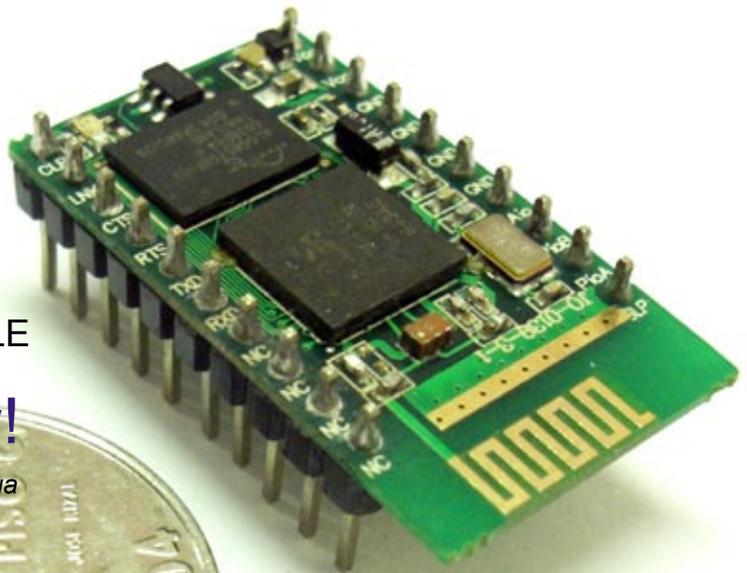


# EGBC-04



Bluetooth made easy!

Written by Henry L. Chua  
Rev. 1r1 June 6, 2007



Bluetooth and Bluetooth Logo are registered trademark of Bluetooth SIG Inc.

Bluetooth wireless technology provides a very reliable and convenient means of making products connect to each other- minus the wires- making possible seamless integration of stunning range of products.

For sure, a lot of experimenters and hobbyists would want to add bluetooth functionality to their circuits, but are prevented from by the complexity involved with the physical and software aspect of the technology. The introduction of Bluetooth chip set by companies, such as the BC-04 chipset (used in this module) from Cambridge Silicon Radio (CSR), may have simplified the hardware enough - yes, it is possible to assemble yourself the hardware part, assuming you can buy the chipset itself and do have the equipment to solder BGA IC package. You are probably also thinking of

writing your own chipset firmware. Read the bluetooth software layer and services requirement as a start. Visit <http://www.bluetooth.com/Bluetooth/Learn/>. See *you back here*.

This bluetooth module takes all these colossal work away from you. It simplifies bluetooth integration by presenting itself as a simple UART channel to its host. Sending and receiving data to and from another Bluetooth device is reduced to a simple task of reading and writing to the equivalent UART channel. All complex bluetooth tasks involved – authentication, encryption, transport, channel hopping, etc, are automatically performed and hidden from your view by the EGBC-04.

