

RTMS - Software Setup

These instructions are for setting up the RTMS (Robot Tracking & Management System) software. This software will run on your PC/MAC and will be used for various labs in order to allow the robot to communicate with your laptop as well as display important information. It allows you to connect the laptop to the robot through Bluetooth. You MUST get your bluetooth working in order to do the labs.

Download the following code for use on the robot:

- [TestRTMS.java](#) - connects to your laptop, gets a Pose from the RTMS and displays it on the robot
- [RBtoPCcom.java](#) - allows data to be sent to/from the PC via bluetooth.
- [Pose.java](#) - represents a position (i.e., x, y and angle) of the robot.

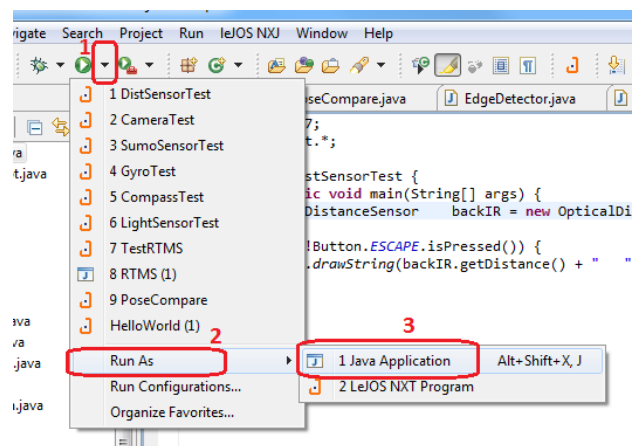
When you run the program, it will wait for the user to press **Connect** within the RTMS software (explained below). Then it will wait for the user to press **Start** within the RTMS software. It will then request the robot's position from the RTMS and display it on the screen for 5 seconds before closing.

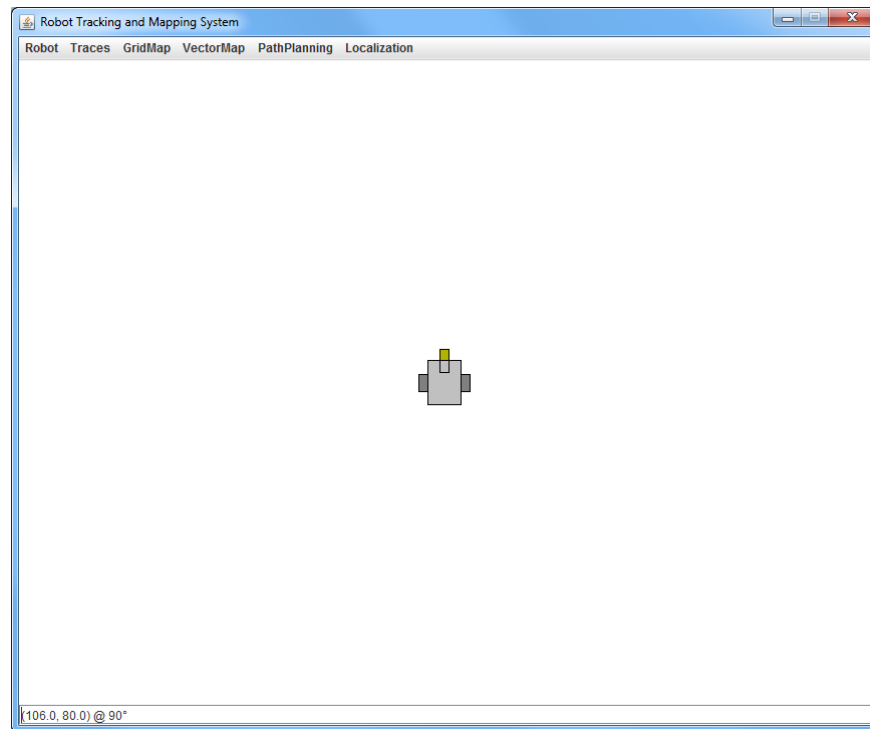
You will need to download the RTMS code (available as file [RTMS.zip](#)). Unzip it to some directory.

