To get started with Arduino and Xbee configuration you will need two different software programs. In order to write and upload code for the Arduinos, you need the Arduino 1.8.5 software, downloaded from here: https://www.arduino.cc/en/Main/Software
To configure the Xbees you need the XCTU software, downloaded from here: https://www.digi.com/resources/documentation/digidocs/90001526/tasks/t_download_and_install_xctu.htm
To read the data from the Xbees, you need CoolTerm, download from here: http://freeware.the-meiers.org/

Arduino Basics

Arduino language is C/C++ functions than can be called from the code. When first opened, the Arduino software will look like this. The “void setup” is where the code that runs once is put and “void loop” is where the code that calls for a continuous function goes.To connect an Arduino board to the software make sure that the correct port is selected. To do this go to *Tools→ Port*, then select the Arduino Board. This allows code to be uploaded to the board and the ensuing data to be monitored. To monitor the data coming from the board after a sketch has been uploaded, go to *Tools→ Serial Monitor* to see the data itself or to *Tools→ Serial Plotter* which will show a graph of the data. To upload a sketch to the board, first click “Compile” which is the check mark in the upper left corner of the screen. Then, hit “Upload” which is the arrow directly right of the check. Comments are indicated by either // or /* */ and usually explain the code written. If you are uploading sketches to the Arduinos when the shield is attached, make sure that the black switch (opposite the side with the screw terminal) is set to DLINE as opposed to UART.

Temperature Sensors

A. Configuration

To attach the temperature sensors to the Arduino board, use the screw terminals that have been soldered onto the red shield. On the temperature sensor, there are three wires that need to be attached. Red is charge, black is ground, and yellow is data. Place each one of these in a separate terminal. Place a 4.7kΩ resistor between the yellow and red (charge and data) wires. Then add jumper wires to each of the three terminals (it’s helpful to keep the same colors). Tighten the screws to keep all the

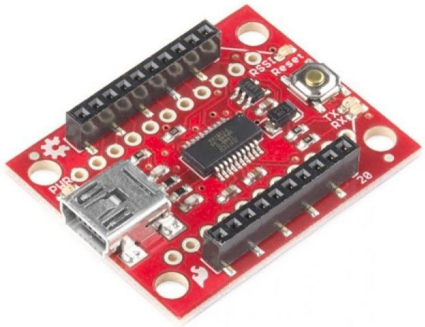
wires in place. The red wire attaches to the 5V pin on the Arduino, the black goes to ground and the yellow goes to pin 2 on the digital side of the Arduino.

B. Coding

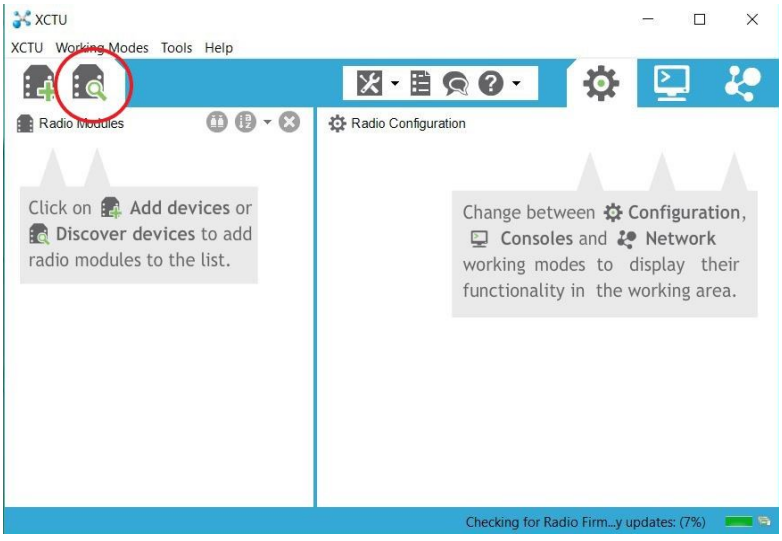
Before beginning any coding, download the Dallas Temperature library, found here: https://github.com/milesburton/Arduino-Temperature-Control-Library
To add it to the Arduino libraries, go to *Sketch→ Include Library→ Add .ZIP Library* and select the Dallas Temperature Library. Then download the Arduino Temperature code from here: https://github.com/lexiegardner/Temperature-Sensors
Upload the sketch to the board and open the serial monitor to ensure it is working.

Xbees

To configure an Xbee, first attach it to the Xbee Explorer which can then be connected to a computer using a mini USB cable. There are three ways to configure an Xbee unit--as a router, a communicator or an endpoint. There has to be one communicator for every network that receives data from the other endpoints and routers. Endpoints can send and receive data but cannot act as a messenger whereas routers can be messengers as well as send and receive data. Use XCTU to configure the Xbees. To do this, click on the “Search” icon in the top left corner of the screen. A list of active COM ports will be displayed, choose the Xbee unit and select the default port parameters on the next screen (do not change anything). After you hit finish, the USB ports are scanned and the Xbee found will be displayed, select it and click “Add Selected Devices”.