## Specifications

**Chipset:** CSR BC04

**Bluetooth Compatibility:** V1.2 and V2.0

**Range:** Class 2 (>10meters)

**Power Supply:** 3.0-3.3VDC

**I/O :** LV Logic level UART, 4 I/O

**Data Rate:** up to 1.2 Mbps

**Preloaded services:** LM,LC,L2cap  
SDP and RFCOMM

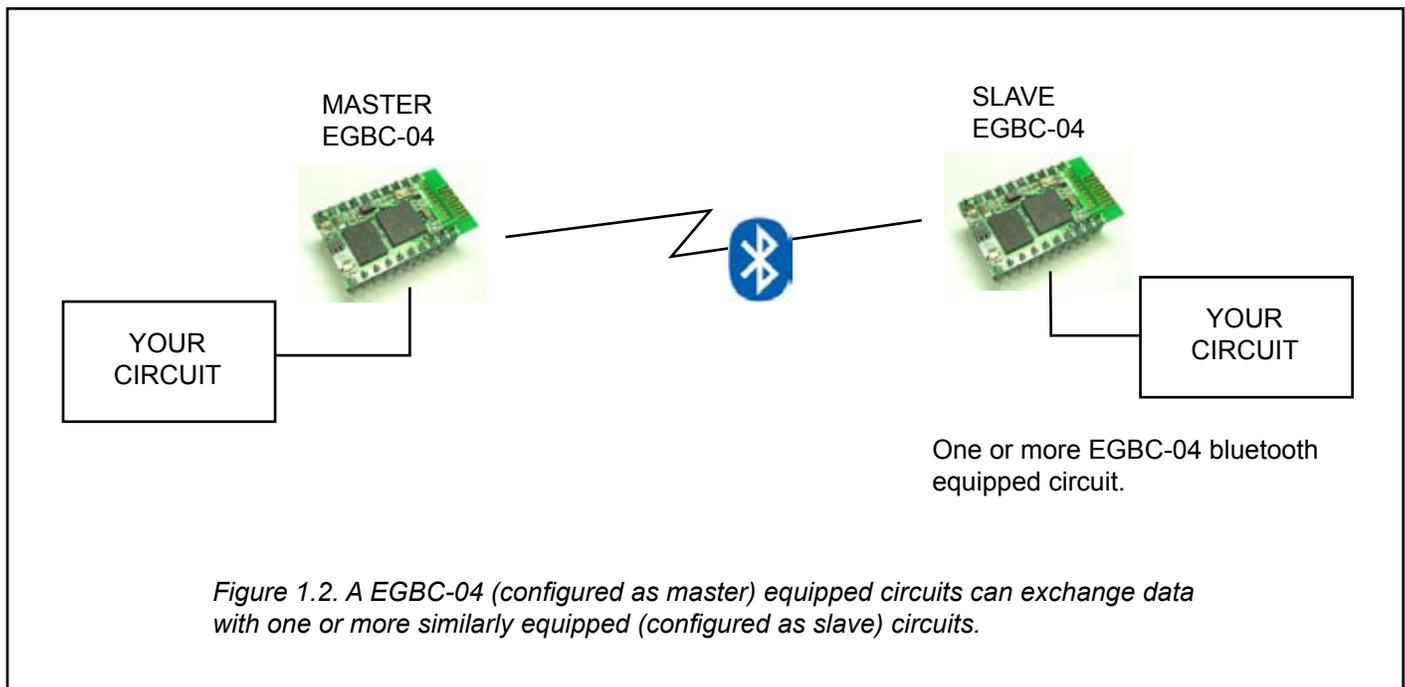
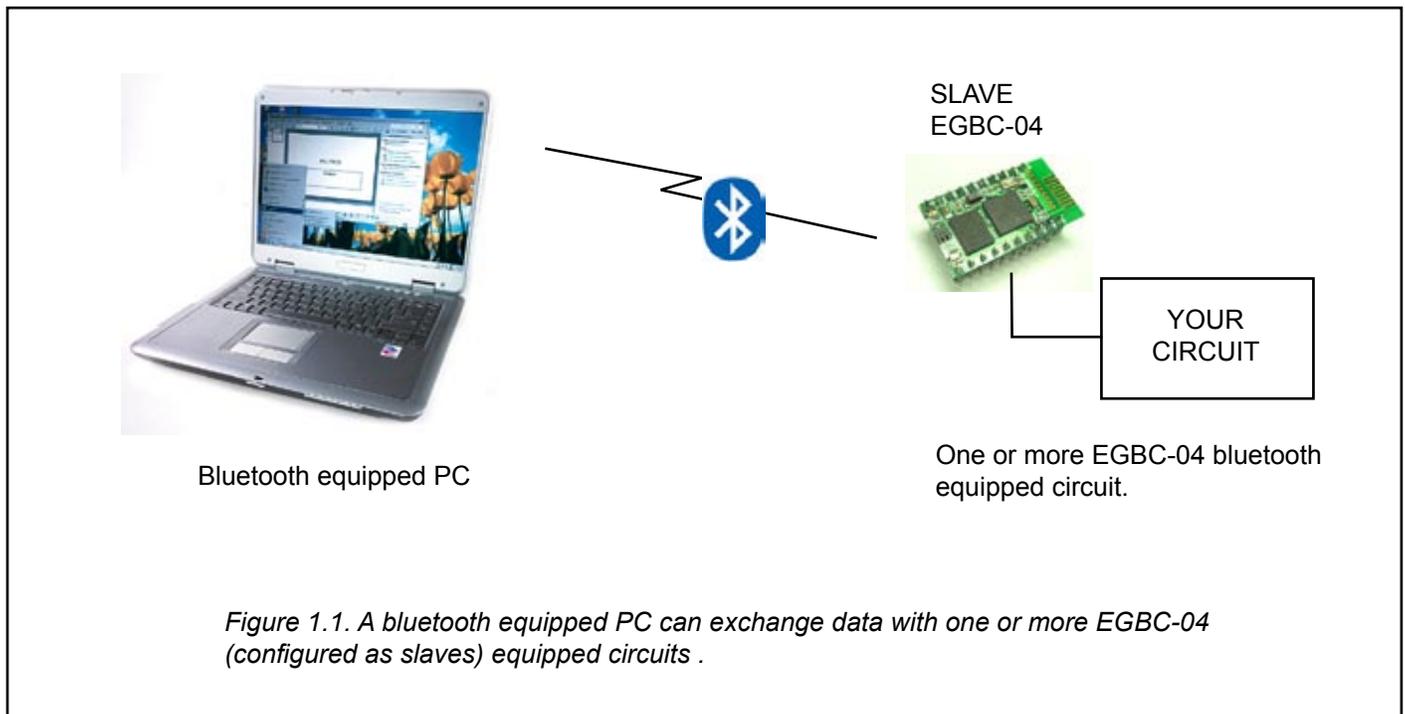
**Preloaded profile:** Serial Profile SPP.