In eclipse, go to the **File** menu and select **New** and then **Java Project (NOT A LeJos PROJECT!!)**. Type in **RTMS** for the "Project name". Make sure that a 32-bit java JRE is selected (i.e., not a 64 bit version). Click **Finish**. In the eclipse Navigator pane, under the Package Explorer, you should see **RTMS** appear. Right click on it and then select **Import...** from the menu. Under **General**, select **File System** and press the **Next>** button. Right click on the RTMS project and then select **Import...** from the menu. Set the "**From directory**" (i.e., by clicking the **Browse...** button) to be the folder that contains the folder titled **RTMS** that you downloaded and unzipped. Select the checkbox in the left pane of the dialog box for the **RTMS** folder (this should select all files in the right pane of the dialog box as well). Click **Finish**. It may ask if you want to overwrite a few files such as **.classpath**, **.project**, etc... just select **yes** to all. In the Package Explorer, you should notice many files appearing under the RTMS project now.

Select the **RTMS** project (if not already selected) and click on the down-facing arrow next to the run/play button (see selection (1) in the snapshot here). Select **Run As** (see selection (2) in snapshot) and then select **Java Application** (see selection (3) in snapshot).

It may then bring up a dialog box asking you to select the Java application. If so, select **RTMS - RTMS**. The RTMS should run and you should see a window appear as shown on the next page.





This is the running RTMS program, showing a simulated robot. From now on, anytime that you want to run the application, you can just select the **RTMS** project in the Package Explorer and press the Run/Play button.

Run the RTMS by selecting the **RTMS** project in the Package Explorer and pressing the Run/Play button. Click on the blue **Connect...** button in the **Robot** menu of the **RTMS**. The RTMS should connect to the robot if all was fine. Sometimes it fails to connect due to a previous run, and may require a 10 second attempt to try to connect, but usually it is fast. If it does not connect, there is a problem somewhere ... you are on your own to figure it out :(. Once connected, the **Connect...** button will change to **Start**. Notice that the robot is waiting for you to press this. Press the **Start** button. You should see the pose of the robot displayed on the LCD of the robot for 5 seconds, and then the program will end. Try grabbing and dragging the simulated RTMS robot to a new location on the window and see if it changes position when you re-run the robot program and re-establish the connection again.

It is possible that when you try to connect to the robot via bluetooth, then you get an error like this:

```
Native Library intelbth_x64 not available
Native Library bluecove_x64 not available
```

If this happens, it is because you have a 64-bit version of the JAVA runtime environment installed. You will need to go to here (<http://www.oracle.com/us/technologies/java/standard-edition/overview/index.html>) to get the latest 32-bit version on the JRE. Click on **Download** and then **Download JAVA SE For Developers**. You just need the JRE. Make sure to get the Windows X86 version. Once installed, then in click the little **arrow** beside the **Play/Run** button in Eclipse while the **RTMS** tab is selected. Select **Run Configurations...** Click on the **JRE** tab. Select the **Alternate JRE** radio button. Then select the jre that you just installed. Click **Apply** and the **Run**. Hopefully the error will be gone now.


LeJos and Bluetooth Setup

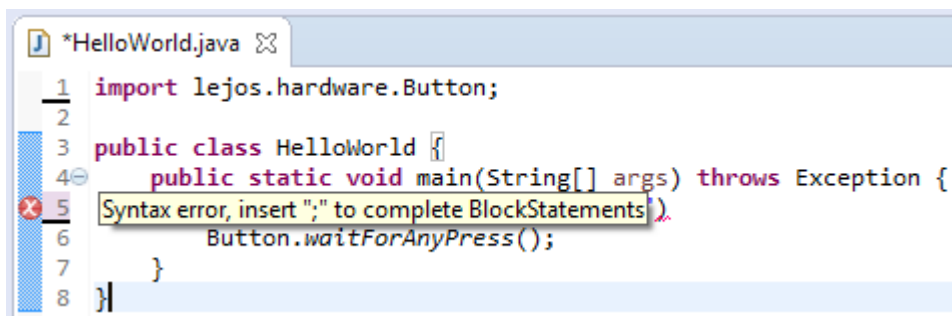
These are the instructions for installing LeJOS and Eclipse onto your computer for use in the course as well as getting things connected via bluetooth. The instructions are for Windows users. If you are using a Mac or a Linux laptop, you will need to look at the wiki page of the www.lejos.org website for further instructions ... which will bring you to the sourceforge website: [http://sourceforge.net/p/lejos/wiki/Installing leJOS/](http://sourceforge.net/p/lejos/wiki/Installing%20leJOS/) It is assumed for the remainder of this document that you are using a laptop running Microsoft Windows.

