

JDY-08 : bluetooth 4.0 module

The JDY-08 BLE module is cheap and not very well documented. It's mandatory to update it before unleashing it's power!

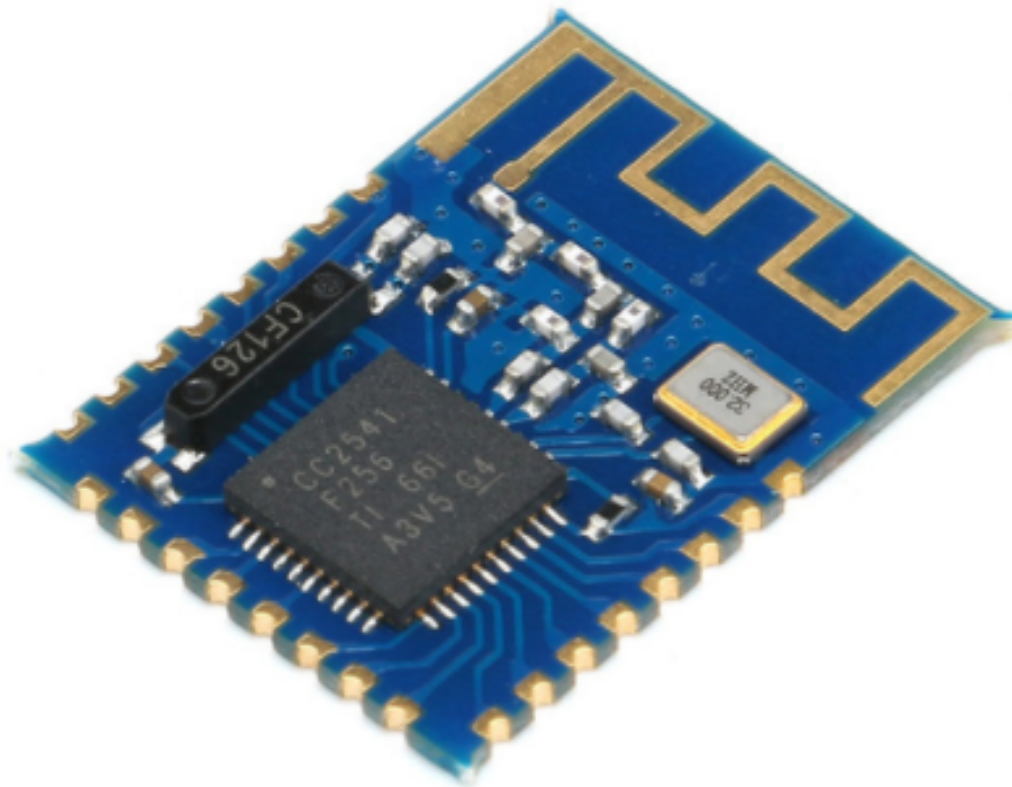


JEAN-MATTHIEU DECHRISTÉ

21 AUG 2017 • 4 MIN READ

 JDY-08 : bluetooth 4.0 module

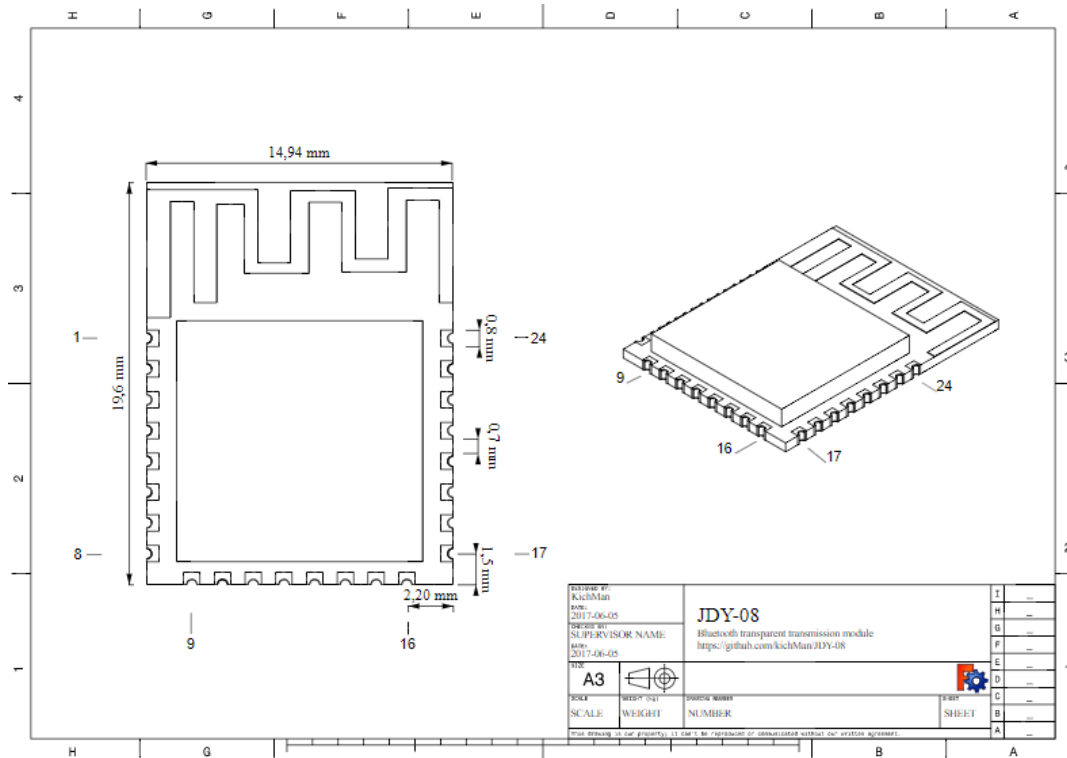
I recently bought a bluetooth LE module, the JDY-08.