**Dimensions:** 31.1 x17.7 x1.82 mm



## 1. Example Configurations

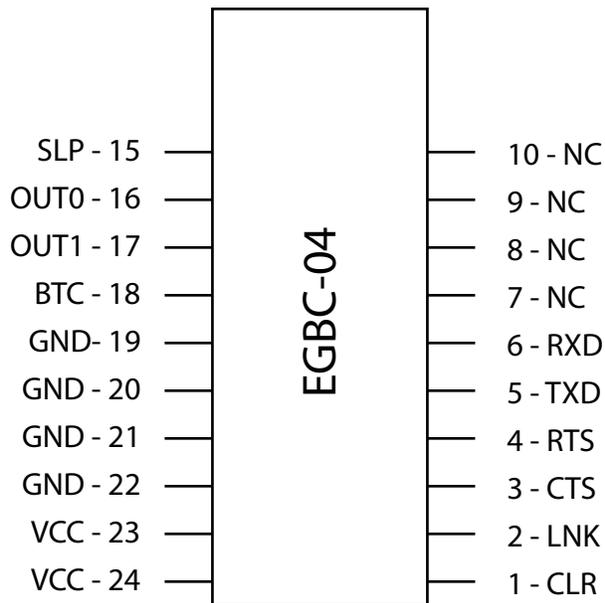
EGBC-04 mimic and replaces wired RS-232C connections. Your PC applications and/or application circuits see normal UART COM port connections and will not even know they are connected via wireless bluetooth network.



### Note:

Communication distance is guaranteed at 10 meters minimum. Actual coverage can easily exceed twice this distance if transfer rate is kept at or below 9600 bps. In general, faster data rates results in shorter coverage distance.

## 2. EGBC-04 Pin Description



TOP VIEW



### CAUTION

The EGBC-04 is ESD sensitive device. You can damage it with improper handling.

### Read

[www.e-gizmo.com/ARTICLES/ESD/ESD1.HTM](http://www.e-gizmo.com/ARTICLES/ESD/ESD1.HTM)  
for more details about ESD safe handling.

