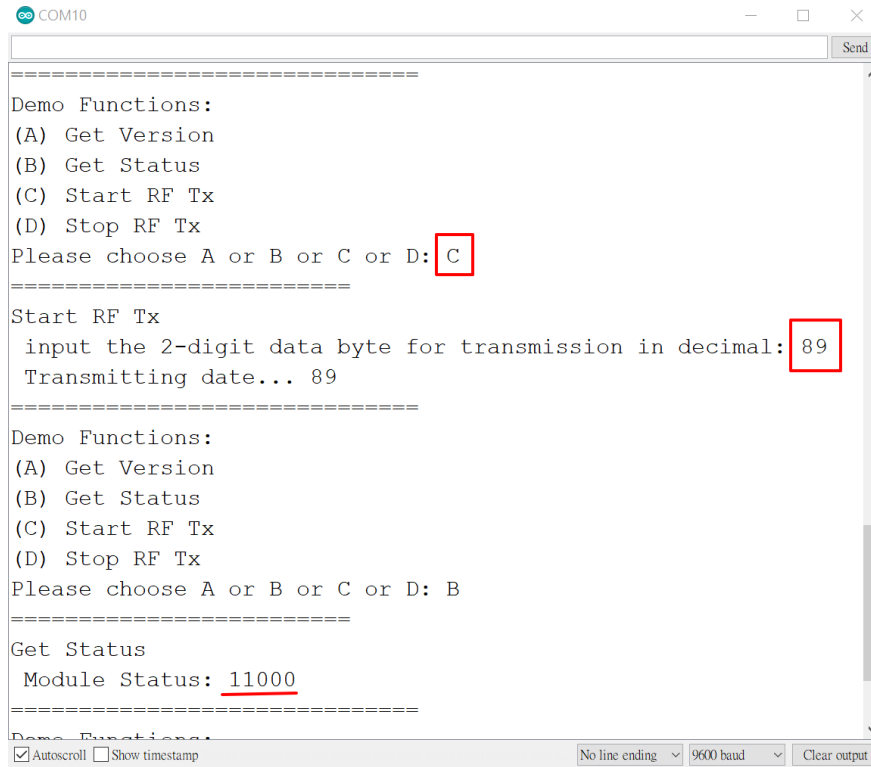
bit3 =1 means the module is transmitting data

bit4 =1 means the module is transmitting data continuously. It will not stop transmitting until it receive the Stop command.

Activate the RF Transmission and Send 1-byte Data

Input C and then input a two-digit decimal value, the value is now transmitted continuously. This example code is designed to transmit continuously and to receive a two-digit decimal value as the data to transmit.

In the below operation, the value 89 is entered as the data.



```
COM10
=====
Demo Functions:
(A) Get Version
(B) Get Status
(C) Start RF Tx
(D) Stop RF Tx
Please choose A or B or C or D: C
=====
Start RF Tx
input the 2-digit data byte for transmission in decimal: 89
Transmitting date... 89
=====
Demo Functions:
(A) Get Version
(B) Get Status
(C) Start RF Tx
(D) Stop RF Tx
Please choose A or B or C or D: B
=====
Get Status
Module Status: 11000
=====
Demo Functions:
[Autoscroll] [Show timestamp] No line ending 9600 baud Clear output
```

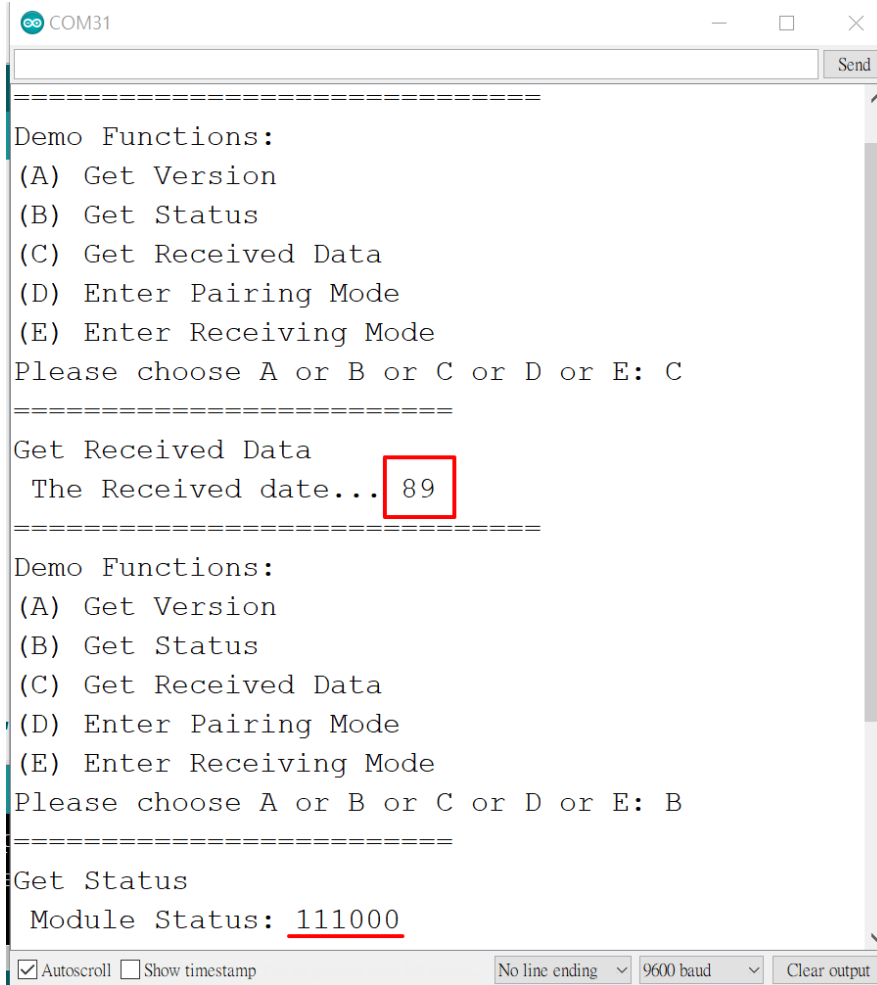
When the module is transmitting data, input B to read status, both the bit 3 and bit 4 equal 1 now.

