

## Using HM-10 BLE Modules as Low-Cost iBeacons

*This document will describe how to use an HM-10 Bluetooth 4.0 module as an iBeacon. Although the HM-10 can also be used as a data link, a standalone sensor or control device, this document will not cover those other use cases.*

The HM-10 is a readily available Bluetooth 4.0 module based on the Texas Instruments CC2540 or CC2541 Bluetooth low energy (BLE) System on Chip (SoC). The module design and firmware originated from the Jinan Huamao Technology Company (JNHuaMao), but is sold by various Chinese suppliers, and by several U.S. and European distributors.

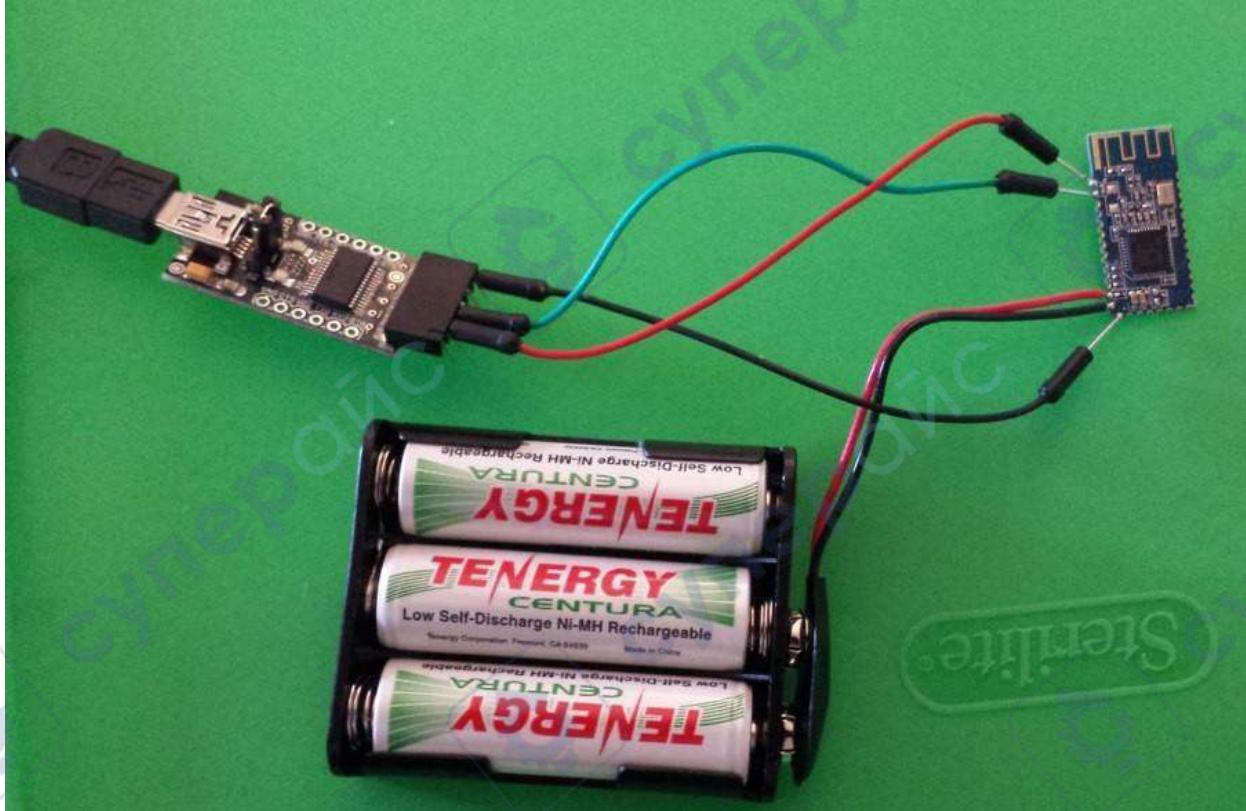
The modules shown here were purchased directly from ITead Studio's iMall web site for **only US\$6.50 each**. They came pre-loaded with HM-10 Firmware version 526, but at the time I received them, version 528 (June 2014) was already available from the JNHuaMao download site, so I upgraded all the modules (more about how to do that later). As of September 2014 firmware version 531 was available.