PIN	NAME	DIR	DESCRIPTION
1	CLR	I	Mode of operation: H - Normal Operation L – Parameter Setting Mode
2	LNK	O	H – Bluetooth Link established
3	CTS	I	UART Clear To Send handshake input
4	RTS	O	UART Ready To Send handshake output
5	TXD	O	UART Serial Data output
6	RXD	I	UART Serial Data input
7-10	NC	x	No connection. Do not connect to anything!
15	SLP	I	Sleep/Wake-up/Clear Address. See Section 3 for details
16	Out0	O	Output Port. See AT Command Manual for usage info.
17	Out1	O	Output Port. See AT Command Manual for usage info.
18	BTC	I	Battery Voltage Control. Do not connect
19-22	GND	P	Power supply GND
23,24	VCC	P	Power supply (+) input. 2.7V to 3.3V 3.0V nominal

### Notes:

CTS-RTS : Connect through 1K resistor if communication handshaking is not used.

### Important:

The Bluetooth module will be damaged when power polarity is connected the wrong way!  
The Bluetooth module is ESD sensitive device.

### 3. SLP - Sleep/Wake-Up/Memory Clear input

This input is kept at logic high during normal operations. Momentarily pulling this input low will result in the following operations:

- If SLP input is pulled low for less than 10ms, EGBC-04 will ignore this event.
- If the SLP input is pulled low for more than 2 seconds, EGBC-04 will erase all stored address.
- If SLP input is pulled low for less than 2 seconds while EGBC-04 is operating, the module will switch to power saving sleep mode.
- If SLP input is pulled low for less than 2 seconds while EGBC is in sleep mode, the module will switch back to active operating mode.

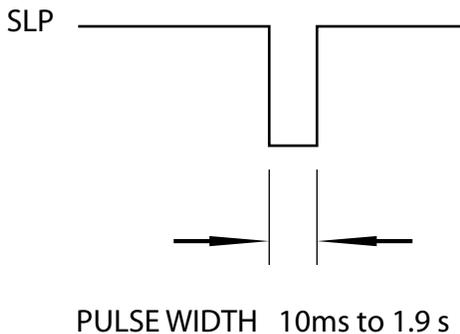


Figure 3.1. The SLP input functions as SLEEP and Wakeup input if pulsed low for less than 2 secs (> 10ms).

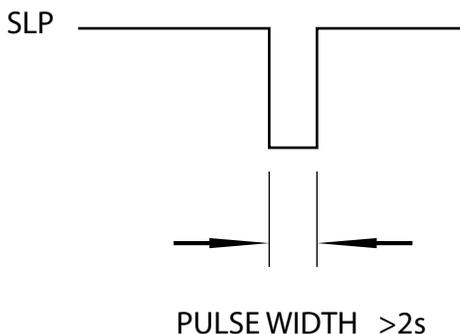


Figure 3.2. The SLP input function as ADDRESS MEMORY CLEAR when pulsed low for more than 2 secs.

### 4. Parameter Setting

The EGBC-04 module is already preconfigured as sold. In most cases, you do not have to change any of the parameters. The EGBC-04 is preconfigured as follows:

Mode: Slave  
Baud Rate: 9600bps  
Data bits: 8 data bits, 1 stop bit  
Handshake : None  
Device Name: eGizmo EGBC04  
Authentication: Enabled  
Password: 1234  
Encryption: Enabled

You will need the following equipment and circuits (see section 5) if you want to change any of the parameters:

- PC running a terminal emulation software, e.g. Hyperterminal.
- A 3V logic to RS232 converter cable.

1. Connect the CLR pin of EGBC-04 to GND. This forces EGBC-04 to Parameter Change Mode of operation.
2. Connect the EGBC-04 with the PC through the RS232 converter cable.
3. Launch the PC Hyperterminal program and open a session using the COM port where you connected the EGBC-04 setup. Set baud rate to 9600, and handshake control to "none".
4. With power already applied to BC-04, type

AT <enter>

through your PC hyperterminal.

