Martyn Currey
Mostly Arduino stuff

Arduino with HC-05 (ZS-040) Bluetooth module – AT MODE

Posted on **October 28, 2014**

Update 20.07.2017

The zs-040 breakout boards are now being used for many different modules and you may not have the exact same boards as those shown below. The modules I am using here use the EGBT-045MS Bluetooth module and have the HC/Wavesen 2.0-20100601 firmware. They also have an EN pin rather than a KEY pin and they have a small button switch just above the EN pin.

There are now newer zs-050 modules that use the real HC SMD modules, these have a newer firmware and include a blue LED at the top right of the SMD daughter board.

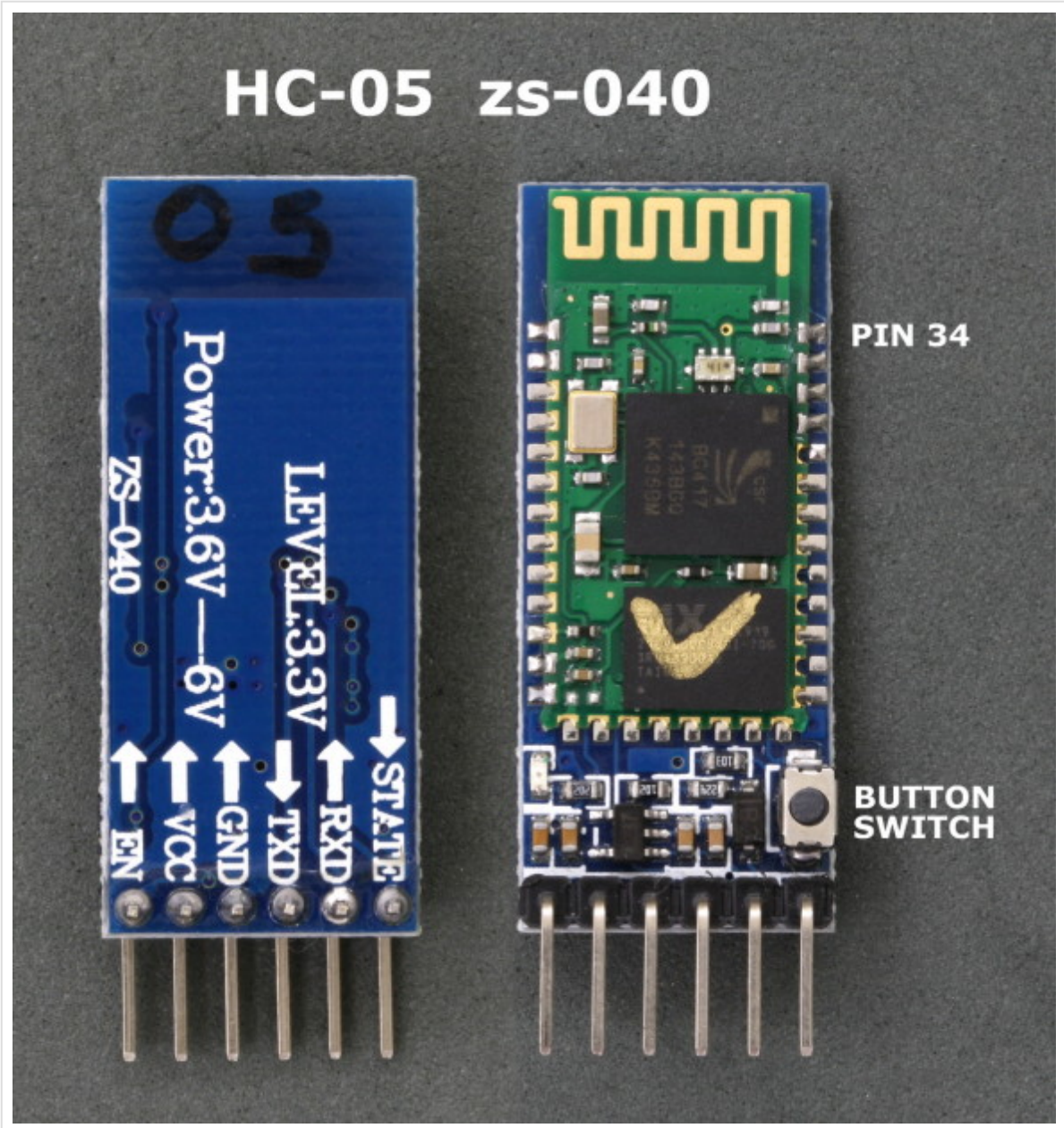
There are also modules that use the same breakout board but have different board markings such as the fc-114 modules:
HC-05 FC-114 and HC-06 FC-114. First Look
HC-05 FC-114 and HC-06 FC-114. Part 2 – Basic AT commands
HC-05 FC-114 and HC-06 FC-114. Part 3 – Master Mode and Auto Connect‘