The module is fairly small as seen in the photograph above. The contact spacing is 1.5 mm. **Care should be taken when soldering wires to the module** which will be necessary to connect it to a power source, and temporarily to a USB serial interface module and PC to configure the module as an iBeacon. The HM-10 is rated to operate at a **supply voltage of 2.0 to 3.7 volts**, and its I/O are 3.3V tolerant.

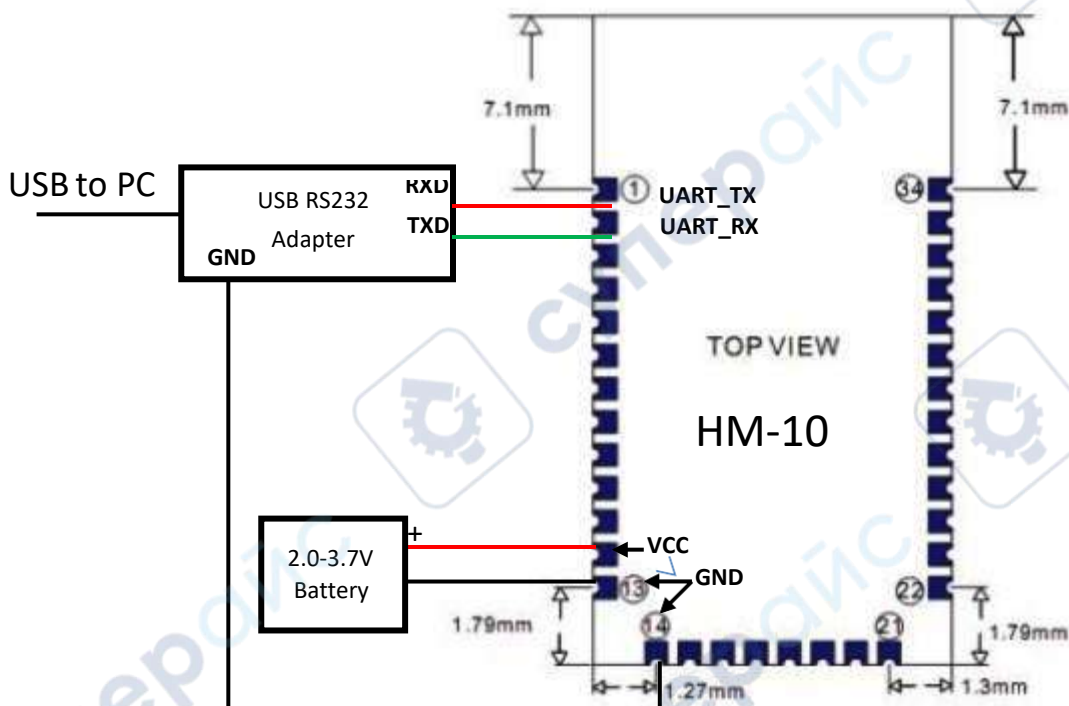
## Hardware Connections

Two wire connections are needed to supply power to the module and three temporary connections are needed to configure the module through a USB serial adapter.



The photo above shows the module hooked up to a battery pack consisting of three NiMH rechargeable cells with a nominal output of  $1.2V \times 3 = 3.6V$ , and to a USB to TTL logic level RS232 serial adapter. If you are an Arduino enthusiast you probably already have one of these adapters, if not they are available for about US \$15 online and from retailers like Fry's and Microcenter, and come in different designs. The one in the photo is by OSEPP. If you are purchasing one, I suggest getting one that has a 3.3V/5.0V jumper or selector switch so that you can set it to supply the proper voltage for other projects. Do not worry if the adapter does not look exactly like the one in the photo.

When configured as described later in this article, **the HM-10 draws an average of about 0.18 mA**. The battery pack shown has a capacity of about 1800 mAh which would last 10,000 hours, or more than a year. If you want to build a very small iBeacon, you could use a single 240 mAh CR2032 coin cell which will last about 55 days. Battery holders for CR2032 coin cells are available. Note: if you are using alkaline cells, you must use a two-cell battery holder to keep the voltage within specifications (nominal  $1.5V \times 2 = 3.0V$ ).