If you got a "OK" response, the hardware setup is good and you are now ready to change any of the parameters.

Example:

To change the password to "3456", type the following through the hyperterminal:

```
AT+PASSWORD=3456 <enter>
```

To check if the password is correctly set, type

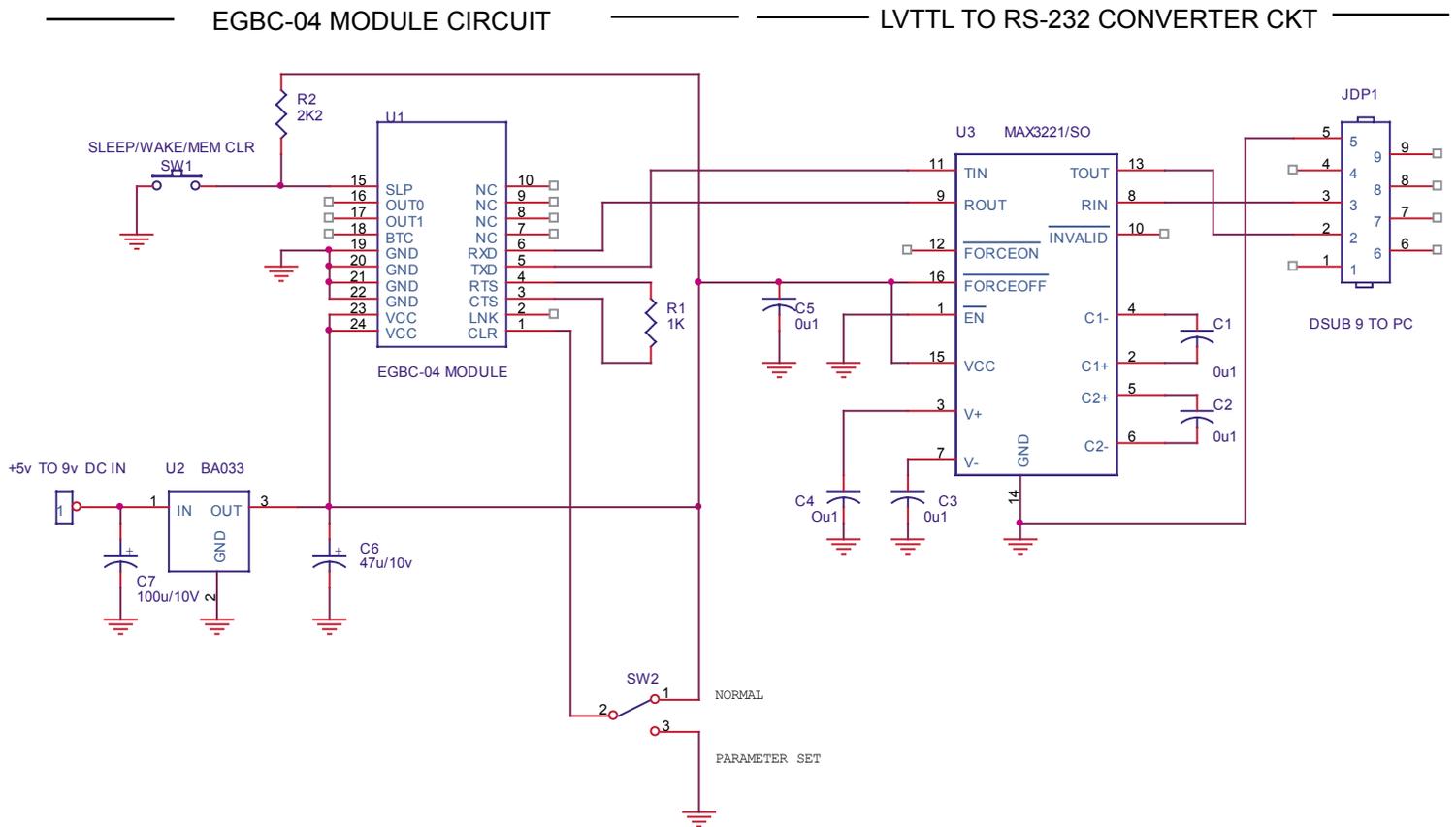
```
AT+PASSWORD?
```

Hyperterminal displays the last entered password.

```
+PASSWORD: 3456
```

The complete list of available AT command set is provided in the AT Command Manual.