AT Command Mode

AT command mode allows you to interrogate the Bluetooth module and to change some of the settings; things like the name, the baud rate, whether or not it operates in slave mode or master mode.
When used as a master device AT commands allow you to connect to other Bluetooth slave devices.

The HC/Wavesen 2.0-20100601 firmware has different AT modes:
A full mode at 38400 baud.
A partial mode at 38400 baud.
A partial mode at the user settable baud rate.
A partial mini mode (even less commands work) at the user settable baud rate.

Most commands work when in any AT command mode but there are some commands that only work when pin34 is HIGH. This fooled me for quite a while. I now believe the partial/mini AT command mode is a bug in the firmware and I now only recommend using the full 38400 baud rate AT command mode.



To activate AT mode on the HC-05 zs-040 modules pin 34 needs to be HIGH on power up. The small push button switch connects pin 34 to +3.3v so we can either:

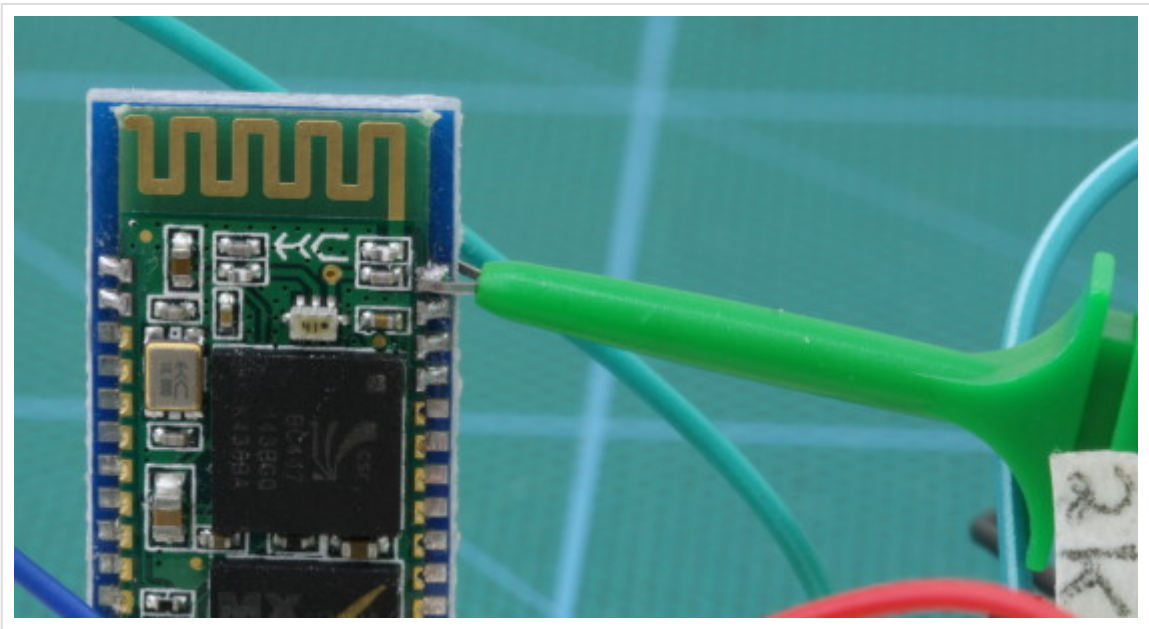
– connect pin 34 directly to +3v3v and power on, or
– hold the button switch closed when starting the module.