1. Download and install the x86 version of the Java SE 7 **JDK** (**DO NOT DOWNLOAD THE x64 VERSION**, although with Mac OS X there is only a 64-bit version. Also, make sure NOT to download the JRE ... you need the JDK). Also, do not download version 8 as it does not yet work on the EV3. When this was written, the latest was **jdk-7u67-windows-i586.exe** for windows) from this website: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
2. Download the **Java SE 7 Documentation** which is on this page: <http://www.oracle.com/technetwork/java/javase/downloads/index.html> near the bottom. Copy the file (e.g., **jdk-7u67-apidocs.zip**) to the Java installation folder (e.g., C:\Program Files (x86)\Java\jdk1.7.0_67). Unzip the file. You should have a folder called **jdk-7u67-apidocs** now. Go in there and drag the **docs** subfolder **up one level** into the **jdk1.7.0_67** folder which already has subfolders **bin, db, include, jre, lib** etc.... Delete the folder called **jdk-7u67-apidocs** as well as the **jdk-7u67-apidocs.zip** file.
3. Download the **0.8.1-beta** version of the leJOS windows installer from <https://sourceforge.net/projects/lejos/files/lejos-EV3/> and run it (for Windows, the file is called **leJOS_EV3_0.8.1-beta_win32_Setup.exe**). Follow the instructions to select the java jdk version that you installed in step 1 (version 7). Allow the remaining default options during the setup ... which will create a **leJOS EV3** folder in the **C:\Program Files** folder. Once installed, it will have a checkbox to **Launch the EV3SDCard utility**. Uncheck this because I have already prepared the micro SD cards for you and will give them to you on the first day of class (or possibly sooner if I see you).
4. Eclipse will be used as the IDE for the course. Download **Eclipse** from this website www.eclipse.org/downloads/. You will want the **Eclipse IDE for Java Developers** (usually the fourth option in the list). Make sure to download the **Windows 32 Bit** version (**NOT THE 64-BIT VERSION**). Just click on the down-facing green arrow. At the time this was written the file was called: **eclipse-java-luna-R-win32.zip**. Unzip the file after it downloads. A folder should have been created with the same name. It is a good idea to drag this folder to the **C:\Program Files (x86)** folder. Inside the folder is an **eclipse** folder and inside there you should see a purple circle icon ... this is the eclipse application. Right-click on it and make a select **Create shortcut**. Drag the shortcut to your desktop and rename it to **Eclipse**. Double-click to run it. It will ask you to create a workspace ... use the name **COMP1807_W2015**. You can close the "welcome" screen.
5. From the **Help** menu, select **Install New Software...** Click the **Add...** button. In the **Name** field, type in **leJOS EV3** and in the **Location** field, type in <http://lejos.sourceforge.net/tools/eclipse/plugin/ev3>. Click the **OK** button. Check off the checkbox that shows up labelled **leJOS EV3 Support**. Click the **Next>** button. Click **Next>** again. Click the radio button to accept the license agreement and then click the **Finish** button. You will get a warning ... just click **OK** to continue. Go ahead and restart Eclipse when it asks you. Once Eclipse has restarted, the plugin should be available and you will see a menu item called **leJOS EV3** as well as a couple of orange leJOS icons.


- Now we need confirm that EV3_HOME is set properly in the preferences. Click the **Window** menu and select **preferences** and then **leJOS EV3** in the list on the left hand side. You should set the **EV3_HOME** to your installed folder location (likely **C:\Program Files (x86)\leJOS EV3**) if it has not been set yet. Make sure that **Run Tools in separate JVM** is set as well as **Run program after upload**. Check off **Connect to named brick** and enter **10.0.1.1** into the text field labelled **Name**. Press the **Apply** and then **OK** buttons.
- Turn on your EV3 at this time. It typically takes a good minute or so for the EV3 to boot up. In fact, the very first time that you turn on the EV3 with the micro SD card, it will take a good 8 minutes or so to initialize everything!
- Now we will get a **Hello World** program running on the robot. All code in Eclipse is organized into *Projects*. Once eclipse has started, go to the **File** menu and select **New** and then **Project...** (**i.e. NOT Java Project**). Select the **LeJOS EV3** folder and then **LeJOS EV3 Project** and press **Next >**. Enter **COMP1807_W15** as the **Project name** at the top. Then click the **Finish** button. You should see the **COMP1807_W15** project appear in the workspace under the Package Explorer on the left side of the window.
- Select the **COMP1807_W15** project that you just made. Select **File/New/Class** from the menu bar.
- Type **HelloWorld** into the **Name:** field (with no spaces) of the resulting window. Select the checkbox that says **public static void main(String[] args)**. Click **Finish**. The new class should appear.
- Adjust code to look as shown here:


```
import lejos.hardware.Button;
public class HelloWorld {
    public static void main(String[] args) throws Exception {
        System.out.println("Hello World");
        Button.waitForAnyPress();
    }
}
```