## 5. Test Circuit (Parameter Setting) Schematic



## 6. Typical Application Circuit

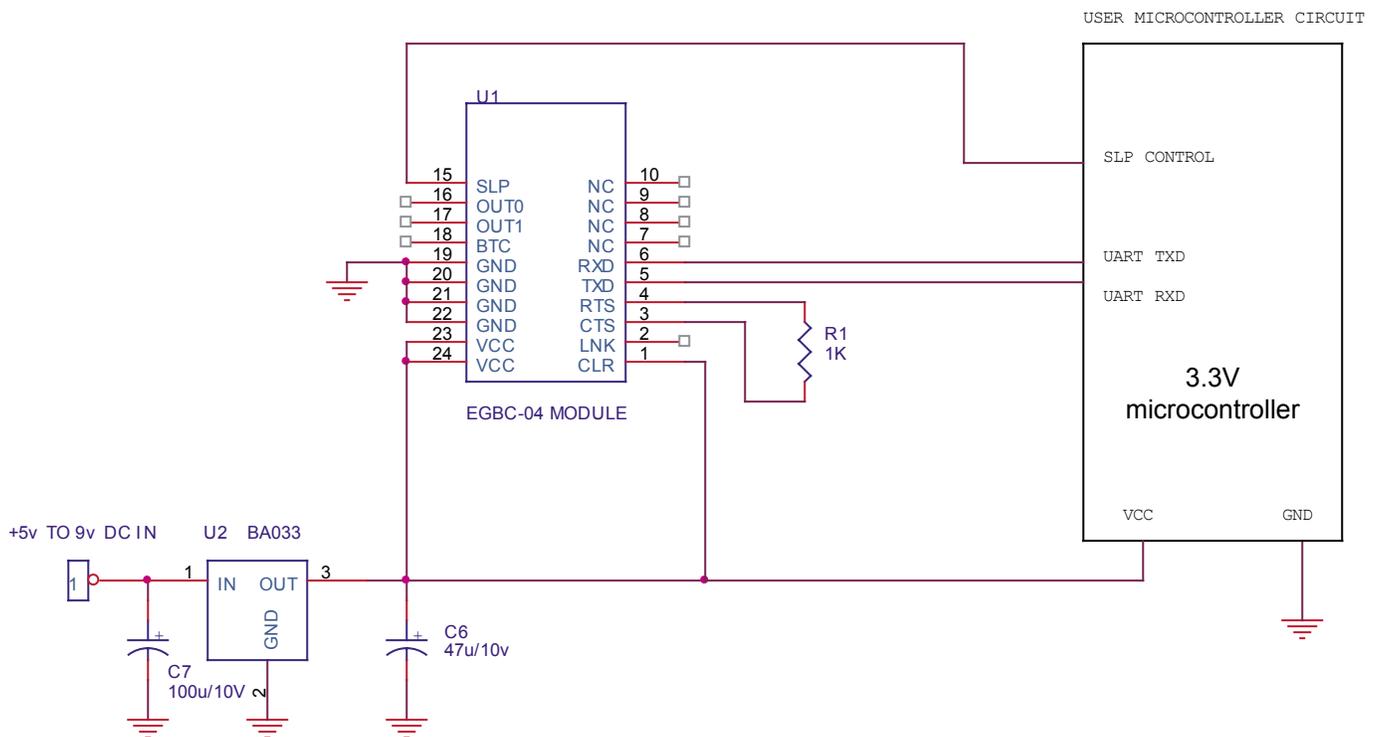


Figure 6.1. Interfacing the EGBC-04 to a 3.3V microcontroller circuit. You can connect SLP input pin 15 directly to Vcc if power saving feature is not used. U2 is a 3.3V voltage regulator. MCU with UART makes programming easier; software UART simulation will work just fine if your MCU do not have a UART.

