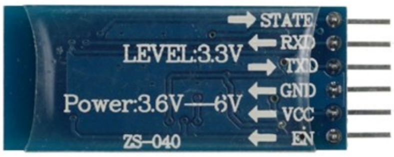
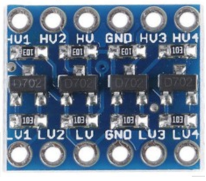
AR488 BlueTooth

Bluetooth is a common short range wireless connection technology for connecting devices and sending data. Bluetooth devices can typically communicate over distances not exceeding 10 metres, although some devices can communicate over greater distances. In order to communicate with each other, Bluetooth devices must be paired. Once paired, the device facilities can be discovered and a connection established.





The AR488 supports Bluetooth by means of the HC05 Bluetooth module for Arduino. Unlike the Arduino Uno or Nano boards, which operate at 5v DC which is common to all USB devices, the HC05 operates at 3.3v. However, HC05 modules can be purchased already mounted on a voltage converter host board which can be supplied with 5v. Such boards usually have 6 pins along one edge for easy connection to the Arduino. Although these boards can be supplied with 5v DC, there is usually no voltage conversion provided for the TXD and RXD pins which also require 3.3v signalling rather than 5v. Fortunately it is also possible to obtain a level shifter board at very little cost. These provide a means to shift the 5v to shift 5v logic to 3.3v and vice-versa. The board is easily supplied with 5v and 3.3v from the Arduino board. The Arduino TXD and RXD connections are made to the HV (5v) side, whereas the connections to the HC05 Bluetooth board are made to the LV (3.3v) side.

The HC05 module should not be confused with the similar HC06 module. While the HC05 module can operate as both slave and master, the HC06 module can operate in slave mode only. Serial communication speeds also seem to vary and while by default some boards operate at 38400 baud, others require a 9600 baud connection.

The HC05 board requires 5 connections in total DC power VCC, GND, TXD and RXD and a control pin (labelled EN) that allows the Arduino to set it in configuration mode. This is connected to pin 6 on the Uno or Nano. The pin marked STATE is left disconnected.

The level onverter board has four transistors across the centre. The 5v supply is connected to GND and HV, whereas the 3.3v supply goes to GND and the LV pin. The ground connections are common so it is not necessary to connect both of them. The 5v logic is connected to the HV side of the board and the 3.3v logic is connected to the LV side of the board. The board has four possible ‘channels’ with the 5v sdide marked HV1-4 and the 3.3v side correspondigly marked LV1-4.

The bluetooth HC05 board requires only two of these. Level shifting for the enable pin is accomplished using a pair of resistors as a voltage divider. It was found that when using the level shifter board, when pin 6 was low, sufficient voltage remained on the EN pin even with a pull-down resistor in place, to prevent the module from switching back to user mode.

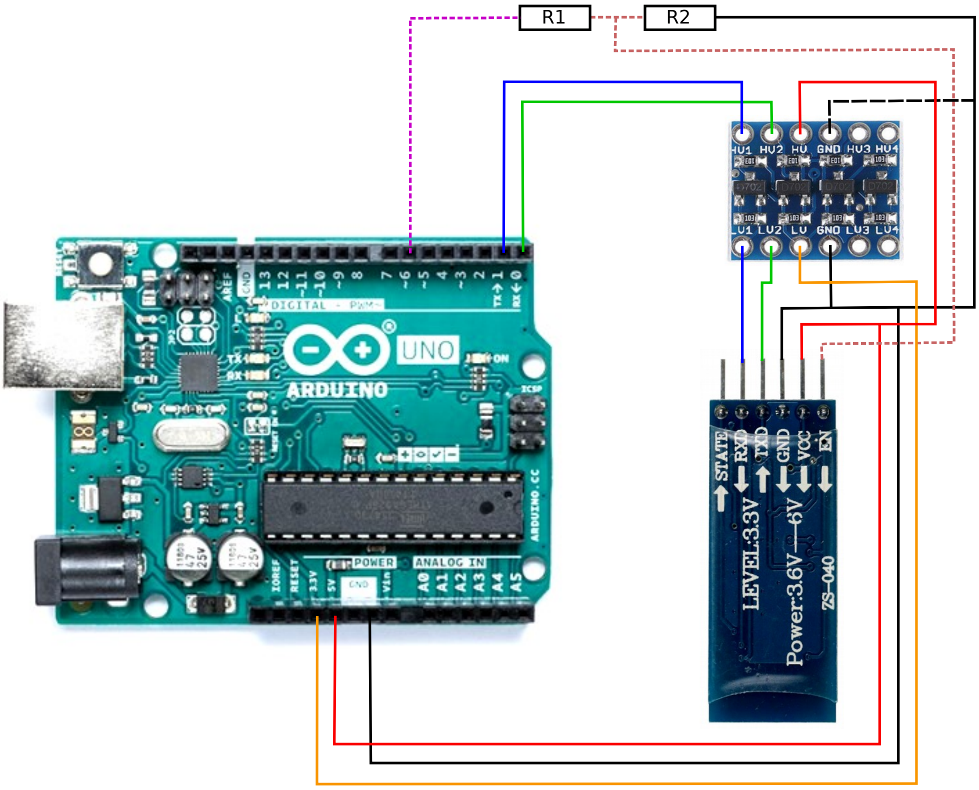


Diagram of the Arduino to HC05 connections

Enabling Bluetooth on the AR488

The Bluetooth feature is disabled by default, but can be enabled by removing the surrounding comments from the lines within the “Bluetooth Support” section.:

```
// Bluetooth support
/*
#define AR_BT_EN 6             // Bluetooth control enable pin
#define AR_BT_NAME "AR488-BT"  // Bluetooth device name
#define AR_BT_BAUD "115200"    // Bluetooth module baud rate
#define AR_BT_CODE "488488"    // Bluetooth pairing code
#define LED 13
*/
// Bluetooth support
```

and changing them to:

```
// Bluetooth support
#define AR_BT_EN 6            // Bluetooth control enable pin
#define AR_BT_NAME "AR488-BT" // Bluetooth device name
#define AR_BT_BAUD "115200"   // Bluetooth module baud rate
#define AR_BT_CODE "488488"   // Bluetooth pairing code
#define LED 13
// Bluetooth support
```

The default baud rate can be set to anything that the Arduino and the HC05 board will support.

The pairing code should also be uncommented by changing the line:

//#define AR_BTPN “488488”

to:

#define AR_BTPN “488488”

The default paring code is 488488, but this can be changed to any 6 digit number.

Upload the amended sketch prior to connecting the HC05 Bluetooth module, or else disconnect the HC05 Bluetooth module temporarily. Since the UNO and Nano have only one serial port, it is not possible to use Bluetooth and the USB at the same time. Attempting to upload the sketch while the HC05 board is connected to Rx and Tx will result in an error. If this should happen, disconnect the Tx and Rx connections to the HC05 board and try again. Once Bluetooth is set up, it will still be possible to power the board via USB or using the DC input jack on the UNO but it is not possible to communicate via both USB and BlueTooth simmultaneously.

Setting up

Once the sketch has been configured and uploaded to the board, connect up the HC05 Bluetooth module as per the diagram and upload the amended sketch.

Once connected, the AR488 can be powered up and on start-up will automatically detect the baud rate and configure the HC05 module. The AR488 will appear as a Bluetooth device called AR488-BT.

The first time that the HC05 board is used, the LED on the HC05 board will blink slowly for about 10 seconds. While this is happening, the LED on the Arduino should blink twice to indicate that the baud rate has been successfully detected, and then 3 times to indicate that configuration has been successful. A few moments after this, the LED on the HC05 will begin flashing rapidly. At this point, the HC05 is ready for pairing.

Once the HC05 board has been configured, on subsequent power up, the LED on the Arduino will blink twice to indicate baud rate detection, and then just once to indicate that connection has been established. It will then switch into pairing mode and the LED on the HC05 will flash rapidly until connected to a device, such as a PC.

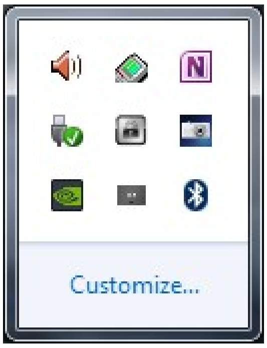
If the pairing code is subsequently changed, the AR488 will perform the auto-configuration process again.

Once paired, the LED on the HC05 board will blink twice every few seconds.