Please be aware that for the full AT command mode, pin 34 has to be HIGH all the time and we cannot do this with just the button switch. When in AT command mode with pin 34 not HIGH (LOW or floating) some commands will not work and so, when using the button switch, you need to press and hold it closed when sending some commands.

Using the button switch to enter AT command mode using 38400 baud rate:
– 1. remove power from the module
– 2. Hold the small button switch closed while powering on the module.
– 3. Press and hold the button switch.
– 4. While still holding the button switch closed, apply power.
– 5. When you see the LED come on you can release the button switch.

Enters AT mode with the built in AT mode baud rate of 38400. The baud rate cannot be changed by the user.
This method allows the module to enter AT mode on start but but does not keep pin 34 HIGH and some commands will not work.

Alternatively, if you are using AT command mode for any length of time, make a direct connection between pin 34 and +3.3v. Either solder a wire to the pin or use a clip.



Using pin 34 to enter full AT command mode using 38400 baud rate.
– 1. Remove power from the module
– 2. Make a connection between pin 34 and +3.3v
– 3. Reapply power.

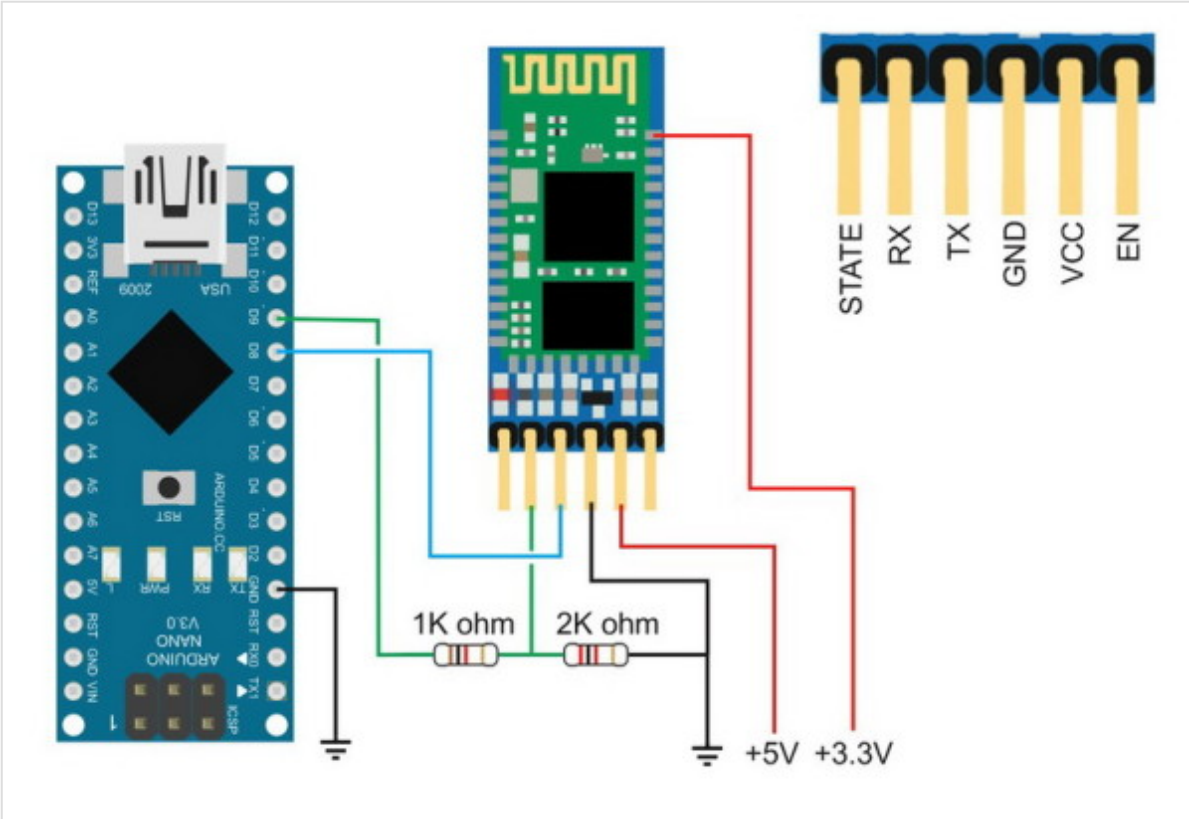
Enters AT mode with the built in AT mode baud rate of 38400. The baud rate cannot be changed by the user.