It is a very small module, based on the TI CC2541 chip.

You can configure it with AT commands, on a serial line.

You can find the pinout on the kichMan github:



And some documentation on a shared Google doc.

And finally, the chinese datasheet.

The module has to be powered by 3.3V.

Warning: The firmware loaded by default on this module is not the same as on HM-10 or EY-08 modules, so the behavior is different.

I have the version **JDY-08-V3.383**

Serial mode

To enter the serial mode :

- P00 has to be linked to GND
- P02 = RX
- P03 = TX

Now, you can send AT commands. Please notice:

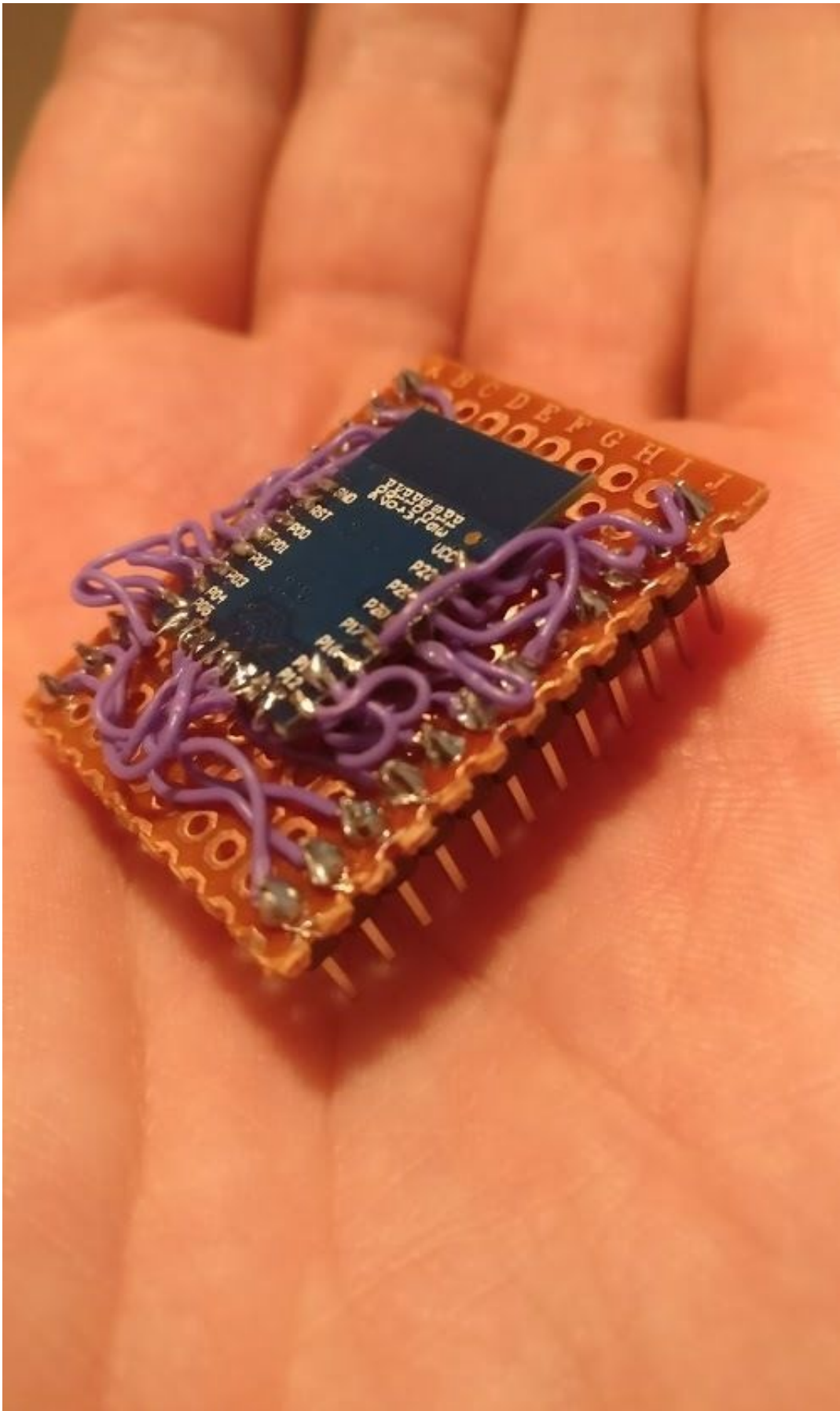
- The commands are key sensitive
- The default baudrate is 115200
- Don't send newline characters, the module will not understand the AT command.
- You have to write the whole command in one time, and not characters by characters (i.e. will not work in Putty).