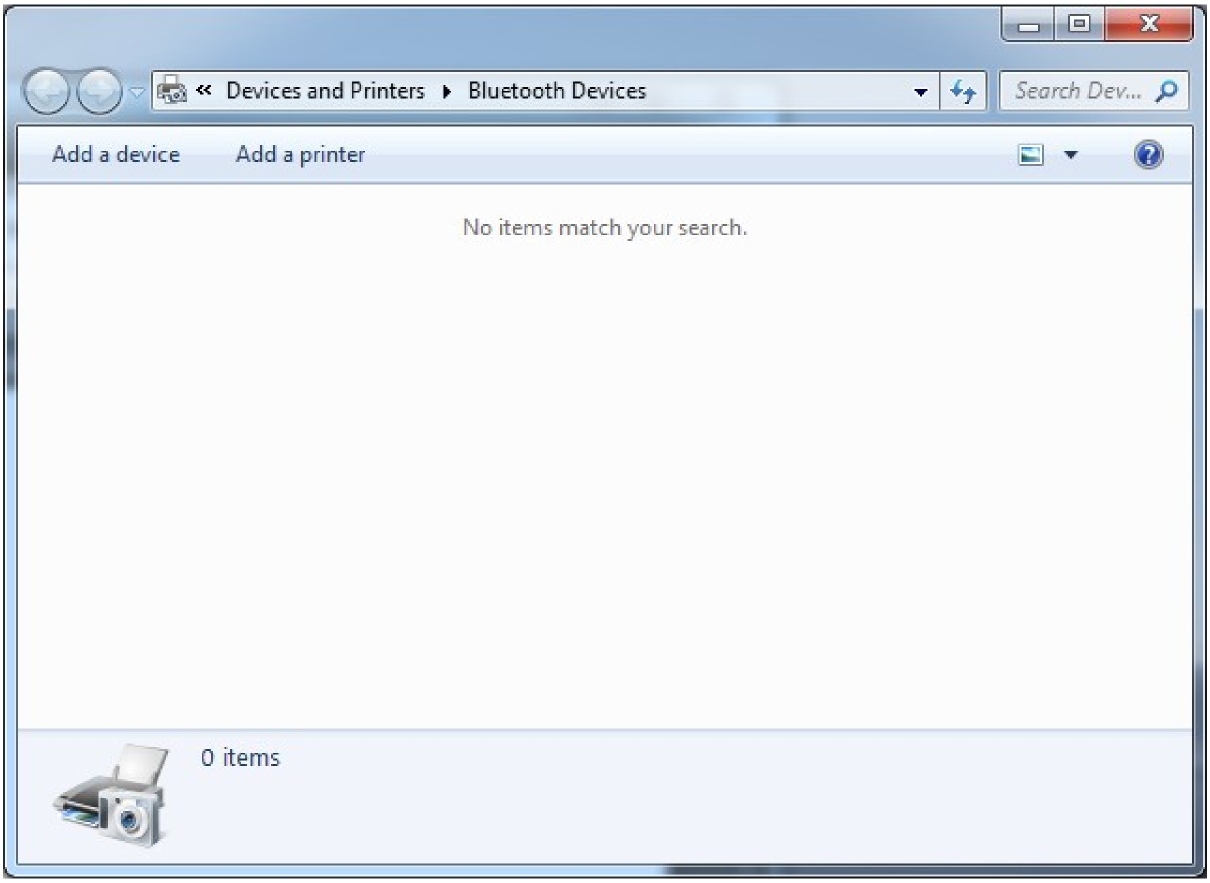
Pairing the AR488 Interface with Bluetooth

Windows 7

Go to the task bar and from the hidden icons panel select the Bluetooth icon:



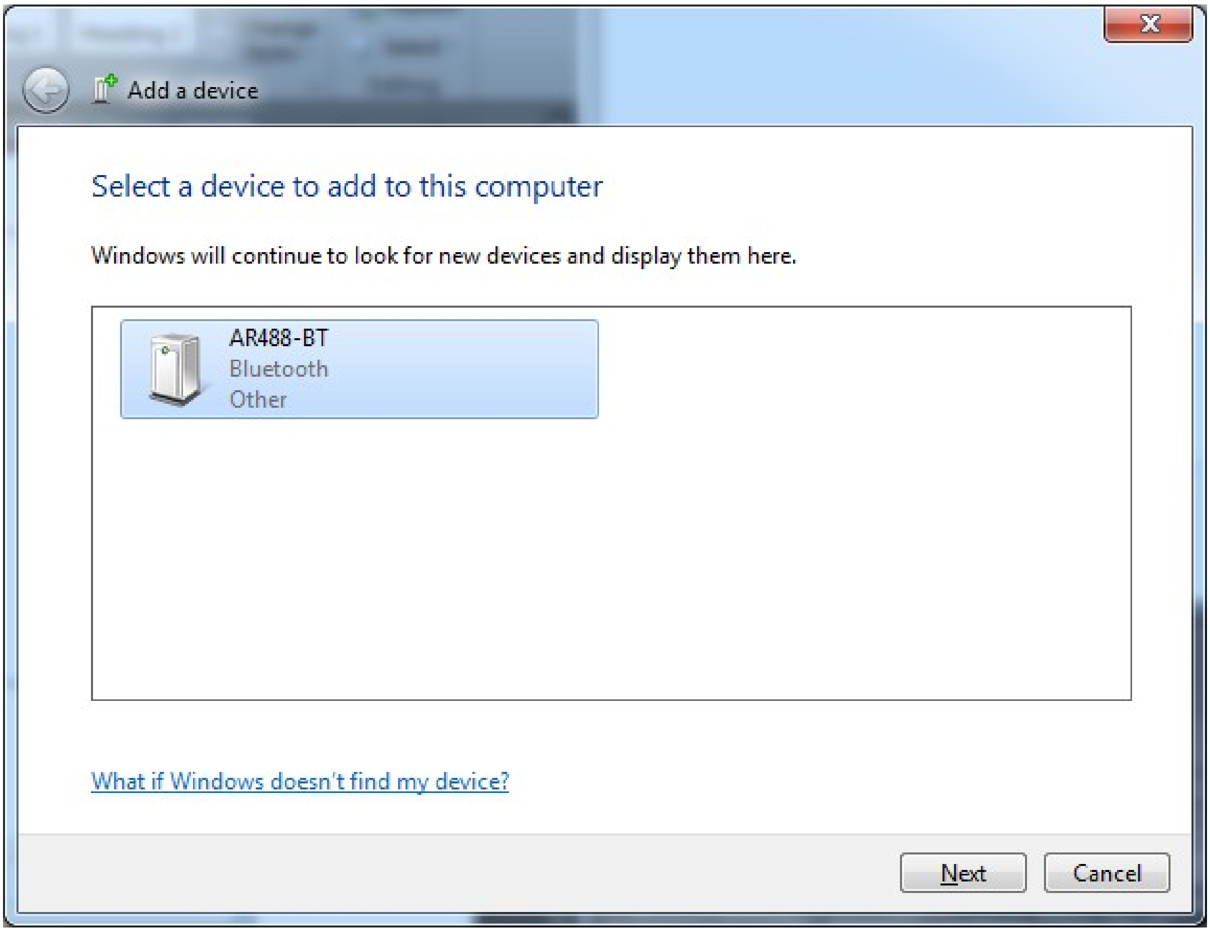
Click *Show Bluetooth Devices*



No items match your search.

0 items

This will open a windows showing your Bluetooth devices. If you have no other devices connected, the window will be empty. Click *Add a Device*


Add a device

Select a device to add to this computer

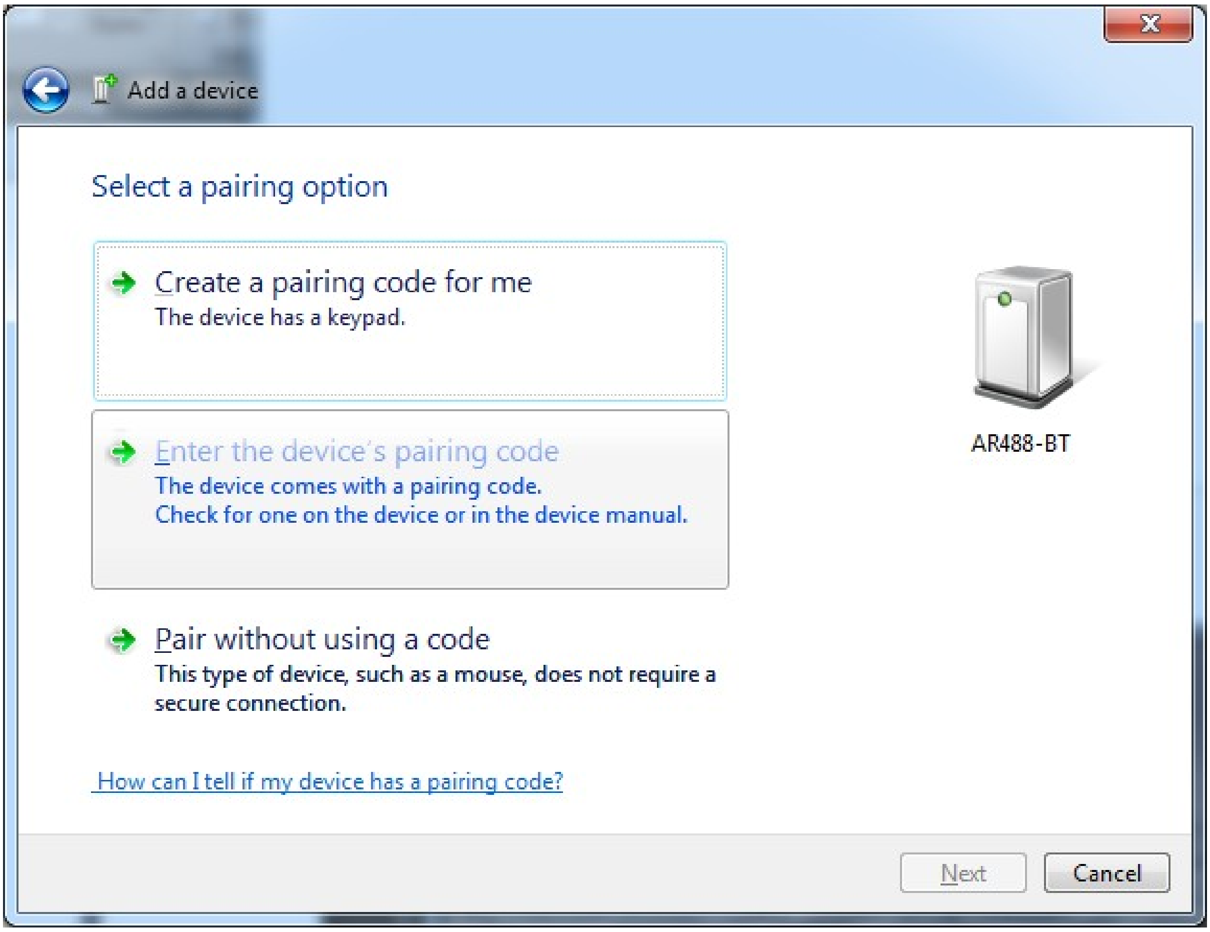
Windows will continue to look for new devices and display them here.