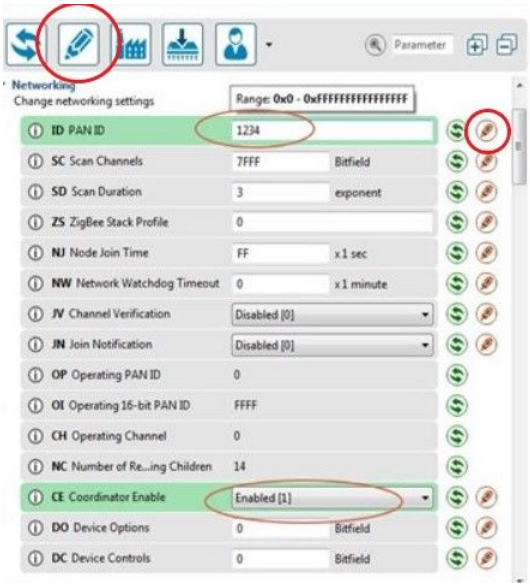
Xbee Explorer



TO CONFIGURE AS A COORDINATOR:

Click on the Xbee icon in the list on the left side of the screen. In the parameters, the first thing to change is the PAN ID (this needs to be the same for all the Xbees in a network). The PAN ID must be unique to your network and should be between 0 and FFFF hex. Scroll down and change the ***CE*** *Coordinator Enabled* to “Enabled (1)”. Further down, change the ***DL*** *Destination Address Low* to “FFFF” so that it is in broadcast mode and can communicate with all other devices on the same PAN ID. The ***DH***

Destination High should remain at 0. To save the changes click the pencil icon at the top of the screen. Changes can also be written as you go by clicking on the pencil icon to the right of the changes you make.

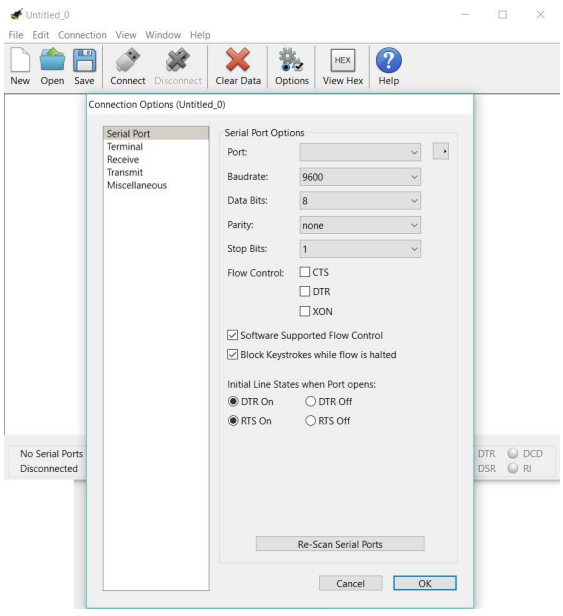


TO CONFIGURE A ROUTER:

Plug the Xbee Explorer, with a new Xbee device attached, into the computer and follow the above instructions so that the device is added to the XCTU main screen. Enter the same PAN ID as used on the Communicator. Make sure that ***CE*** *Coordinator Enabled* is Disabled. Enable the ***JV*** *Channel Verification*. Leave both the ***DH*** *Destination High* and ***DL*** *Destination Low* at 0, which is the default address of the Coordinator. Write the changes by clicking the pencil icon at the top of the screen.

The link below is very helpful if you run into any issues configuring the Xbees.
https://alselectro.wordpress.com/2017/01/23/zigbee-xbee-s2c-how-to-configure-as-coordinator-router-end-device/

To use the Xbees with the Arduinos, attach them to the shield. When uploading sketches be sure to remove the Xbees from the shield (also as aforementioned, make sure the switch is in DLINE). When using the Xbees make sure the switch on the shield is in the UART position. To use the Xbees, plug the Arduino into an external battery source (either the wall or a battery, not a computer) and make sure that the Coordinator is attached to the Xbee Explorer and plugged into the computer. Use CoolTerm to read the data coming in off the Xbees transmitting data. When removing the Xbees from the shields make sure the board is not plugged in.



CoolTerm

To select a Serial Port in CoolTerm select Options at the top and there will be a dropdown menu. Select a port and hit okay. Once the port is selected, hit connect on the top bar. This will begin to read the data coming in off the Xbee. To begin to capture the data to a text file click on *Connections* at the very top in the task bar and at the bottom choose *Capture to Text File→ Start.* This will begin recording the data to a text

file that can be accessed later. To add a timestamp to the data, click Options, then under Receive, choose *Add timestamp to received data*. The timestamp will only appear if you are capturing the data to a text file. It will not show up in the monitor itself. Any settings you change will not be saved unless indicated that you want them to be saved as defaults. To save your current settings as defaults, click File then choose *Save as Default*.