- If there are any syntax errors, then will be shown with an  along the left side of the code:

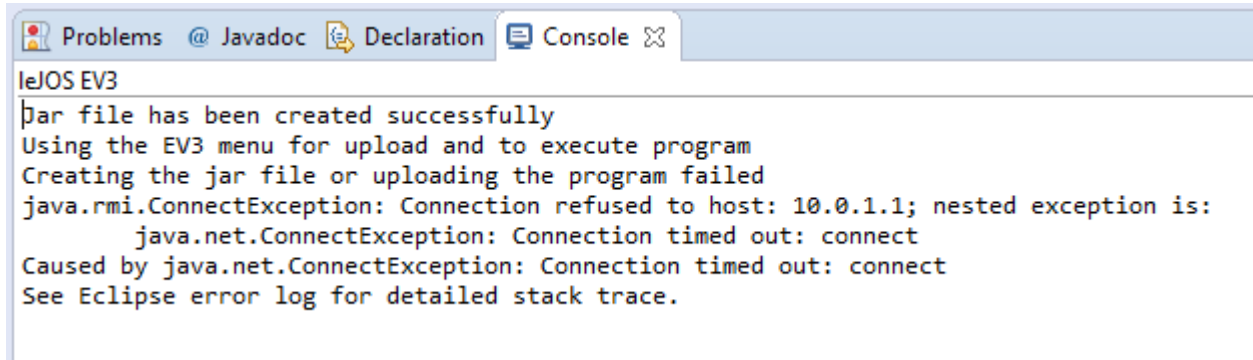


```
*HelloWorld.java ✕
1 import lejos.hardware.Button;
2
3 public class HelloWorld {
4     public static void main(String[] args) throws Exception {
5         Syntax error, insert ";" to complete BlockStatements
6         Button.waitForAnyPress();
7     }
8 }
```

Fix any syntax errors before continuing and then save your code by pressing the save button  (or press CTRL + S).

13. Once your code compiles and saves ok, we can run it. Make sure that the EV3 is turned on and that it is not currently running a program. Connect the EV3 to your laptop using the USB cable provided in the kit. In Eclipse, press the run button  to upload your program onto the robot and run it. The first time you do this for each program, it will ask how you want to run the program. Select **LeJOS EV3 Program** and press **OK**.

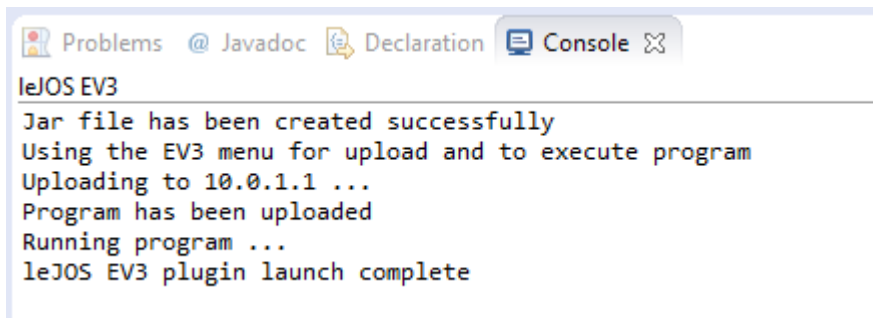
If the upload did not work, you will get a message like this in the bottom Console pane of Eclipse:



```
leJOS EV3
Jar file has been created successfully
Using the EV3 menu for upload and to execute program
Creating the jar file or uploading the program failed
java.rmi.ConnectException: Connection refused to host: 10.0.1.1; nested exception is:
    java.net.ConnectException: Connection timed out: connect
Caused by java.net.ConnectException: Connection timed out: connect
See Eclipse error log for detailed stack trace.
```

If this happens, just unplug the USB cable and plug it back in again.