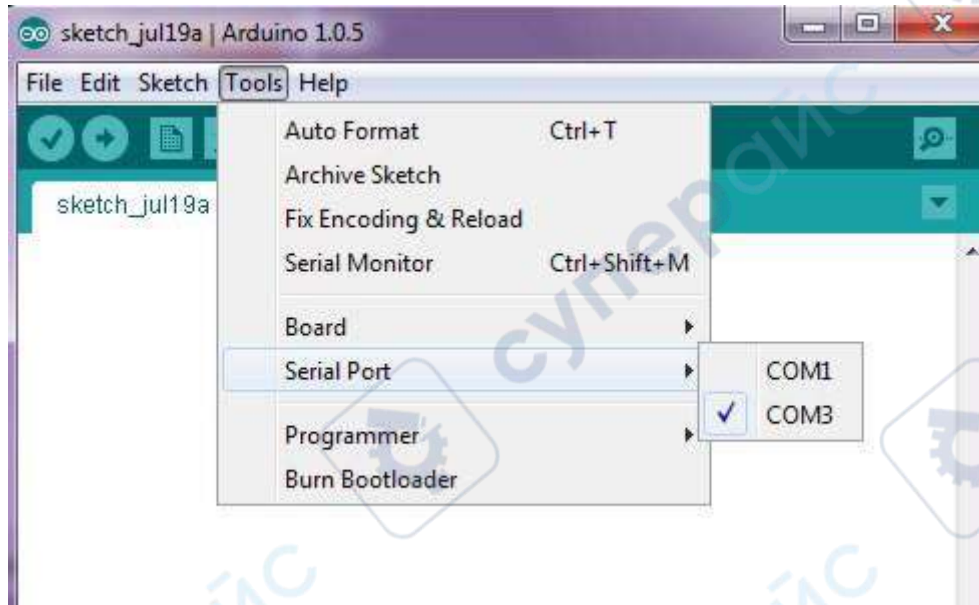


The diagram above shows the connections in detail. The battery is connected to pins 12 (+) and 13 (-) of the module. The RS232 adapter receive data (RXD) connects to pin 1, and transmit data (TXD) connects to pin 2. (Note: The transmit and receive signals to the HM-10 are crossed. This is correct.) The ground pin of the adapter connects to pin 14.

### Using a Terminal Emulator to Talk to the HM-10

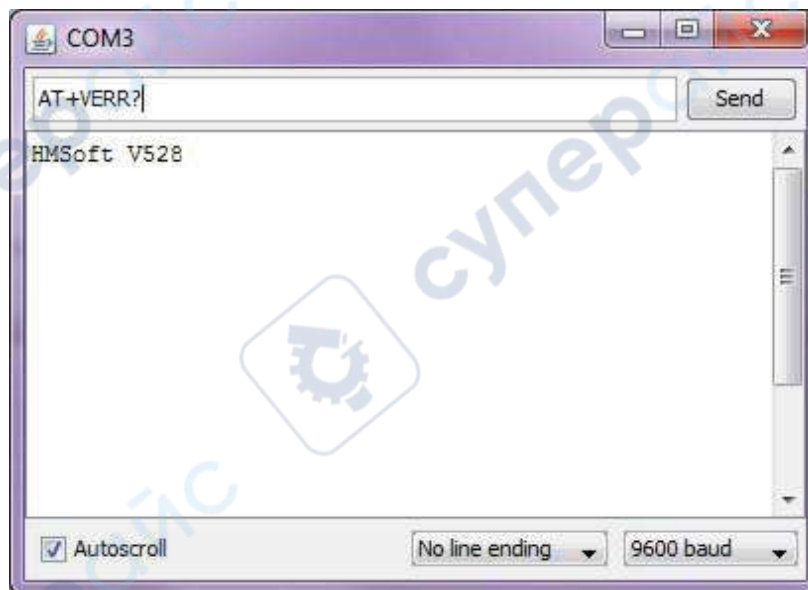
Once your hardware is ready you will want to be able to check if your HM-10 is working. With the battery and serial adapter connected to the HM-10, connect the USB cable of the adapter to your PC. If this is the first time you are using the adapter, you may be asked for a device driver for the module. In most cases the operating system will find the driver automatically (it may take a couple of minutes). If not, refer to the documentation that came with the device. If you can, take note of the COM port assigned to the adapter.