At this time, if you have the corresponding receiver module BCM-2401-C03 and let the receiver module runs its example code, you may input C to read the received data, and 89 is the value received.

The status byte definition of the receiver module BCM-2401-C03 is listed below for reference.

- bit2=1 means the module is in pairing mode
- bit3=1 means the module is in receiving mode
- bit4=1 means the module has ever completed the pairing process
- bit5=1 means the module has data to be read
- other bits are not used and read as 0

At this time, read the receiver module status and we get 111000b, i.e. bit 3 to bit 5 are all 1.

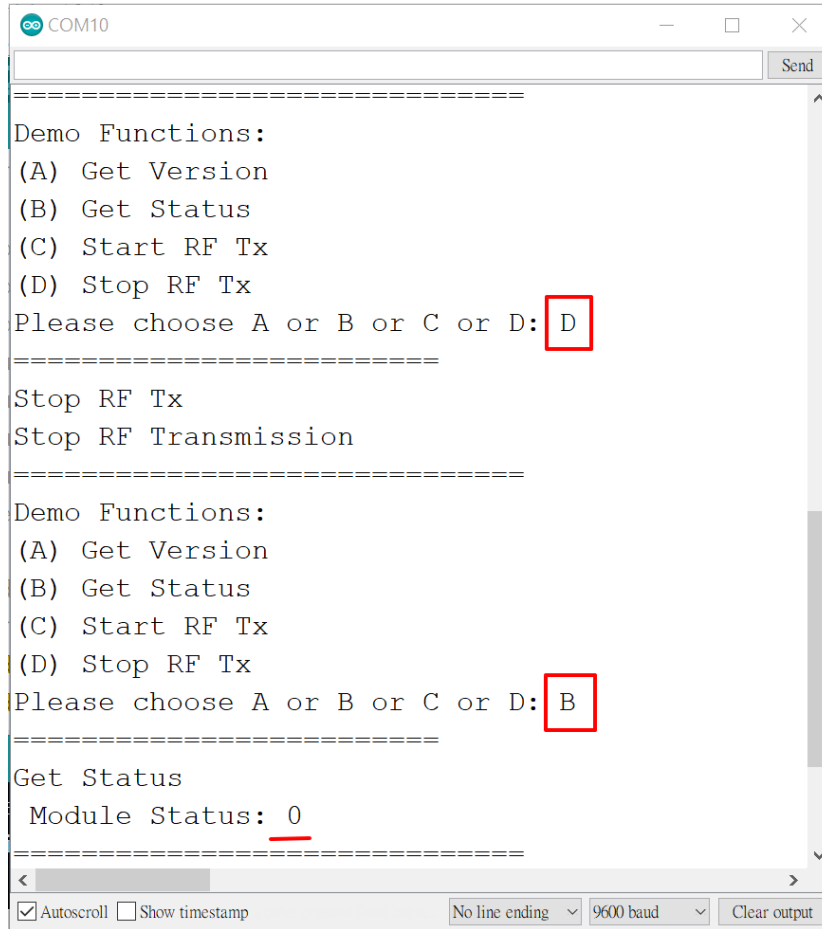


```
COM31
=====
Demo Functions:
(A) Get Version
(B) Get Status
(C) Get Received Data
(D) Enter Pairing Mode
(E) Enter Receiving Mode
Please choose A or B or C or D or E: C
=====
Get Received Data
The Received date... 89
=====
Demo Functions:
(A) Get Version
(B) Get Status
(C) Get Received Data
(D) Enter Pairing Mode
(E) Enter Receiving Mode
Please choose A or B or C or D or E: B
=====
Get Status
Module Status: 111000
=====
 Autoscroll  Show timestamp
No line ending 9600 baud Clear output
```

Note: this image comes from executing the example code of the receiver module BCM-2401-C03

Stop RF Transmission

Before transmitting a new data byte, you must input D to stop RF transmission. After you input D, and then input B to read the module status, now we get 0, the module is not transmitting.



```
COM10
=====
Demo Functions:
(A) Get Version
(B) Get Status
(C) Start RF Tx
(D) Stop RF Tx
Please choose A or B or C or D: D
=====
Stop RF Tx
Stop RF Transmission
=====
Demo Functions:
(A) Get Version
(B) Get Status
(C) Start RF Tx
(D) Stop RF Tx
Please choose A or B or C or D: B
=====
Get Status
Module Status: 0
=====
Autoscroll Show timestamp No line ending 9600 baud Clear output
```