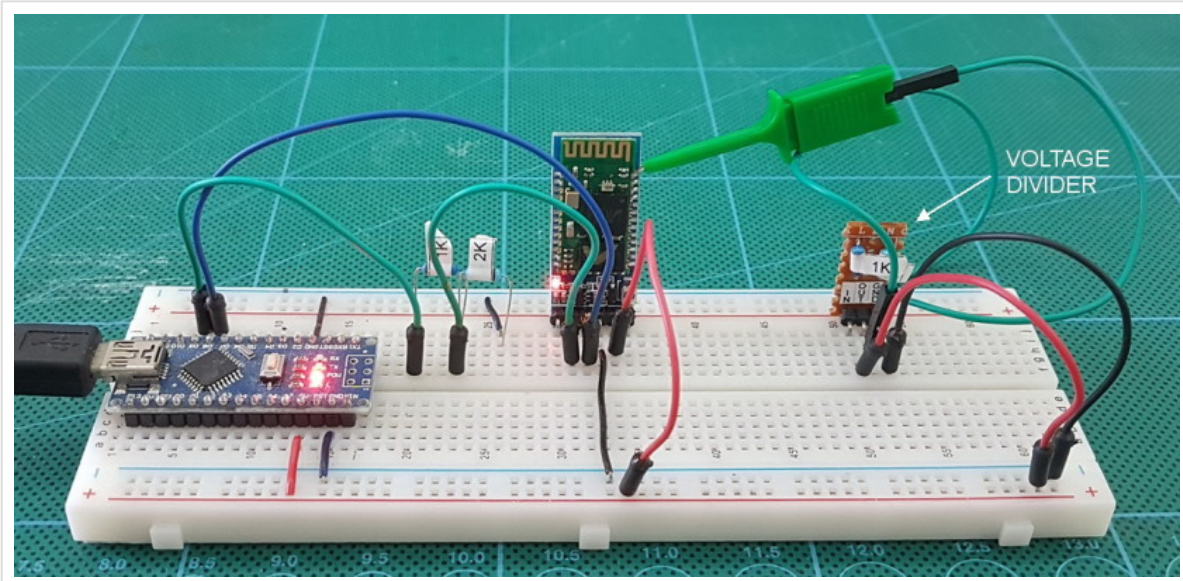
If you keep pin 34 HIGH you will enable the “full” AT mode which allows all AT commands to be used.
If you let pin 34 return LOW after power on then “mini” AT mode will be enabled.

Connecting the HC-05 to the Arduino

There are a couple of common ways to connect the HC-05 to a computer; via a serial UART adapter or with an Arduino. Here I am using an Arduino.

Build the following circuit and upload the below sketch.





– Arduiono 5V to BT VCC
– Arduino GND to BT GND
– Arduino D8 to BT TX
– Arduino D9 to BT RX through a voltage divider (reduce 5v to 3.3V)
– Optional. Connection from pin 34 but do not apply 3.3v yet.