AR488-BT
Bluetooth
Other

What if Windows doesn't find my device?

After a few moments, the AR488 -BT device should be detected. Please not that it may initially show simply as *Other*, but when you click on it, the name should be updated to AR488-BT.

Click the *Next* button. This will momentarily show a *Connecting to devices* status and then open the *Add a Device* window:



Add a device

Select a pairing option

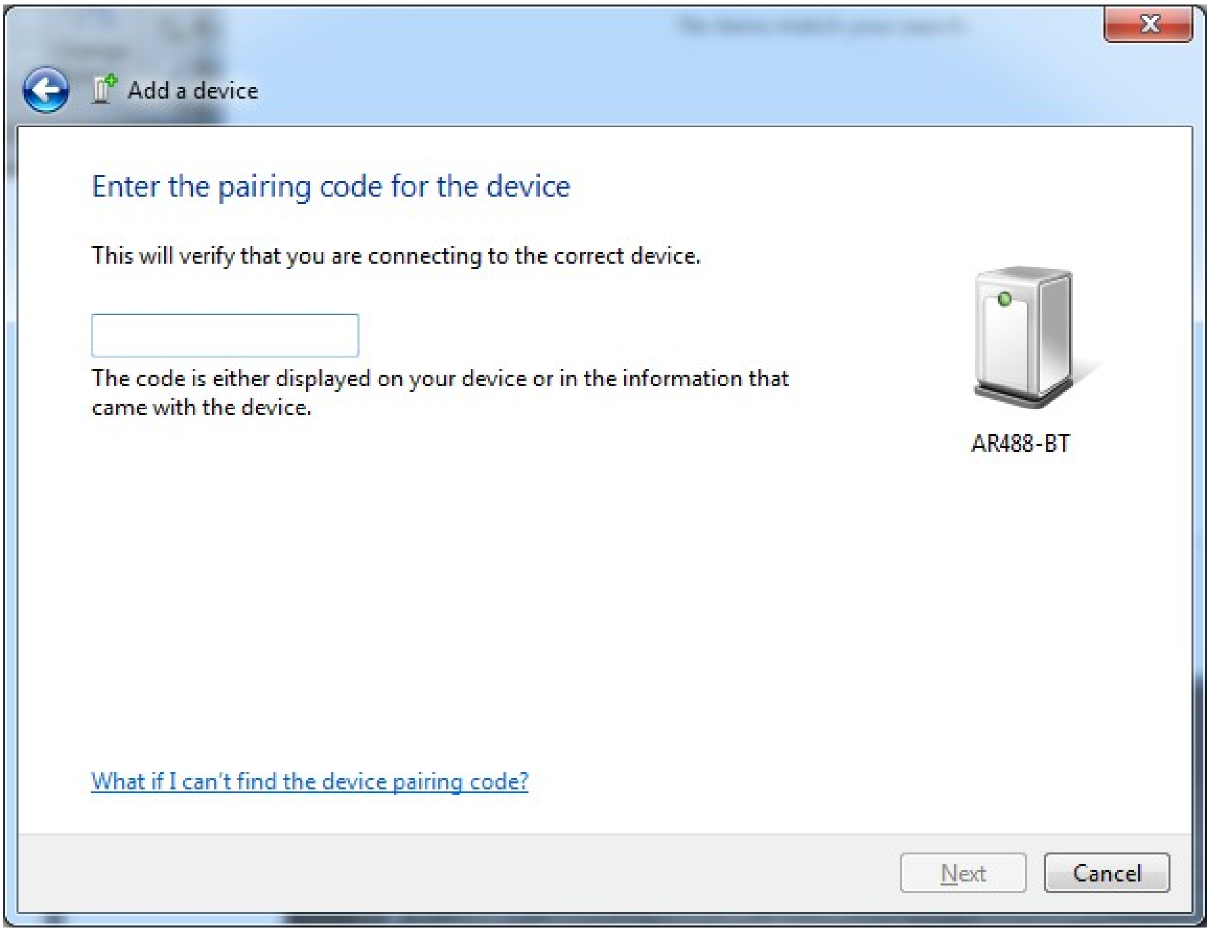
Create a pairing code for me
The device has a keypad.

Enter the device's pairing code
The device comes with a pairing code.
Check for one on the device or in the device manual.

Pair without using a code
This type of device, such as a mouse, does not require a secure connection.

How can I tell if my device has a pairing code?

Select *Enter the device's pairing code* panel. Te next dialog will prompt you to enter the device pairing code:



Add a device

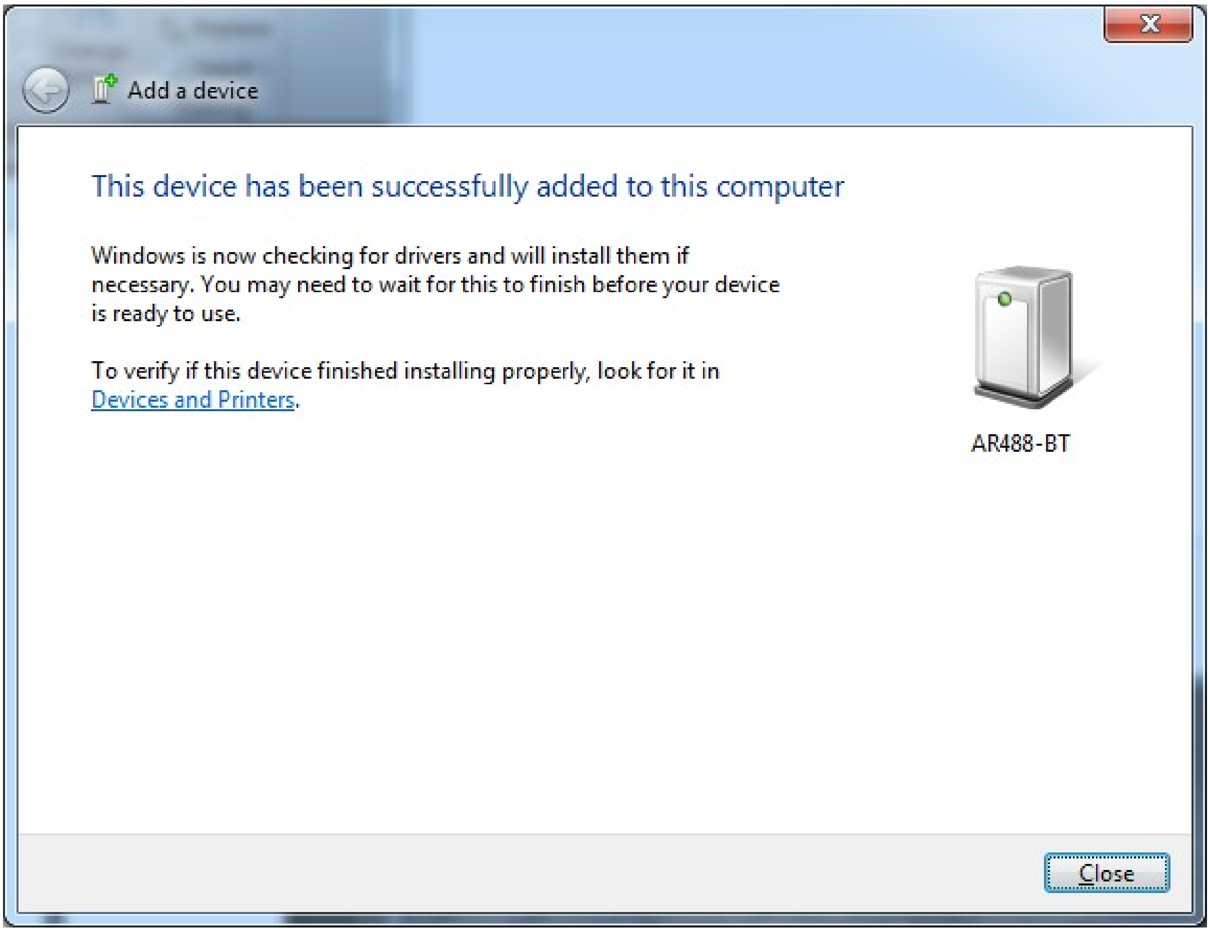
Enter the pairing code for the device

This will verify that you are connecting to the correct device.