The HM-10 uses an AT command set that requires very unusual timing when manually typing in commands from the PC. The commands are not terminated by a carriage return or line feed, and rely on a very short delay after the command line has been entered to complete. Therefore I strongly recommend using the Arduino software serial monitor to talk to the HM-10. For most people reading this, the Arduino IDE is probably already installed on their PC. If it is not yet installed, please install version 1.0x (currently 1.05) available from <http://arduino.cc/en/Main/Software>. Once installed, start the program up.



Under the tools menu pull down, select Serial Port, and then select the COM port corresponding to your serial adapter. In most cases this will be the highest numbered COM port available. See the screenshot above where the COM port is COM3.

Next launch the serial monitor by selecting the menu item under tools. At the bottom menu, select "9600 Baud" and "No line ending". Now you can talk to the HM-10. Try typing in the command "AT" (without the quotation marks) followed by the Enter key. The HM-10 should respond with "OK". The HM-10 does not terminate its output with a new line, so all its responses will be strung one after another.



Next you can check the firmware version in your HM-10. Enter the command "AT+VERR?". If you recently bought the HM-10 it should be version 526 or later. If so, you can skip the Firmware Update section below, and go directly to the iBeacon Configuration section. If you want the latest anyway, go ahead and update the firmware.

## Firmware Update

If your HM-10 needs a firmware update, you must download the appropriate version from the JNHuaMao web site. Determine which TI chip your HM-10 is using. It will either be a CC2540 (older units) or CC2541 (newer units). You can look at the square chip on the HM-10 under (use a magnifier if needed) and you should be able to read the markings.

Go to the download site at [http://www.jnhuamao.cn/download\\_rom\\_en.asp?id=#](http://www.jnhuamao.cn/download_rom_en.asp?id=#)