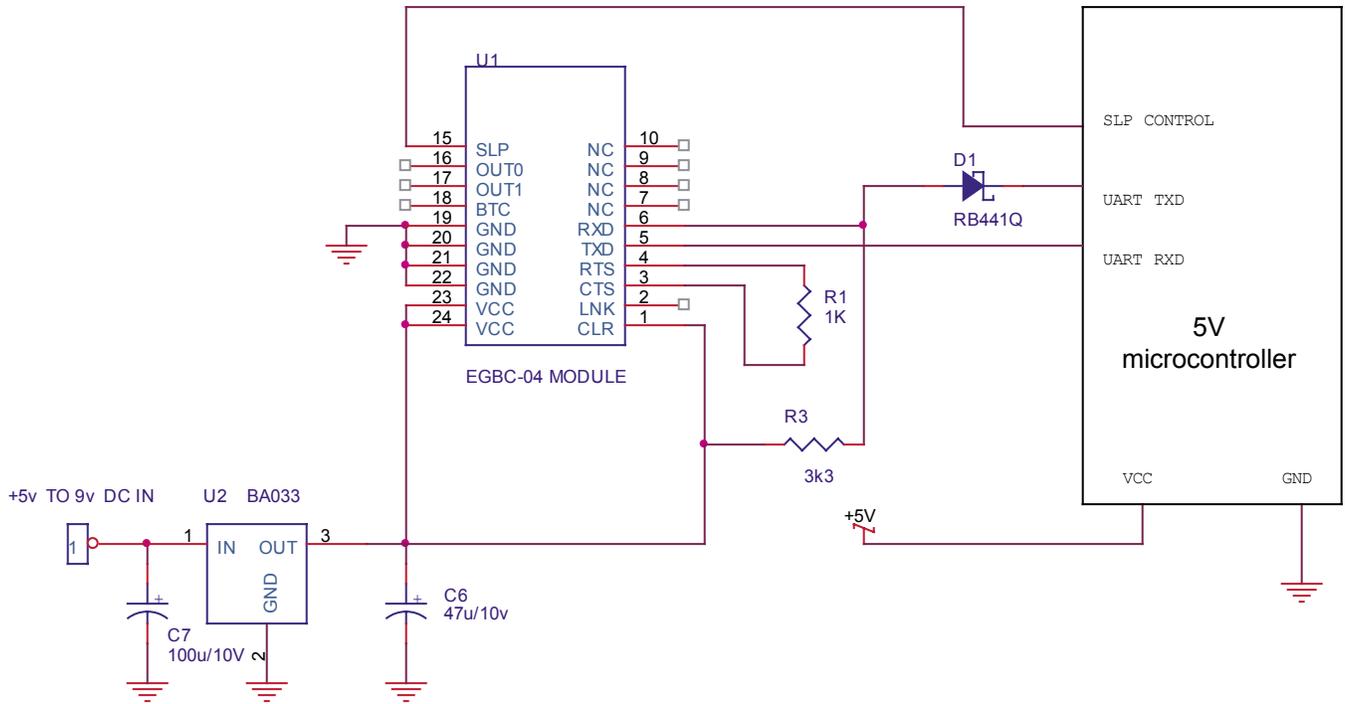


Figure 6.2. EGBC-04 works on 3V logic level. If a 5V microcontroller is used, you must insert D1 and R3 in the RXD input of EGBC-04 as shown. These components translate the 5V output of the MCU into a 3V logic that the EGBC-04 can safely handle.

## 7. PC Side Set up Example

The EGBC-04 can connect to any bluetooth equipped PC, such as notebook computers with internal bluetooth or external USB type bluetooth dongle.