Breakout board

As the device doesn't have a 2.54mm pitch, I build a custom breakout board for developpement purposes.

I hope I will be able to find a real / create a real PCB one day...





Flash the device with HM10

firmware

The firmware I have on one of my chip seems defective.

To be able to use it, I want to flash it with the HM10 firmware, based on the same chip (CC2541).

CC Debugger connection

I bought a CC Debugger :

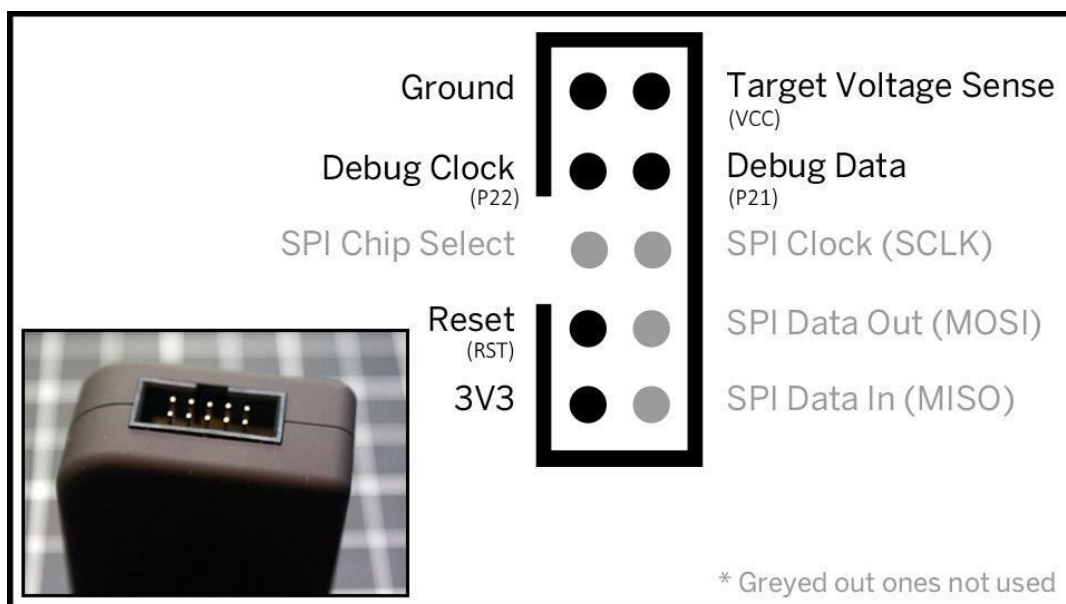




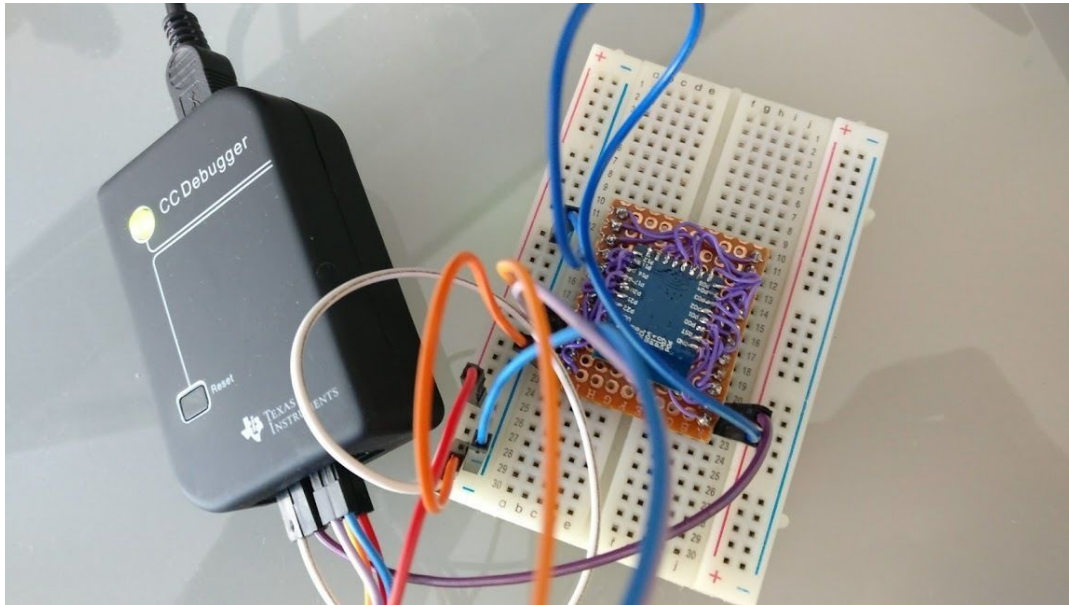
You can also do it with an arduino, but it's less simple.

You will need to install the driver of the CC Debugger

To connect it to the device, follow those instructions :



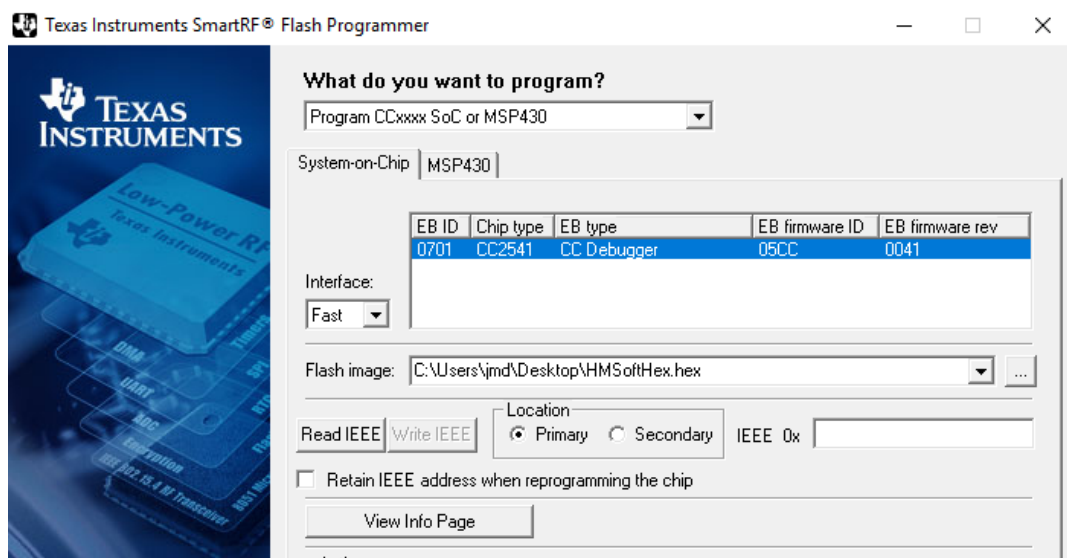
In picture :