The code is either displayed on your device or in the information that came with the device.

Enter the configured pin code, for example *488488* and then click *Next*.

Windows should now connect the device and install a virtual serial port. A configmation should now be displayed to indicate that the device has been successfully paired:

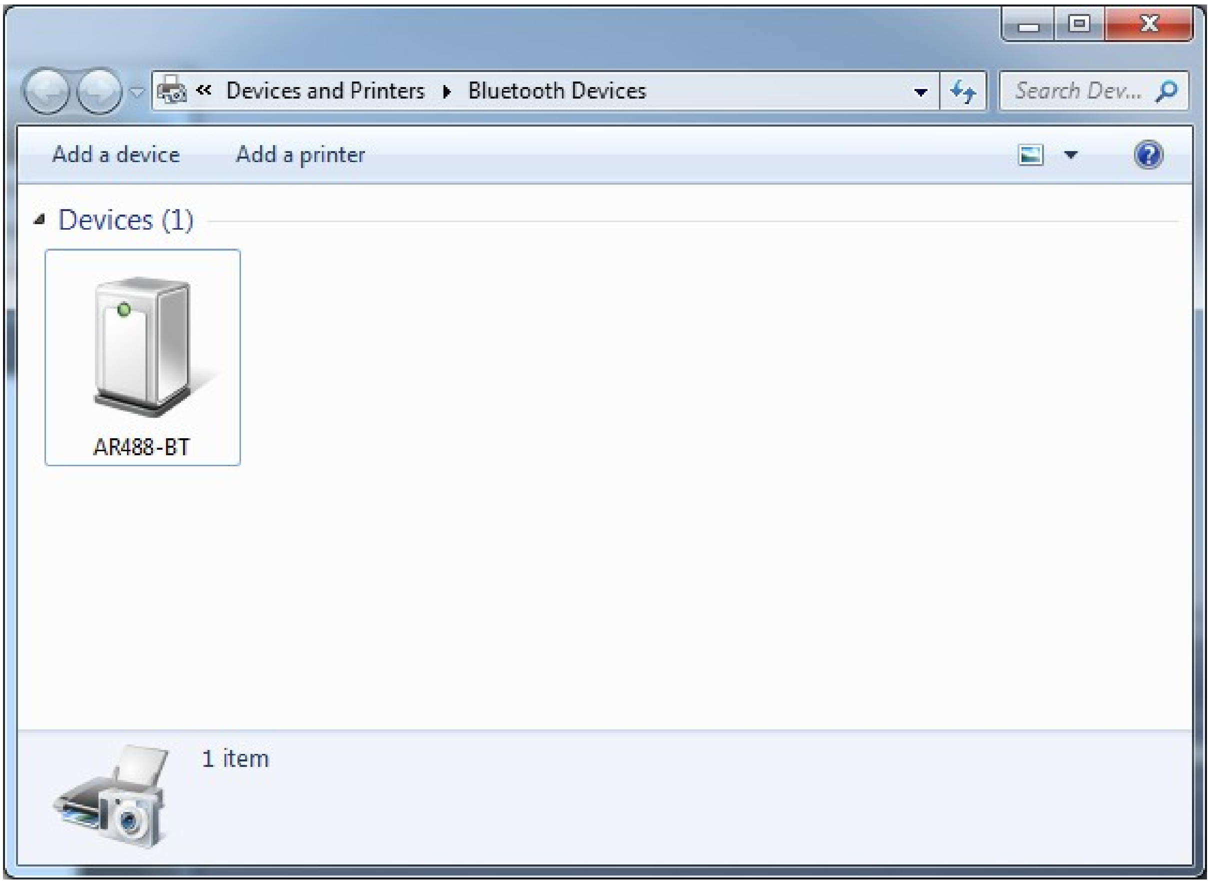


This device has been successfully added to this computer

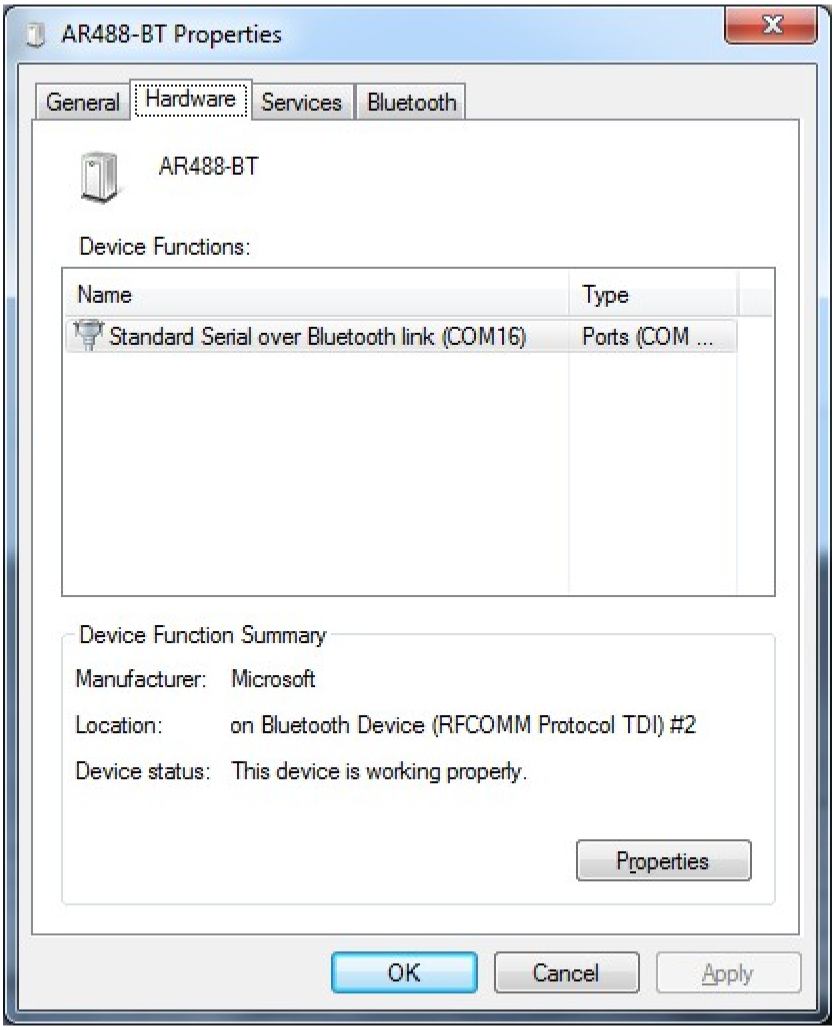
Windows is now checking for drivers and will install them if necessary. You may need to wait for this to finish before your device is ready to use.

To verify if this device finished installing properly, look for it in Devices and Printers.

After clicking *Close*, you should now see the connected device:



Double click on the device and open the *Hardware* tab:



AR488-BT Properties

AR488-BT

Device Functions:

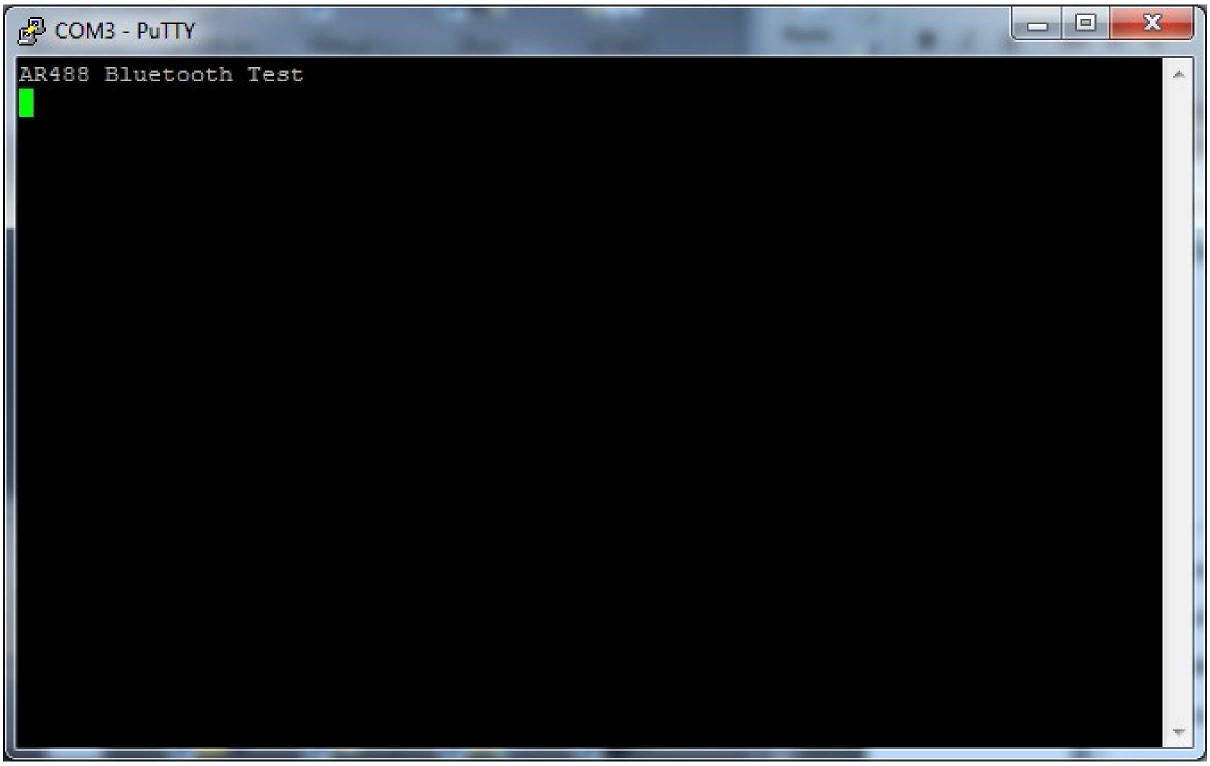
Name	Type
Standard Serial over Bluetooth link (COM16)	Ports (COM ...