Connecting EGBC-04 based device to a bluetooth equipped PC involves the following general steps:

- Device discovery: Let your PC search and find your EGBC-04.
- Password setup: Tell your PC the EGBC-04 password to make available EGBC-04 services to the PC.
- Connect and determine equivalent COM port.

This example illustrates how to setup a EGBC-04 connection using the popular BlueSoleil from IVT corporation.

1. Click **My Bluetooth > Bluetooth Device Discovery**. Wait until the discovery process is completed.



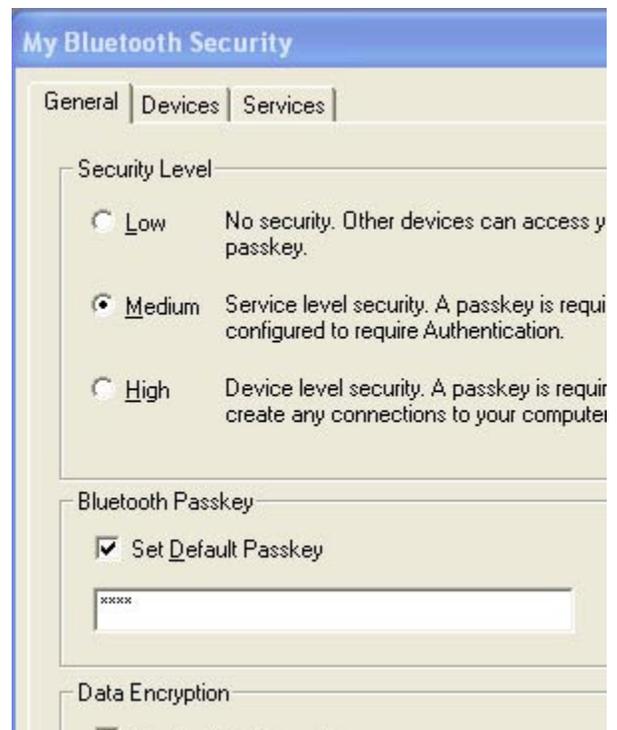
2. You should find EGBC-04 among the devices detected. Click on the EGBC-04 icon.



3. Click **My Bluetooth > Security..**



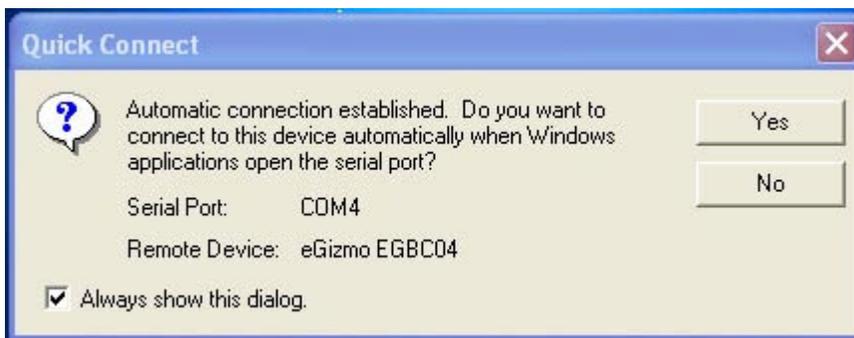
4. Select Medium security level and enter password (EGBC-04 default = 1234). Click OK after entry is completed.



5. Double click on the EGBC-04 icon to activate device service discovery. The serial port icon should be highlighted indicating acceptance of password and activation of Serial Profile.



6. Double click on the Serial port Icon top connect with the EGBC-04 serial device. A window will appear similar one shown below indicating the COM port assignment of your EGBC-04 bluetooth service.



Your remote EGBC-04 equipped bluetooth circuit can now be accessed through the indicated COM port (COM4 in the example). All windows application will now see EGBC-04 as an ordinary com port, and use it as such. As an example, with Visual Basic, you can now communicate with your EGBC-04 equipped circuits via the MSCOMM object. ●