The sketch is a basic serial in – serial out that displays a “>” at the start of a line to high light the user entered command.

I use software serial on Arduino pins 2 and 3 to talk to the HC-05. This means I can still use the hardware serial to talk to the serial monitor on a host computer.

```
//  Sketc: basicSerialWithNL_001
//
//  Uses hardware serial to talk to the host computer and software serial
//  for communication with the Bluetooth module
//  Intended for Bluetooth devices that require line end characters "\r\n"
//
//  Pins
//  Arduino 5V out TO BT VCC
//  Arduino GND to BT GND
//  Arduino D9 to BT RX through a voltage divider
//  Arduino D8 BT TX (no need voltage divider)
//
//  When a command is entered in the serial monitor on the computer
//  the Arduino will relay it to the bluetooth module and display the result.
//


#include <SoftwareSerial.h>
SoftwareSerial BTserial(8, 9); // RX | TX

const long baudRate = 38400;
char c=' ';
boolean NL = true;
```

```
void setup()
{
    Serial.begin(9600);
    Serial.print("Sketch:   ");   Serial.println(__FILE__);
    Serial.print("Uploaded: ");   Serial.println(__DATE__);
    Serial.println(" ");

    BTserial.begin(baudRate);
    Serial.print("BTserial started at "); Serial.println(baudRate);
    Serial.println(" ");
}

void loop()
{

    // Read from the Bluetooth module and send to the Arduino Serial Monitor
    if (BTserial.available())
    {
        c = BTserial.read();
        Serial.write(c);
    }


    // Read from the Serial Monitor and send to the Bluetooth module
    if (Serial.available())
    {
        c = Serial.read();
        BTserial.write(c);

        // Echo the user input to the main window. The ">" character indicates the user e
        if (NL) { Serial.print(">");  NL = false; }
        Serial.write(c);
        if (c==10) { NL = true; }
    }

}
```

Connect the Arduino to the host computer. The LED on the HC-05 should be blinking quickly at about 5 times a second.

If using the connection on pin 34, remove power, connect pin 34 to +3.3v and re-apply power to the module.

If using the button switch, remove power, press and hold the button switch, re-apply power (while holding the button switch). When you see the LED come release the button switch.

The LED should be blinking slowly on/off once every couple of seconds. This indicates AT mode.

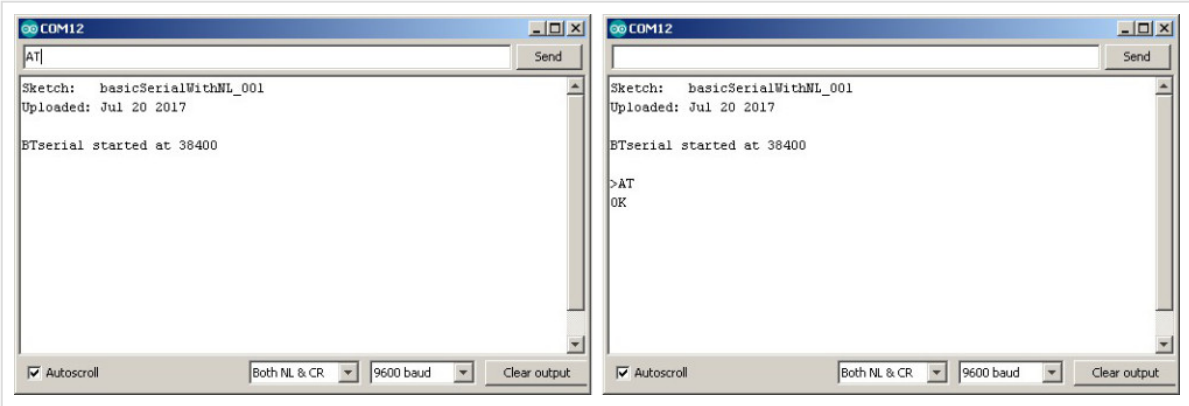
AT Commands

The HC-05 expects commands to include a carriage return and newline characters (\r\n). You can add these automatically in the serial monitor by selecting “Both NL & CR” at the bottom of the window.