Device Function Summary

Manufacturer:	Microsoft
Location:	on Bluetooth Device (RFCOMM Protocol TDI) #2
Device status:	This device is working properly.

This should display the virtual serial port that has been assigned to the device, e.g.in the above example it is COM16.

Open a terminal session to the device using the baud rate that was configured in the sketch:

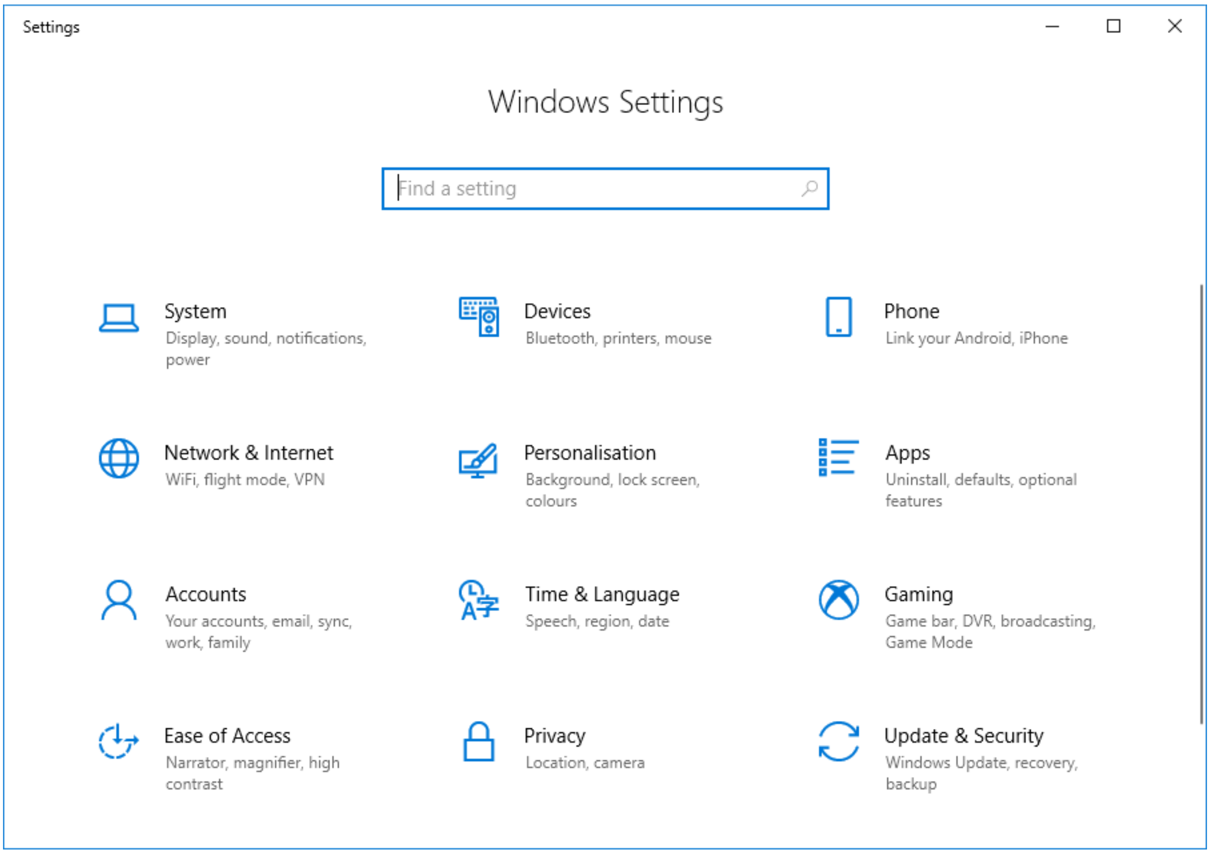


```
AR488 Bluetooth Test
```

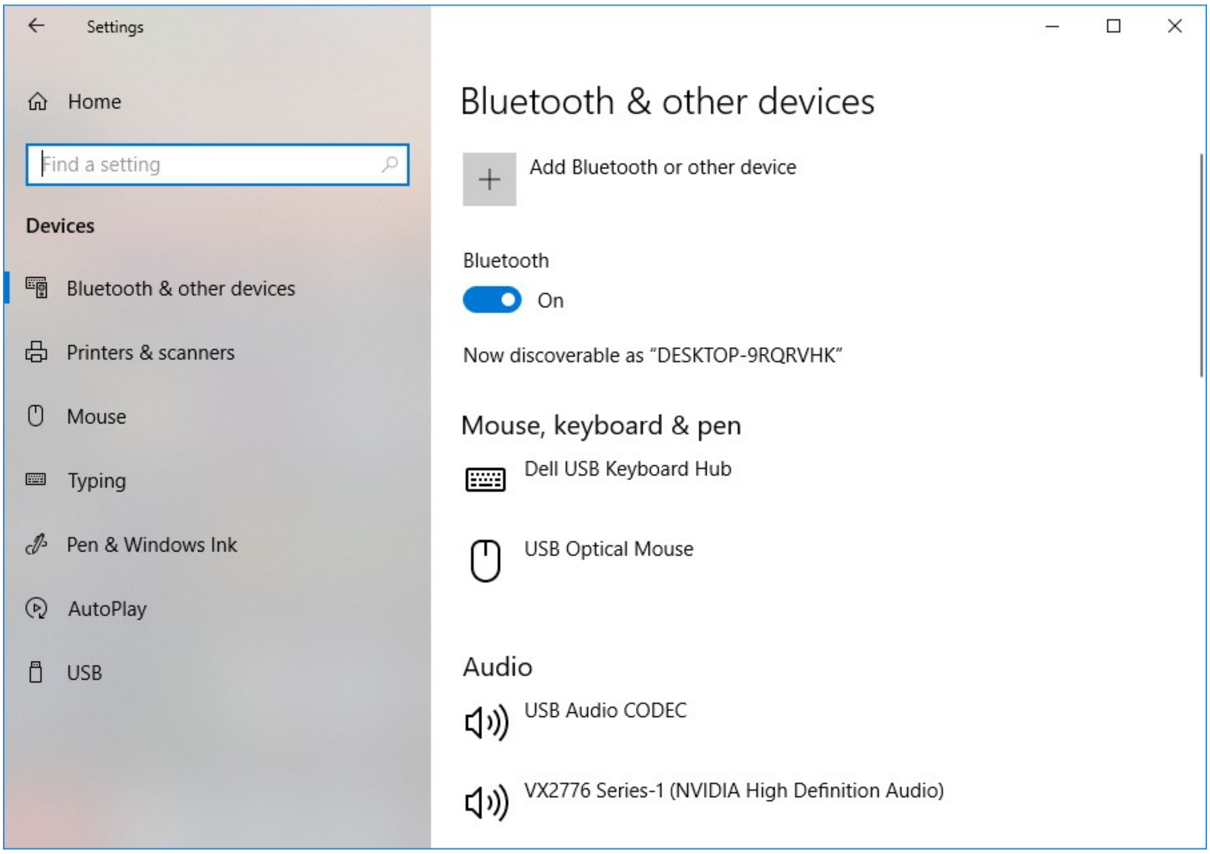
The interface should respond as normal.