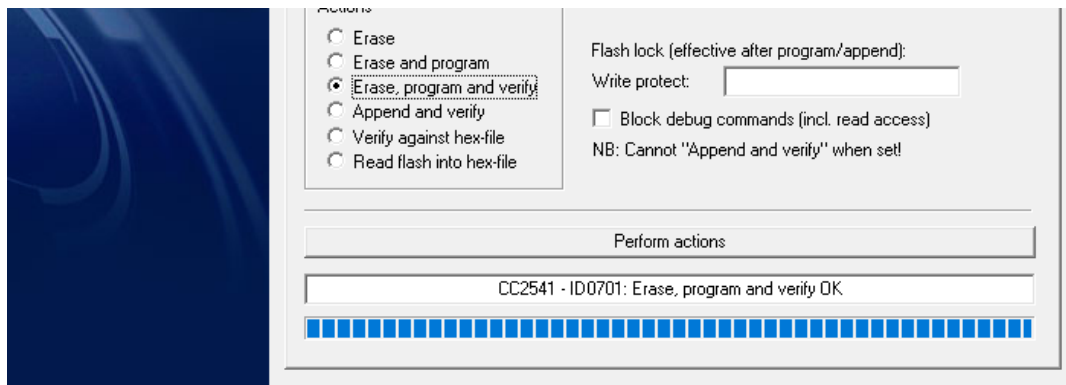


After powering the probe, the LED goes from red to green.

Flash the binary

To flash the new firmware, you need to download the SmartRF Flash Programmer from TI, and the firmware V550 (CC2541hm10v550.hex) that I created (see at the end of the post) - or the V540 (CC2541hm10v540.hex).





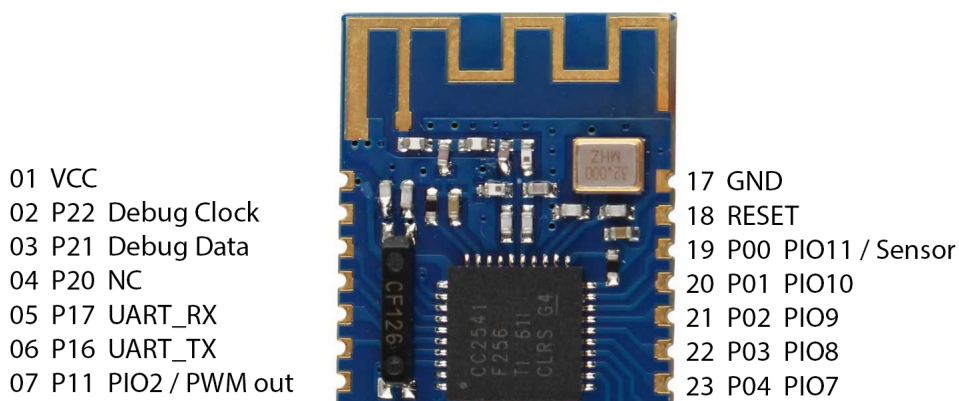
1. Launch "SmartRF Flash Programmer"
2. Select your probe if it's not selected by default
3. Select the hex file (firmware)
4. Uncheck "Retain IEEE address..."
5. Select "Erase, program and verify"
6. Click on "Perform actions" and wait the end of the process

After that, you should already be able to see the device with your phone, named "HMSoft".

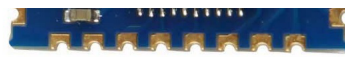
Test the new firmware

After flashing the HM-10 firmware, the pinout changed^[1]:

JDY-08 PINMAP
with HM-10 firmware



08 P12 PIO1 / System LED



24 P05 PIO6

16 P06 PIO5
 15 P07 PIO4
 14 P10 PIO3
 13 P14 UART_CTS
 12 PIN3 SDA
 11 PIN2 SCK
 10 P15 UART_RTS
 09 P13 PIO0 / System Key

Be carefull:

- P13 to GND is used to exit sleep mode or disconnect the module if connected (and not P00 anymore)
- P17 become UART RX
- P16 become UART TX

To send AT commands to the module through UART interface, the pin 09 (marked P13 on JDY-08, or the PIO0 on the HM-10) must be pulled down (GND) for at least 1 second after the board power on and after every bluetooth connection established.