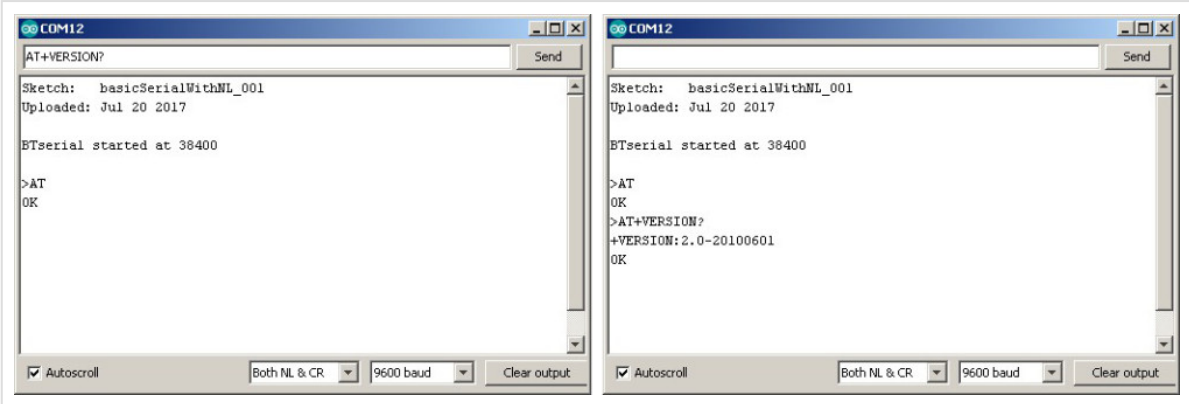
You can also enter them manually in the form AT\r\n. If you forget to add carriage return and newline characters the HC-05 will not respond.

Example commands

AT – simple feedback request. Will return “OK”



AT+VERSION – returns the firmware version.
“+VERSION:2.0-20100601
OK”



AT+STATE – returns the current state of the module
“+STATE:INITIALIZED
OK”

AT+ROLE – the possible values are ; 0 – Slave, 1 – Master, 2 – Slave-Loop
Returns
“+ROLE:0
OK”

To change to Master Mode, enter AT+ROLE=1, returns “OK”

AT+UART – returns the baud rate used by the HC-05 in communication mode. The default for the modules I have is 9600. Returns:
“+UART:9600,0,0
OK”

To change the baud rate to 38400 – AT+UART=38400,0,0
Returns “OK”

Windows does not support baud rates above 115200. If you accidentally set the baud rate higher than 115200 you will not be able to use communication mode. You should still be able to enter AT mode at 38400 using method 1 or method 2 above and change the communication mode baud rate to something Windows can handle.

AT+NAME
Querying the modules name with AT+NAME? only works in “full” At mode. If you cannot get AT+NAME? to work you need to bring pin34 HIGH.
Changing the modules name with AT+NAME=newname works in “full” AT mode and “mini” AT mode.

What you should get is:
AT+NAME?, returns
+NAME:HC-05
OK
(or something similar depending what your module is called)

Other commands that require pin 34 to be HIGH are AT+INQ and AT+RNAME. This is not a complete list through.