Windows 10

Go to *Windows Settings* (cog icon on the left of the windows menu) and select *Devices*:


Settings
Windows Settings
Find a setting
System
Display, sound, notifications, power
Devices
Bluetooth, printers, mouse
Phone
Link your Android, iPhone
Network & Internet
WiFi, flight mode, VPN
Personalisation
Background, lock screen, colours
Apps
Uninstall, defaults, optional features
Accounts
Your accounts, email, sync, work, family
Time & Language
Speech, region, date
Gaming
Game bar, DVR, broadcasting, Game Mode
Ease of Access
Narrator, magnifier, high contrast
Privacy
Location, camera
Update & Security
Windows Update, recovery, backup

This opens the *Bluetooth & other devices* dialog:



Bluetooth & other devices

Bluetooth

On

Now discoverable as "DESKTOP-9RQRVHK"

Mouse, keyboard & pen

Dell USB Keyboard Hub

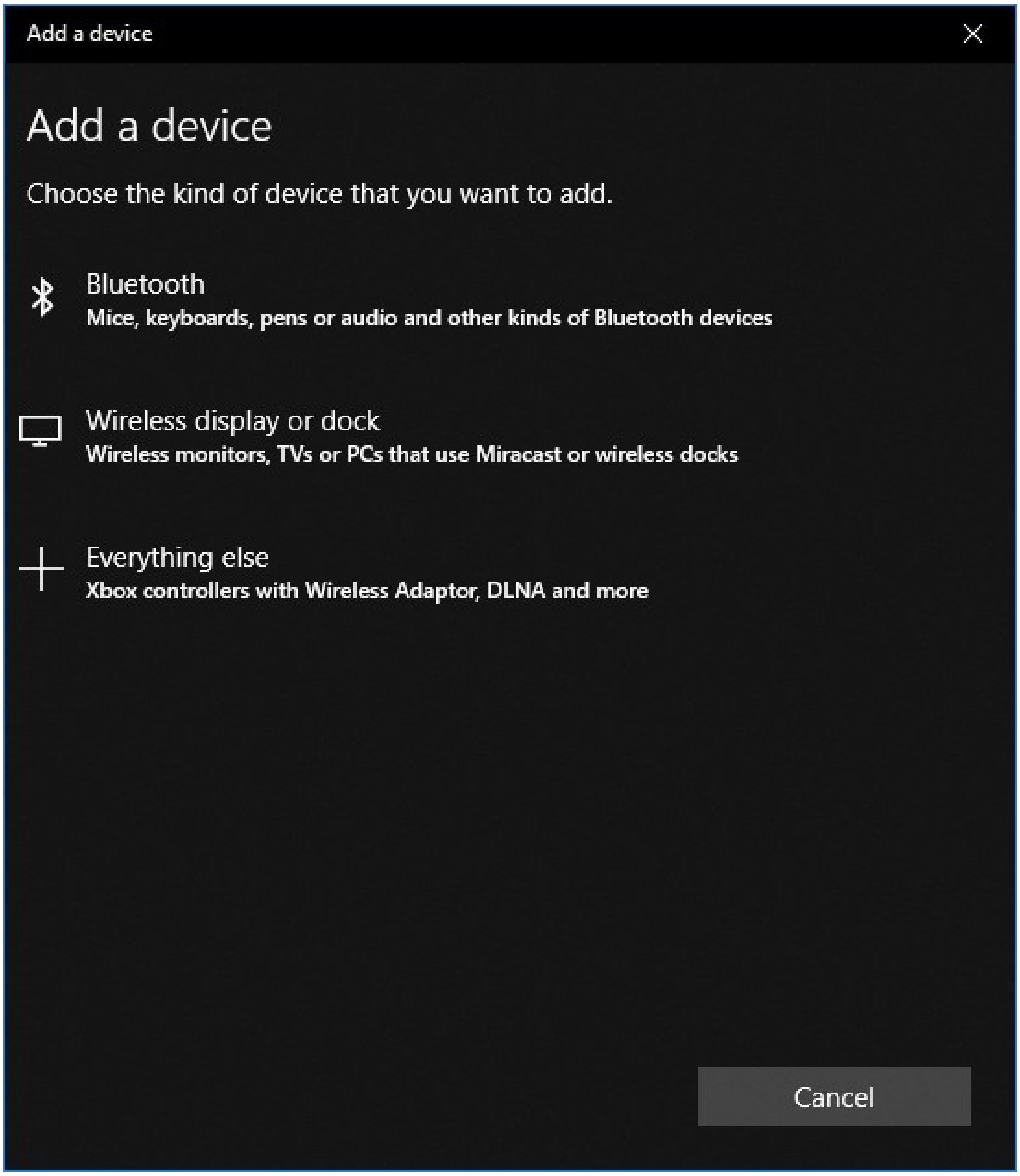
USB Optical Mouse

Audio

USB Audio CODEC

VX2776 Series-1 (NVIDIA High Definition Audio)

Make sure that Bluetooth is turned on. Click *Add Bluetooth or other device*. This will open another Window:



Add a device

Add a device

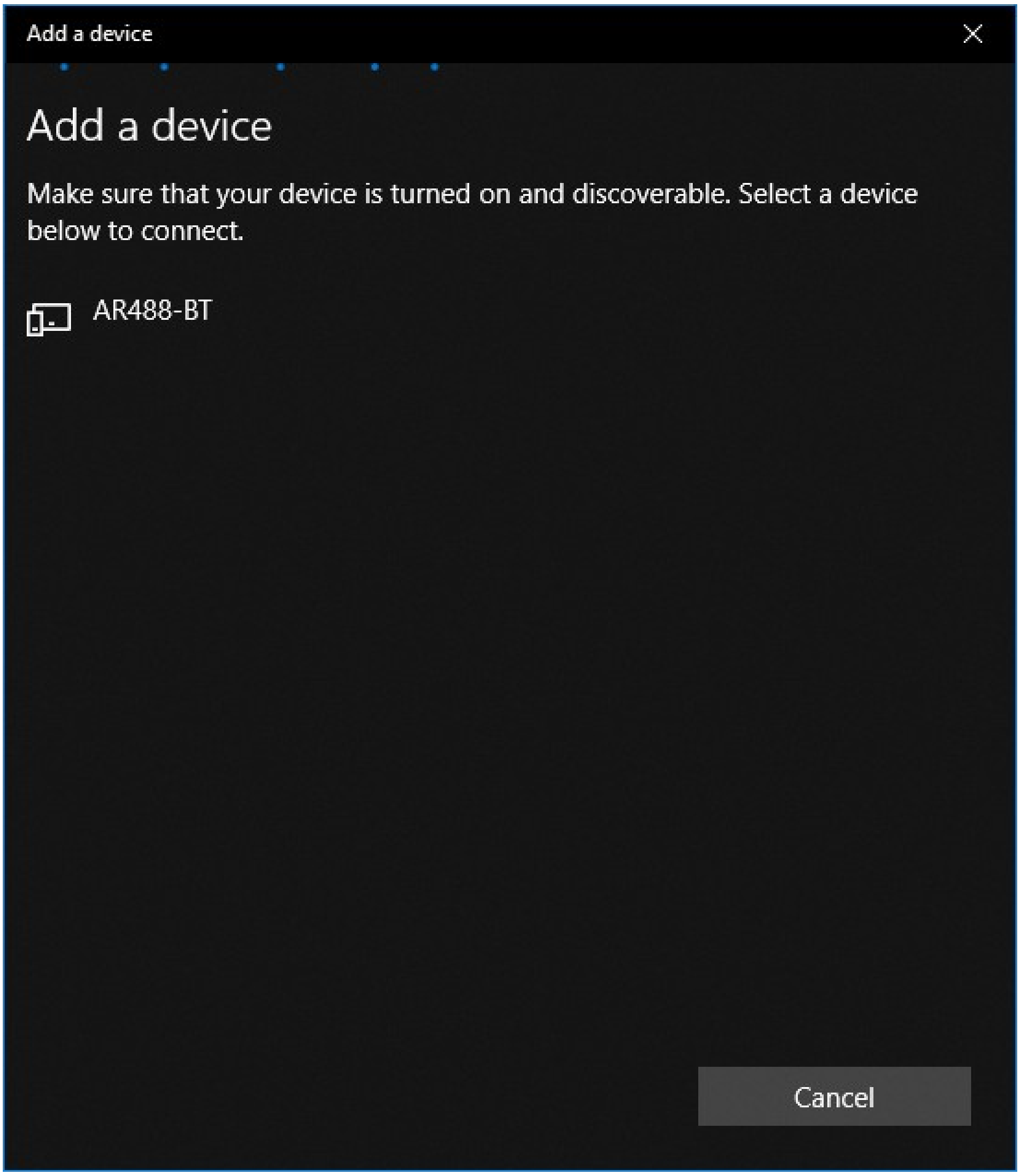
Choose the kind of device that you want to add.

Bluetooth
Mice, keyboards, pens or audio and other kinds of Bluetooth devices

Wireless display or dock
Wireless monitors, TVs or PCs that use Miracast or wireless docks

Everything else
Xbox controllers with Wireless Adaptor, DLNA and more

Click *Bluetooth*. Windows should now look for devices. It should momentarily show the AR488-BT device as *unknown device*, but this should quickly change to *AR488-BT*.