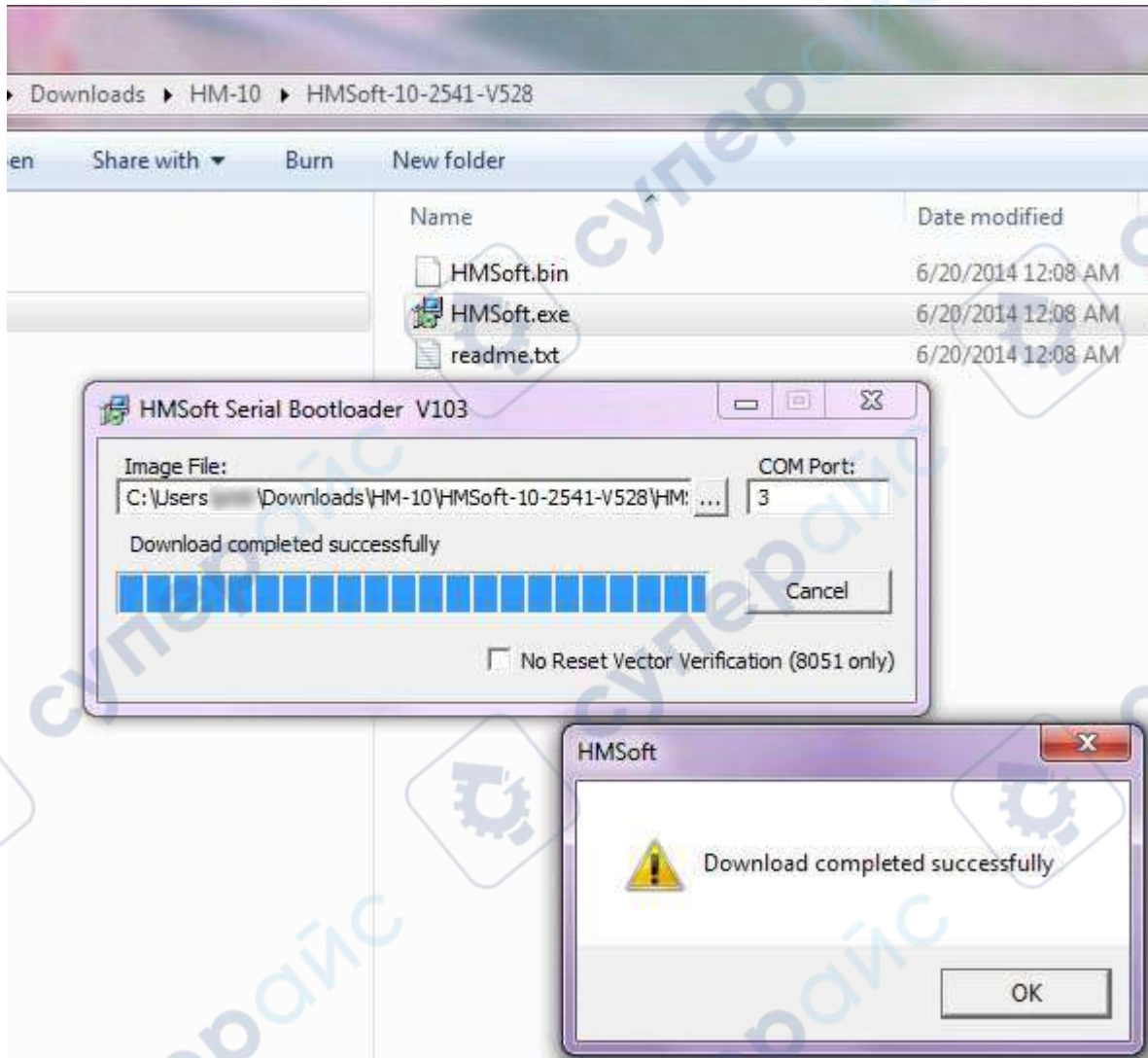
Make sure you are downloading the correct version of the firmware! That would be for the HM-10, and for the TI chip you have on your board.

Unzip the firmware update file, and place all the contents in the same folder. You will see at least 3 files. One will be a readme.txt file with brief instructions. The second will be the firmware file with a .bin suffix. The third file will be the firmware update application .exe file for Windows (sorry Linux and Mac users). It will probably be named HMSoft.exe.

Before updating the HM-10 firmware, make sure you have reliable connections to the module, and that the wires and cables cannot be accidentally knocked loose. Keep pets and small children from the area where you are doing the firmware update to avoid accidental interruption of the update.

1. Using the Arduino serial monitor enter the command **AT+SBLUP**  
The HM-10 should respond with OK+SBLUP. At this point the HM-10 is waiting for the firmware update.
2. **Exit the serial monitor program.**
3. Next, launch the firmware update program (the .exe file that you extracted from the zip file) most likely called HMSoft.exe.
4. In the COM Port field, enter the COM port number of the port connected to your HM-10. For example if it is COM3, enter **3** in the field.
5. Click on the "..." button for the Image File field, and select the .bin file name of the file that came in the zip file: HMSoft.bin.
6. Finally, click **on Load Image** button.
7. The firmware update will proceed and will take a couple of minutes. **Do not interrupt the update process.** Do not do anything else on the PC while the update is taking place to reduce the chances the process will be interrupted.

8. When verification is complete, it may take a few seconds before the **Download completed successfully** message is displayed.



The screenshot above shows an example of a successfully concluded firmware update. When finished, exit the firmware update program, and re-launch the Arduino serial monitor program.

### iBeacon Configuration

You have to type in a few commands to the HM-10 to configure it as an iBeacon. Use the Arduino serial monitor program to do this. In the following list, the bold initial text is the command you should type into the HM-10 and the rest is a comment on what it does. Each command will be acknowledged with an OK...