Full list of AT commands

AT COMMAND LISTING

	COMMAND	FUNCTION
1	AT	Test UART Connection
2	AT+RESET	Reset Device
3	AT+VERSION	Query firmware version
4	AT+ORGL	Restore settings to Factory Defaults
5	AT+ADDR	Query Device Bluetooth Address
6	AT+NAME	Query/Set Device Name
7	AT+RNAME	Query Remote Bluetooth Device's Name
8	AT+ROLE	Query/Set Device Role
9	AT+CLASS	Query/Set Class of Device CoD
10	AT+IAC	Query/Set Inquire Access Code
11	AT+INQM	Query/Set Inquire Access Mode
12	AT+PSWD	Query/Set Pairing Passkey
13	AT+UART	Query/Set UART parameter
14	AT+CMODE	Query/Set Connection Mode
15	AT+BIND	Query/Set Binding Bluetooth Address
16	AT+POLAR	Query/Set LED Output Polarity
17	AT+PIO	Set/Reset a User I/O pin
18	AT+MPIO	Set/Reset multiple User I/O pin
19	AT+RMOD	Query User I/O pin
20	AT+PSCAN	Query/Set Scanning Parameters
21	AT+SNIFF	Query/Set SNIFF Energy Savings Parameters
22	AT+SENM	Query/Set Security & Encryption Modes
23	AT+RMSAD	Delete Authenticated Device from List
24	AT+FSAD	Find Device from Authenticated Device List
25	AT+ADCN	Query Total Number of Device from Authenticated Device List
26	AT+MRAD	Query Most Recently Used Authenticated Device
27	AT+STATE	Query Current Status of the Device
28	AT+INIT	Initialize SPP Profile
29	AT+INQ	Query Nearby Discoverable Devices
30	AT+INQC	Cancel Search for Discoverable Devices
31	AT+PAIR	Device Pairing
32	AT+LINK	Connect to a Remote Device
33	AT+DISC	Disconnect from a Remote Device
34	AT+ENSNIFF	Enter Energy Saving mode
35	AT+EXSNIFF	Exit Energy Saving mode

ERROR CODES

ERROR CODE	VERBOSE
0	Command Error/Invalid Command
1	Results in default value
2	PSKEY write error
3	Device name is too long (>32 characters)
4	No device name specified (0 length)
5	Bluetooth address NAP is too long
6	Bluetooth address UAP is too long
7	Bluetooth address LAP is too long
8	PIO map not specified (0 length)
9	Invalid PIO port Number entered
A	Device Class not specified (0 length)
B	Device Class too long
C	Inquire Access Code not Specified (0 length)
D	Inquire Access Code too long
E	Invalid Inquire Access Code entered
F	Pairing Password not specified (0 length)
10	Pairing Password too long (> 16 characters)
11	Invalid Role entered
12	Invalid Baud Rate entered
13	Invalid Stop Bit entered
14	Invalid Parity Bit entered
15	No device in the Pairing List
16	SPP not initialized
17	SPP already initialized
18	Invalid Inquiry Mode
19	Inquiry Timeout occurred
1A	Invalid/Com length address entered
1B	Invalid Security Mode entered
1C	Invalid Encryption Mode entered

This list is taken from the EGBT-045MS bluetooth module user guide and not all commands may be supported or work straight away. For example AT+NAME? only works when pin 34 is HIGH.

For more information look at the HC-05 user guide or the EGBT-046S/EGBT-045MS user guide

Next: Linking 2 Bluetooth Modules

Connecting 2 Arduinos by Bluetooth using a HC-05 and a HC-06: Easy Method Using CMODE
Connecting 2 Arduinos by Bluetooth using a HC-05 and a HC-06: Pair, Bind, and Link

This entry was posted in **Arduino**, **Bluetooth** and tagged **arduino**, **AT mode**, **Bluetooth**, **HC-05** by **Martyn**. Bookmark the **permalink [http://www.martyncurrey.com/arduino-with-hc-05-bluetooth-module-at-mode/]** .

---

224 THOUGHTS ON “ARDUINO WITH HC-05 (ZS-040) BLUETOOTH MODULE – AT MODE”

montassar
on **February 13, 2015 at 3:17 pm** said:

Thank you so much , you’re article is the only that answered my question about how to get the zs40 into AT mode.