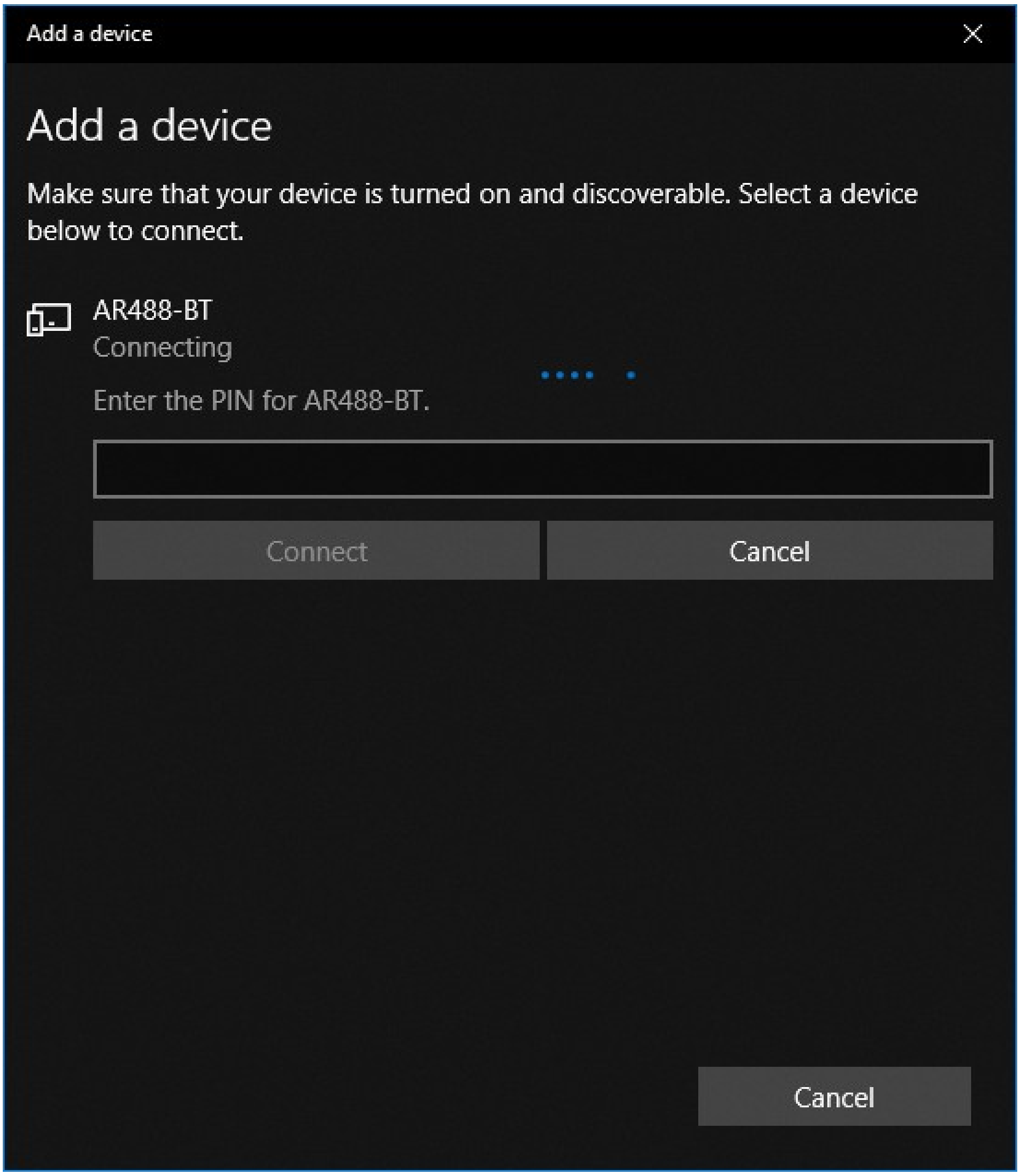


Add a device

Add a device

Make sure that your device is turned on and discoverable. Select a device below to connect.

Click the AR488-BT device. After a few moments prompt will appear requesting the pin.



Add a device

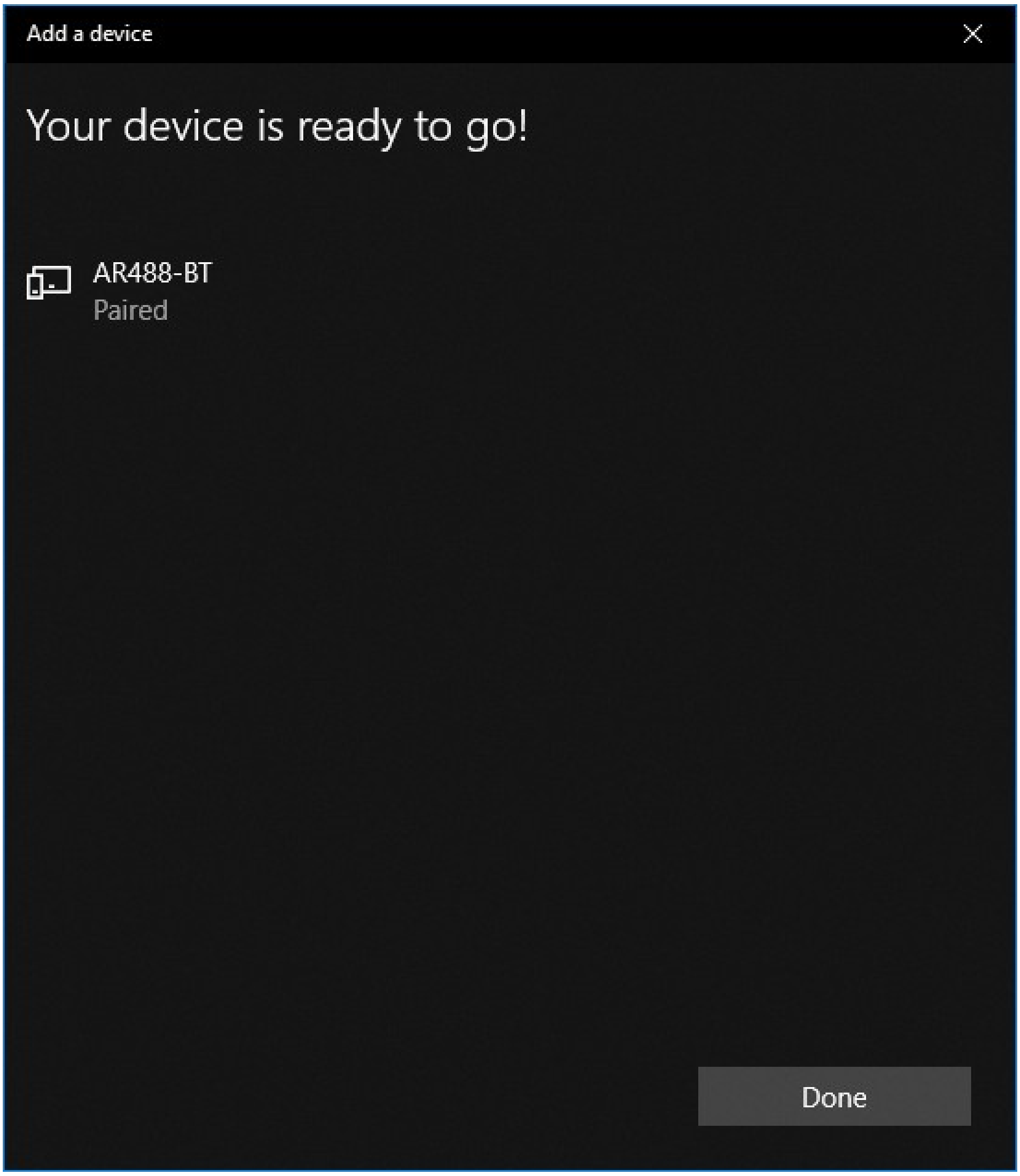
Add a device

Make sure that your device is turned on and discoverable. Select a device below to connect.

AR488-BT
Connecting

Enter the PIN for AR488-BT.

Enter the pin and click *Connect*. If it times out, the dialog may show Try connecting your device again. Click on the device again to try once more. Once successful, this will be configmed:



Add a device

Your device is ready to go!