1. **AT+RENEW** Restores factory defaults
2. **AT+RESET** Reboot HM-10

3. **AT** Wait for OK
4. **AT+MARJ0x1234** Set iBeacon Major number to 0x1234 (hexadecimal)
5. **AT+MINO0xFA01** Set iBeacon Minor number to 0xFA01 (hexadecimal)
6. **AT+ADVI5** Set advertising interval to 5 (546.25 milliseconds)
7. **AT+NAMEDOPEY** Set HM-10 module name to DOPEY. Make this unique.
8. **AT+ADTY3** Make non-connectable (save power)
9. **AT+IBEA1** Enable iBeacon mode
10. **AT+DELO2** iBeacon broadcast-only (save power)
11. **AT+PWRM0** Enable auto-sleep. This reduces power from 8 to 0.18 mA
12. **AT+RESET** Reboot

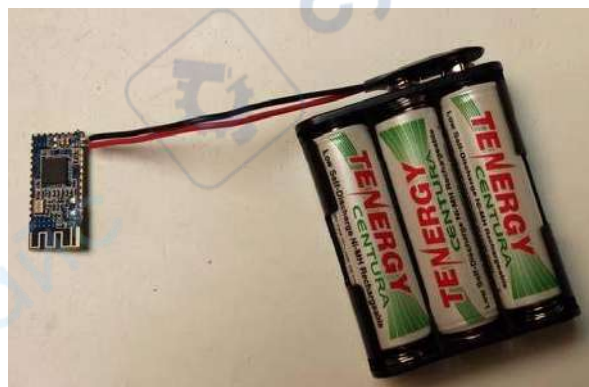
Note: The values of the parameters in **red** should be what you want them to be. The above are just examples.

After sending it this set of commands, the HM-10 should be visible on your iDevice or Android device (using you favorite iBeacon App) as an iBeacon. You can select the appropriate Major and Minor Numbers in steps 4 and 5. The Major number is the same in an area (e.g. a store or building) and the Minor number uniquely identifies the iBeacon. The above procedure does not alter the default HM-10 UUID which is a standard proximity UUID. If you want to change it, you may do so using the AT+IBE0, AT+IBE1, AT+IBE2 and AT+IBE3 commands. The 16 byte UUID is divided into 4 byte chunks and each one is altered with a different command. The table below illustrates which part of the UUID is updated with each of the 4 commands.

UUID: <b>74278BDA-B644-4520-8F0C-720EAF059935</b>			
<b>AT+IBE074278BDA</b>	<b>AT+IBE1B6444520</b>	<b>AT+IBE28F0C720E</b>	<b>AT+IBE3AF059935</b>

### Removing the Programming Cable

When you are convinced the HM-10 is configured the way you want, you can unsolder the connection to the serial adapter, and leave the power connector in place.