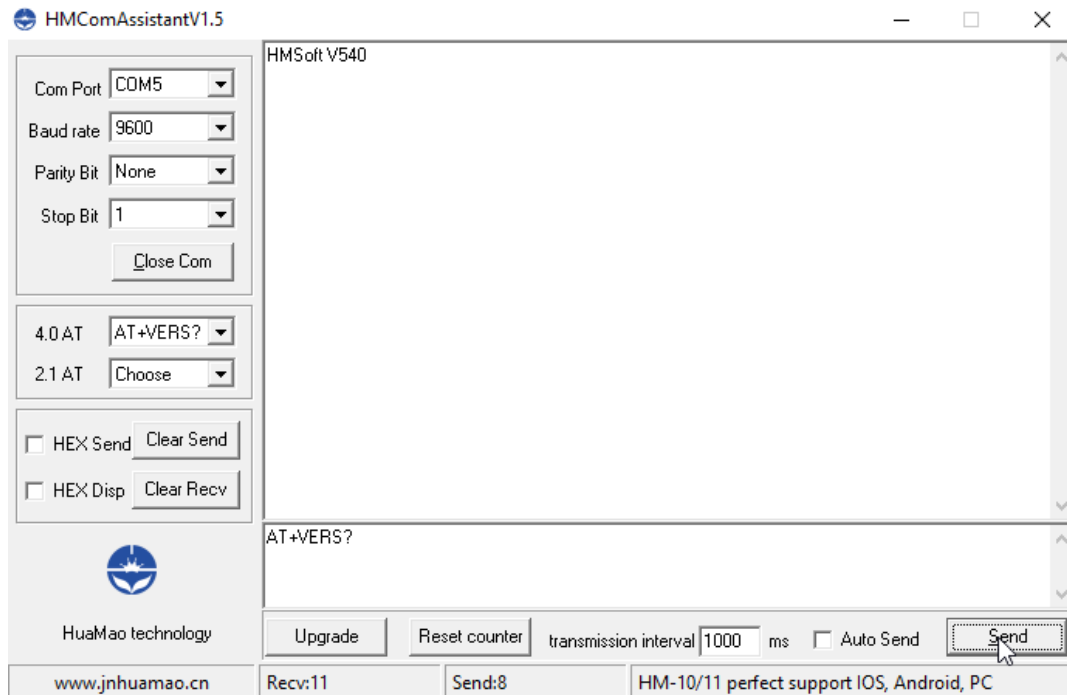
Moreover, the factory default setting with HM10 firmware:

- Baud: 9600, N, 8, 1;
- No carriage return after AT command;
- Pin code: 000000;

To test the command line, you can send `AT` (expected response `OK`) ou `AT+VERS?` .

To reset factory settings, send `AT+RENEW` (expected response `OK+RENEW`).

The software HMComAssistant.exe can help you to connect and exchange data with this firmware.



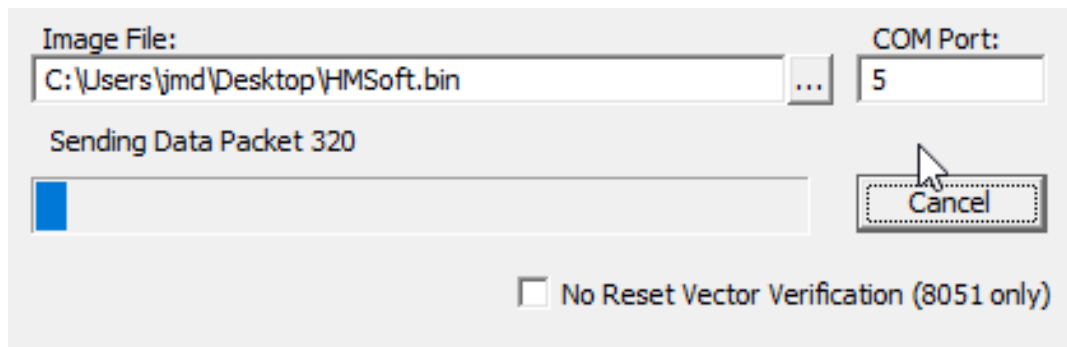
Upgrade the HM10 firmware

Once that a HM10 firmare has been flashed on the JDY08 module, you can upgrade the firmware with a serial connection.

To do that:

1. Send `AT+SBLUP` to the module (expected response: `OK+SBLUP`)
2. Close the COM port or your software used to send commands to the device
3. Launch the HMSoft.exe software
4. Select the new HM10 firmware file (bin), the COM port and click on "Load Image". The transfer should begin, and at the end, you should see "Download completed successfully".





Annex: How I created the HEX file

First, I tried to download the last firmware version (HM-10 HMSoft CC2541 V550 firmware upgrade file.....[2017-07]), extract the HMSoft.bin file, and transform it into a hex file with srecord : `srec_cat.exe HMSoft.bin -Binary -o HMSoftHex.hex -Intel`

Sadly, it didn't worked.

If found another file that seemed to work for some guys, named CC2541hm10v540.bin.

With this file, I created a hex file (`srec_cat.exe CC2541hm10v540.bin -Binary -o CC2541hm10v540.hex -Intel`), and I flashed it as explained before.

After that, I upgraded the firmware through serial port, with the last binary found on the manufacturer website (V550 the 2017-08-21).

As I did not activate the read lock, I was able to read the new firmware and create the final HEX file for the V550.

1. Source: Arduino forum ↩