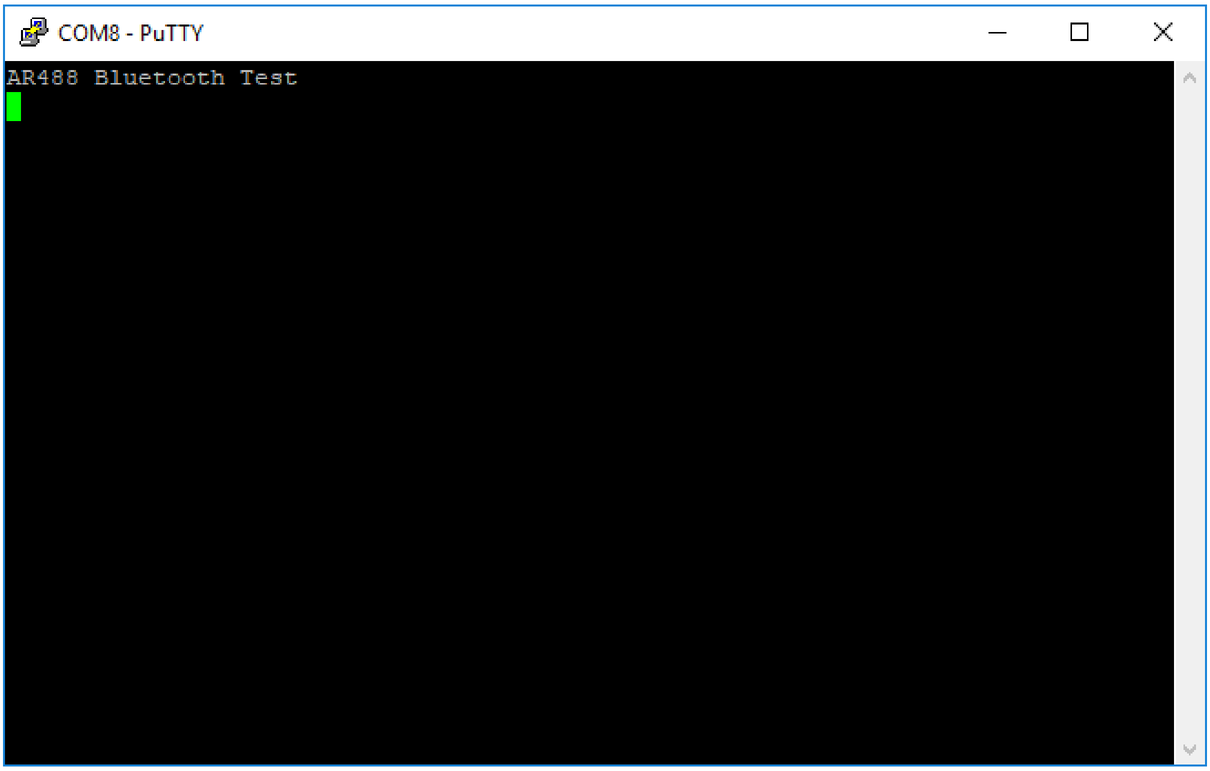
AR488-BT
Paired

The device status should now be shown as *Paired*. Click the Done button to close device configuration window.

Back on the *Bluetooth & other devices* window, scroll down. The AR488-BT device should be listed under *Other Devices*. The window can now be closed.

Right-click This PC and select Manage. Click on Device Manager and expand the Port (COM & LPT) section. The device should be shown as Standard Serial over Bluetooth link (COMx) where COMx will be the assigned COM port.

Open a terminal session to the assigned COM port and test communication with the device:



```
AR488 Bluetooth Test
```

Linux MINT

These instructions should work on Linux Mint 19 and above and probably Ubuntu. Depending on your distro of Linux, you may need to download the *bluez* or *bluez5* tools package or compile it from source. On Linux Mint 19 9and likely Ubuntu) this will be installed by default or can be readily downloaded from the repository using *apt or Software Manager*.

Make sure that your Bluetooth dongle or built-in device is working correctly on your computer or laptop.

Make sure that Bluetooth is turned on and your system can identify your Bluetooth hardware.

Once you have configmed that your Bluetooth hardware is working, open a terminal and at the command prompt type:

```
% bluetoothctl
```

This should list any known bluetooth devices, confirm that the Agent is registered and open a `[bluetooth]#` prompt:

```
$ bluetoothctl
[NEW] Controller 00:80:98:94:AB:7E agabus [default]
[NEW] Device 78:3A:84:93:BC:B9 iPad
[NEW] Device 10:2F:6B:BD:49:F1 N930 phone
Agent registered
[AR488-BT]#
```

At the prompt type:

```
scan on
```

This should initiate a scan for new devices. In a few seconds any new devices should be listed:

```
[bluetooth]# scan on
Discovery started
[CHG] Controller 00:80:98:94:AB:7E Discovering: yes
[NEW] Device 98:D3:31:F9:4E:6D AR488-BT
```

The AR488-BT device should be detected and its mac address listed.

To pair the device type:

```
pair 98:D3:31:F9:4E:6D
```

where the MAC address is the address of YOUR bluetooth device. Bluetoothctl should respond with:

```
[bluetooth]# pair 98:D3:31:F9:4E:6D
Attempting to pair with 98:D3:31:F9:4E:6D
[CHG] Device 98:D3:31:F9:4E:6D Connected: yes
Request PIN code
[AR481m[agent] Enter PIN code:
```

Enter the pairing code of the HC05 deivce. The default pairing code is 488488, but a custom six digit code can be specified within the sketch. Bluetoothctl will now attempt to pair with the device. If sucessful, the output should be something like:

```
[AR481m[agent] Enter PIN code: 488488
[CHG] Device 98:D3:31:F9:4E:6D UUIDs: 00001101-0000-1000-8000-00805f9b34fb
[CHG] Device 98:D3:31:F9:4E:6D ServicesResolved: yes
[CHG] Device 98:D3:31:F9:4E:6D Paired: yes
Pairing successful
[AR488-BT]#
```

At this point the device is paired, but we do not yet have a serial port to connect with. We now need to use another tool to create one. First exit bluetoothctl by typeing:

```
[bluetooth]# exit
```

The utiltity will respond with the following and return to the system prompt:

```
Agent unregistered
[DEL] Controller 00:80:98:94:AB:7E agabus [default]
$
```

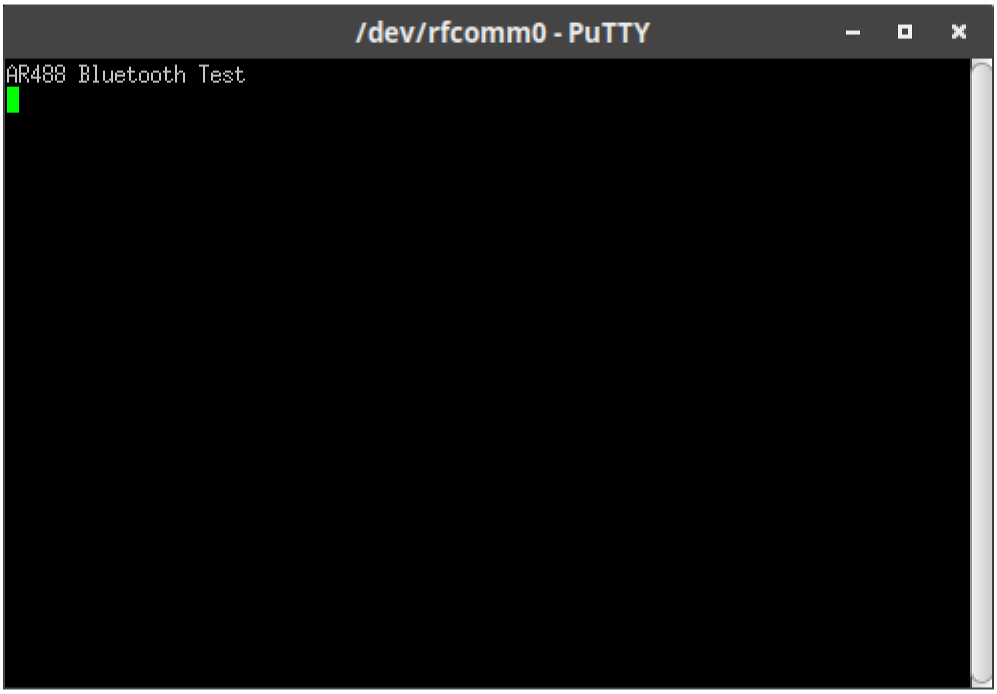
Type the following to create the /dev/rfcomm0 device:

```
sudo rfcomm bind 0 98:D3:31:F9:4E:6D 1
```

You may be prompted for the sudo password. That the port binding has been established can be verified with:

```
$ ls /dev/rfcomm*
/dev/rfcomm0
```

Now that the port is available, we can open a tty terminal such as PuTTY:



```
AR488 Bluetooth Test
```

Further information:

To unbind the /dev/rfcomm0 port, from the system `$` prompt type:

sudo rfcomm unbind 0 98:D3:31:F9:4E:6D 1

To get further information about a device from the `[bluetooth]#` prompt type:

info 98:D3:31:F9:4E:6D

where the MAC address is the address of the device you would like further information on. The program should respond with something like this:

```
[AR488-BT]# info 98:D3:31:F9:4E:6D
Device 98:D3:31:F9:4E:6D (public)
      Name: AR488-BT
      Alias: AR488-BT
      Class: 0x00001f00
      Paired: yes
      Trusted: no
      Blocked: no
      Connected: yes
      LegacyPairing: yes
      UUID: Serial Port               (00001101-0000-1000-8000-
00805f9b34fb)
[AR488-BT]#
```

You can also list all the devices you have paired:

```
[AR488-BT]# paired-devices
Device 98:D3:31:F9:4E:6D AR488-BT
Device 10:2F:6B:BD:49:F1 N930 phone
[AR488-BT]#
```

To remove a paired device type:

```
[bluetooth]# remove 98:D3:31:F9:4E:6D
[DEL] Device 98:D3:31:F9:4E:6D AR488-BT
Device has been removed
[bluetooth]#
```