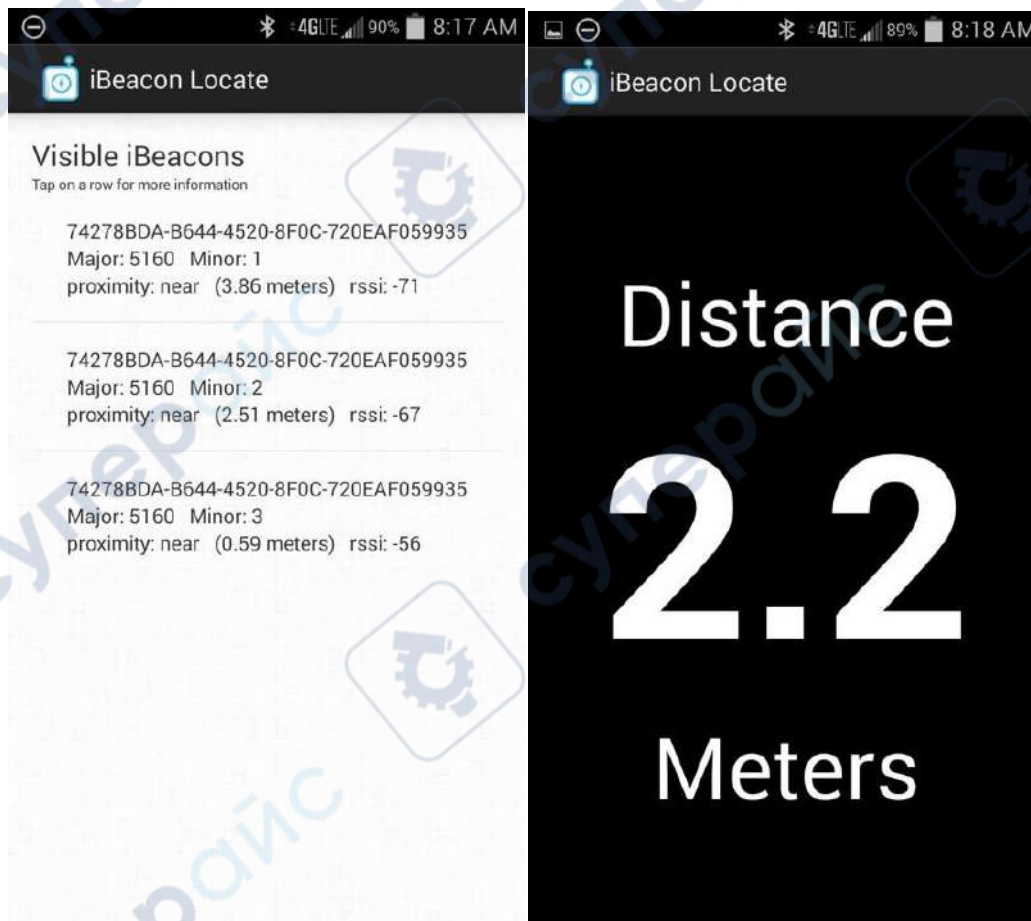


To protect the battery pack and module you should install the assembly in a case.



### Sample iBeacon Scan

The following screenshots show three of the HM-10 based iBeacons scanned by an Android phone, and the distance to one of the devices.





## Using the HM-10 as an iBeacon Proximity Device

The following eight UUIDs are built into Apple AirLocate as proximity devices. Use these to ensure compatibility with iOS apps that use AirLocate.

```
E2C56DB5-DFFB-48D2-B060-D0F5A71096E0  
5A4BCFCE-174E-4BAC-A814-092E77F6B7E5  
74278BDA-B644-4520-8F0C-720EAF059935      HM-10 Default  
112ebb9d-b8c9-4abd-9eb3-43578bf86a41  
22a17b43-552a-4482-865f-597d4c10bacc  
33d8e127-4e58-485b-bee7-266526d8ecb2  
44f506a4-b778-4c4e-8522-157aac0efabd  
552452fe-f374-47c4-bfad-9ea4165e1bd9
```

## Waking Up the HM-10

When the HM-10 is in auto-sleep mode, it will quickly go to sleep when powered up or rebooted. After sending it the **AT+PWRM0** command and it is restarted, the HM-10 will no longer respond to AT commands. To wake up the device, send it a long line of random alphabetic characters which is 80 characters or more. It will wake up and respond with **OK+WAKE**. If you want to work with the device for a while, send it an **AT+PWRM1** command so it does not sleep on you. You can then send it an **AT+PWRM0** when you are done to reduce power consumption.

## References and Web Links

1. iBeacon general article: <http://en.wikipedia.org/wiki/IBeacon>
2. HM-10 Firmware: [http://www.jnhuamao.cn/download\\_rom\\_en.asp?id=#](http://www.jnhuamao.cn/download_rom_en.asp?id=#)
3. Locate iBeacon for iOS  
<https://itunes.apple.com/us/app/locate-for-ibeacon/id738709014?mt=8>
4. iBeacon Scavenger Hunt  
<https://itunes.apple.com/us/app/ibeacon-savenger-hunt/id821092269?mt=8>  
<https://play.google.com/store/apps/details?id=com.radiusnetworks.scavengerhunt&hl=en>
5. iBeacon Scanner for Android  
<https://play.google.com/store/apps/details?id=kr.co.ixsoft.ibeacon.scanner&hl=en>