Hopefully, the program will load and run on the robot. Note that it takes a bit of time for the program to start up. If it worked, you should see the following in the bottom Console pane of Eclipse:



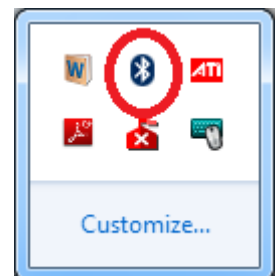
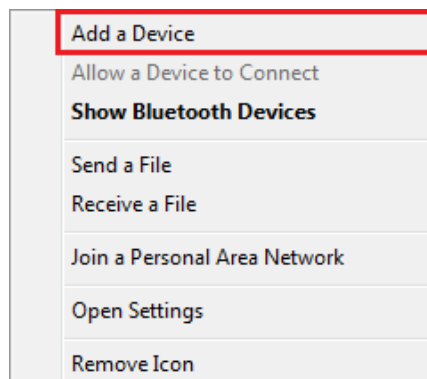
```
leJOS EV3
Jar file has been created successfully
Using the EV3 menu for upload and to execute program
Uploading to 10.0.1.1 ...
Program has been uploaded
Running program ...
leJOS EV3 plugin launch complete
```

You should also see the words **Hello World** appear on the EV3. Press any button on the EV3 to quit the program.

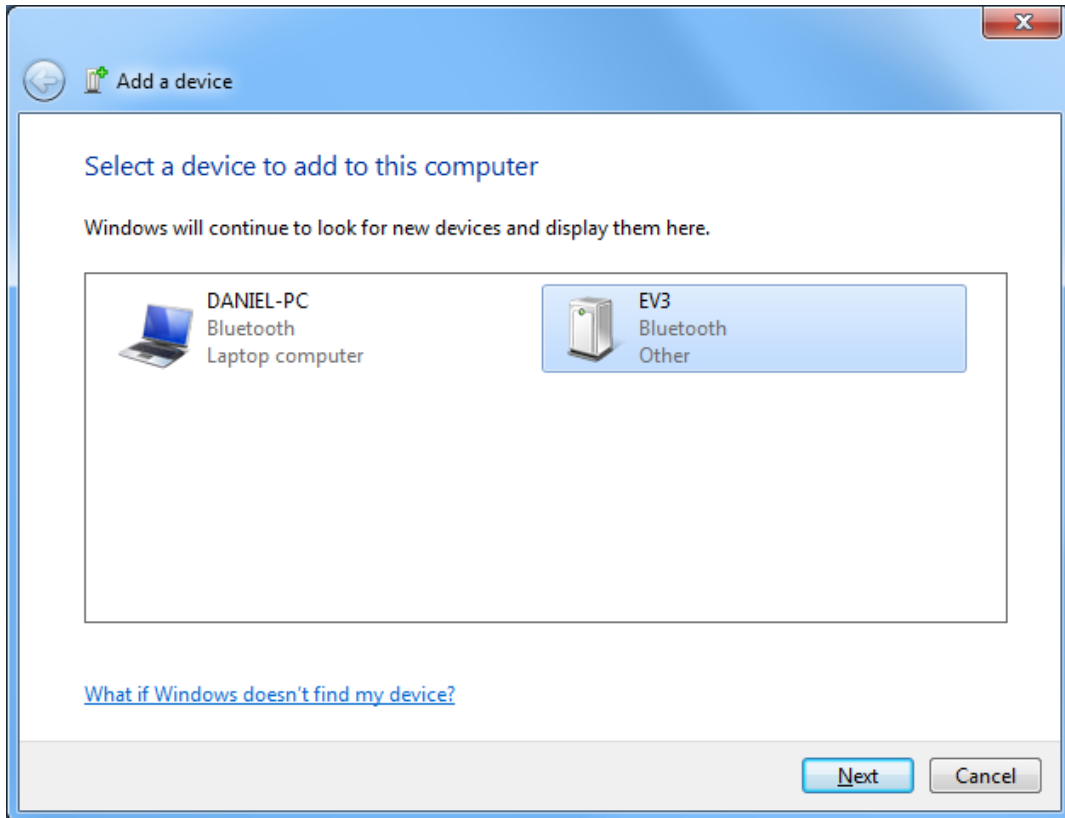
14. Unplug the USB cable ... we'll try to get it working with the Bluetooth. On the EV3, use the left and right buttons on the menu to scroll until it says "Bluetooth" ... with the bluetooth icon centered. Select it by pressing the center dark gray button. Make sure that the EV3 is visible by scrolling to "Visibility" (i.e., the eye icon). It should say "Visibility on", otherwise press the dark gray button to turn it on.

15. In the taskbar of windows, off to the right, you should see something like this under the hidden icons. Select the **Bluetooth** icon as shown here circled:

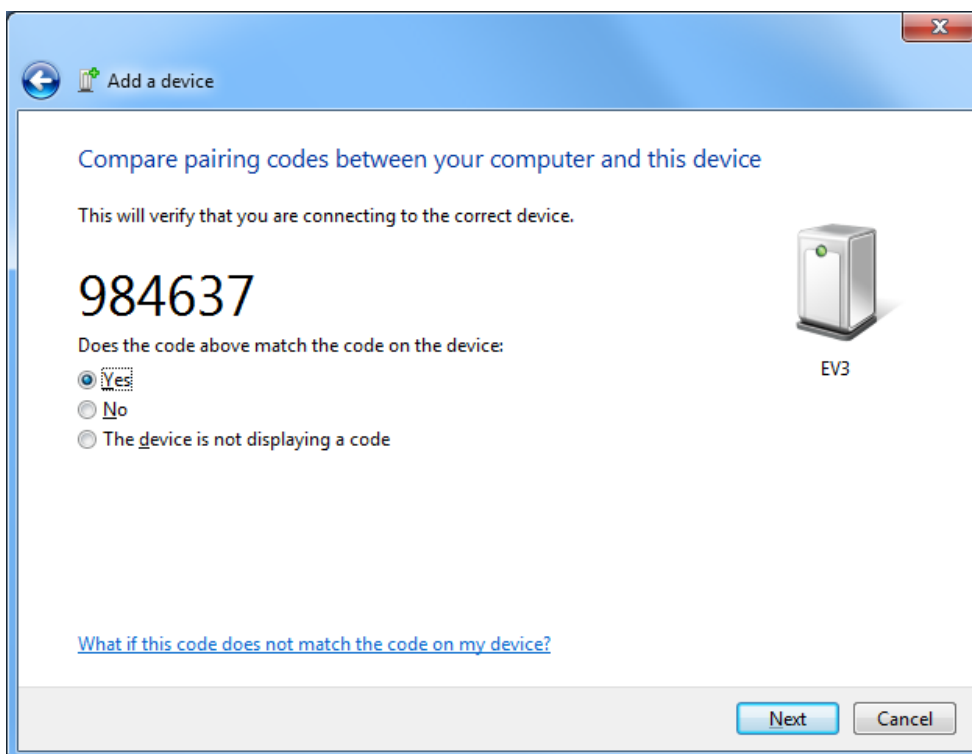
Now select **Add a Device** from the popup menu that appears:



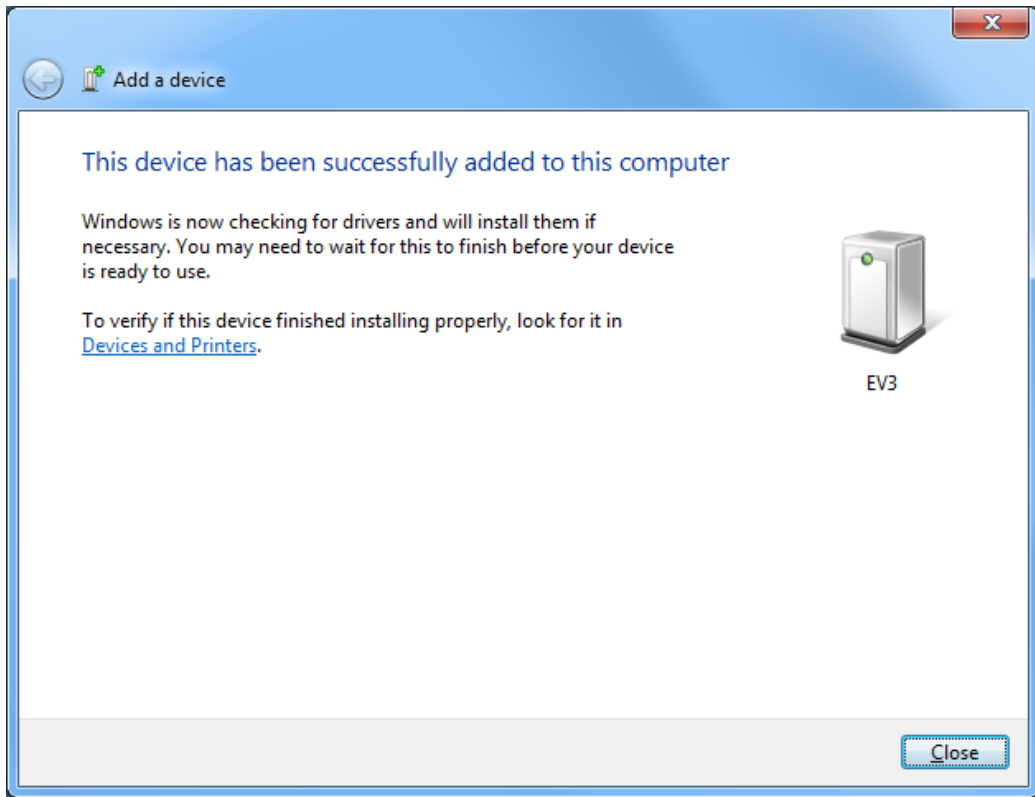
16. Hopefully, the EV3 will appear in the list of discoverable bluetooth devices. If not, double check that the Visibility is enabled on the EV3. Select the EV3 and press **Next**:



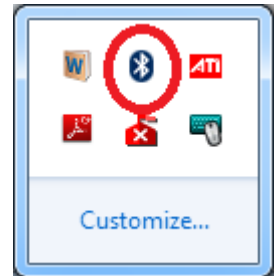
17. It will try to confirm the device that you are connected to. It should have the correct number ... provided that there is only one EV3 around. Don't try doing this in the lab when everyone has their EV3 on!!! Press **Next**. It should say that it is installing device driver software.



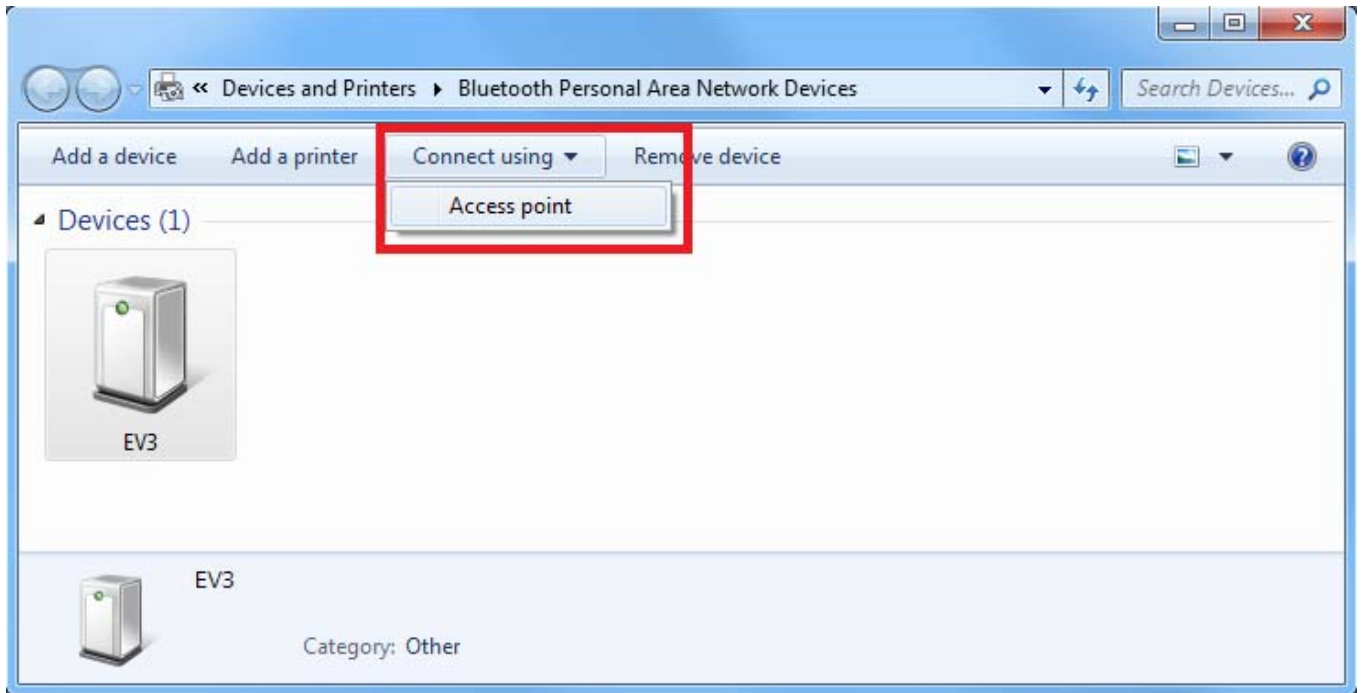
18. Hopefully you see this:



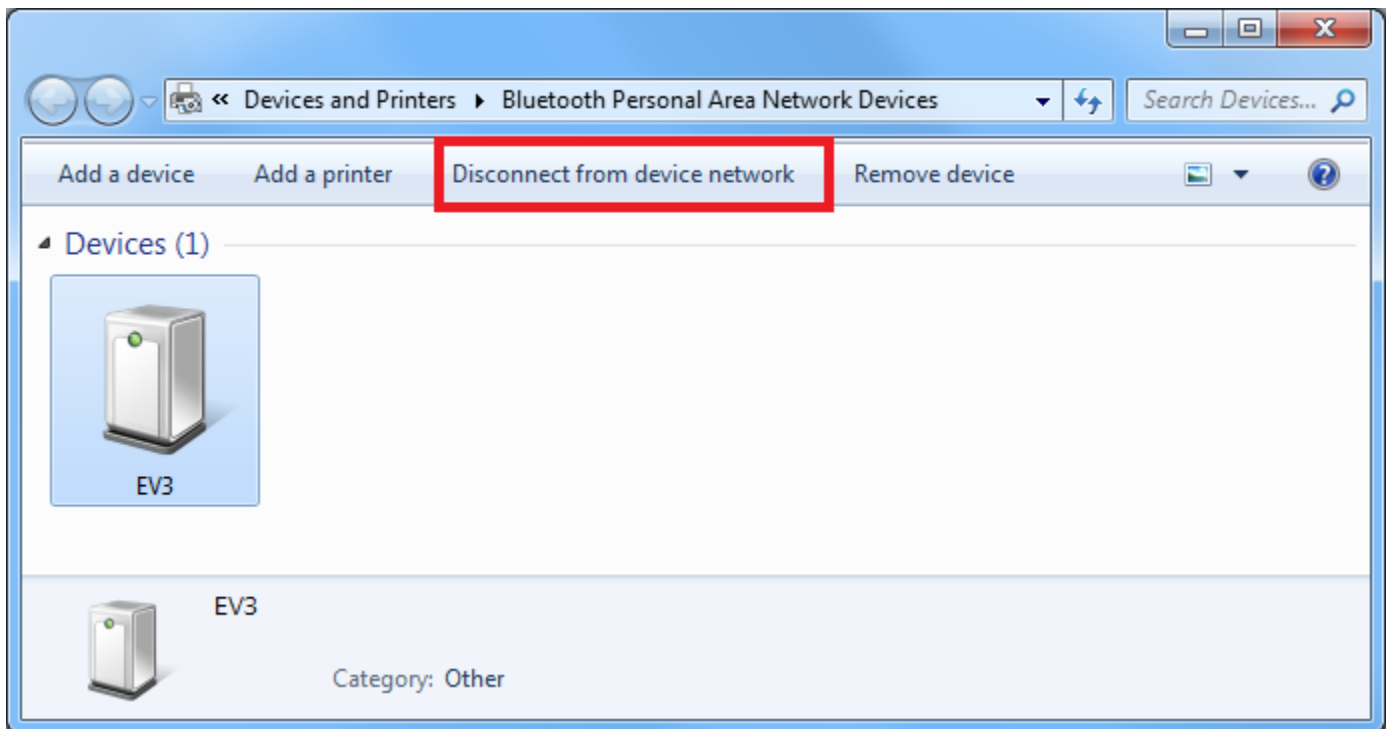
19. In the taskbar of windows, off to the right, select the **Bluetooth** icon again as shown here circled. This time though, select **Show Bluetooth Devices**.



20. Highlight the **EV3** and select **Access point** from the **Connect using** menu item.



It should connect and then the toolbar should change to say **Disconnect from device network**. Don't select this though! Your EV3 is connected now.



You should be able to upload code using bluetooth now. Try to run your program again (make sure that the USB cable is unplugged). Each time that you turn on your EV3, you will need to repeat steps 19 and 20.