Pebble Watch User Documentation

Group Member: Jue Liu, Ao Sun, Xiaojun Sun

This user documentation will give you detailed instruction on how to use our pebble watch application. In general, our system contains three components. Below is the step-by-step user guide on how to make each of the components work and finally connect them to work collaborately.

1. Pebble Watch

* Pebble watch send the requests to the web servers through blue tooth connection with the cell phone, and display results after receiving responses.
* To run the Pebble watch app, just open the app. You will see a menu like this:

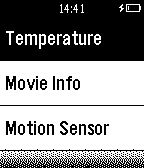


Figure1 - Initial Interface

* Use UP and DOWN button to choose the feature and click SELECT button to open it.
* In temperature mode, click SELECT button to switch the display among “current temperature”, “highest temperature”, “lowest temperature”, “average temperature”. Click UP button to switch the temperature unit between “Celsius” and “Fahrenheit”. Click DOWN button to start and stop the standby mode. If the Arduino is disconnected or the server is shut down, error message will be displayed.

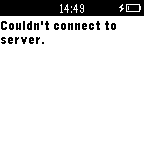
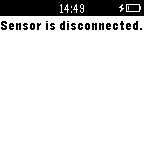


Figure 2 - Server disconnection error message display

* In the movie mode, click the SELECT button to get the latest movie information of the Penn 6 Cinema.

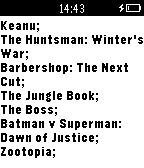


Figure 3 - Daily update movie information

* In the sensor mode, click the SELECT button to gather sensor data. If the Arduino is disconnected or the server is shut down, error message will be displayed.



Figure 4 - Ping sensor data message

* In the weather mode, click the SELECT button to get the latest weather information of Philadelphia, and then, send the weather information to the server.

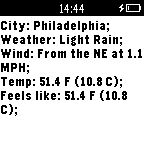


Figure 5 – Weather information display interface

2. Server

* Server plays the role of middleware to help communication between Pebble Watch and Arduino.
* To run the server, use Command Line Interface, go to the root file, type in “make” to compile the server.cpp. Then type in “./server [port\_number] [arduino\_file\_path]” to start server. For example, you can type in “./server 3001 /dev/cu.usbmodem1421”. Tip: after Arduino is connected properly through a USB port, you can get the Arduino port address from Tools → Port → /dev/cu.usbmodem1421(for example).
* To stop the server, just enter a single ‘q’ in the command line. After one more request send from Pebble, the server connection will be closed.

3. Arduino Setup & Sensor Display

* All the programming work that controls the sensor and display is realized with Arduino microcontroller. For hardware configuration, connect Arduino with Mac machine via USB. For proximity sensor, attach it to a piece of shield and connect the three signal pins to the corresponding pins on Arduino via wires. Up to now, all hardware is set up.

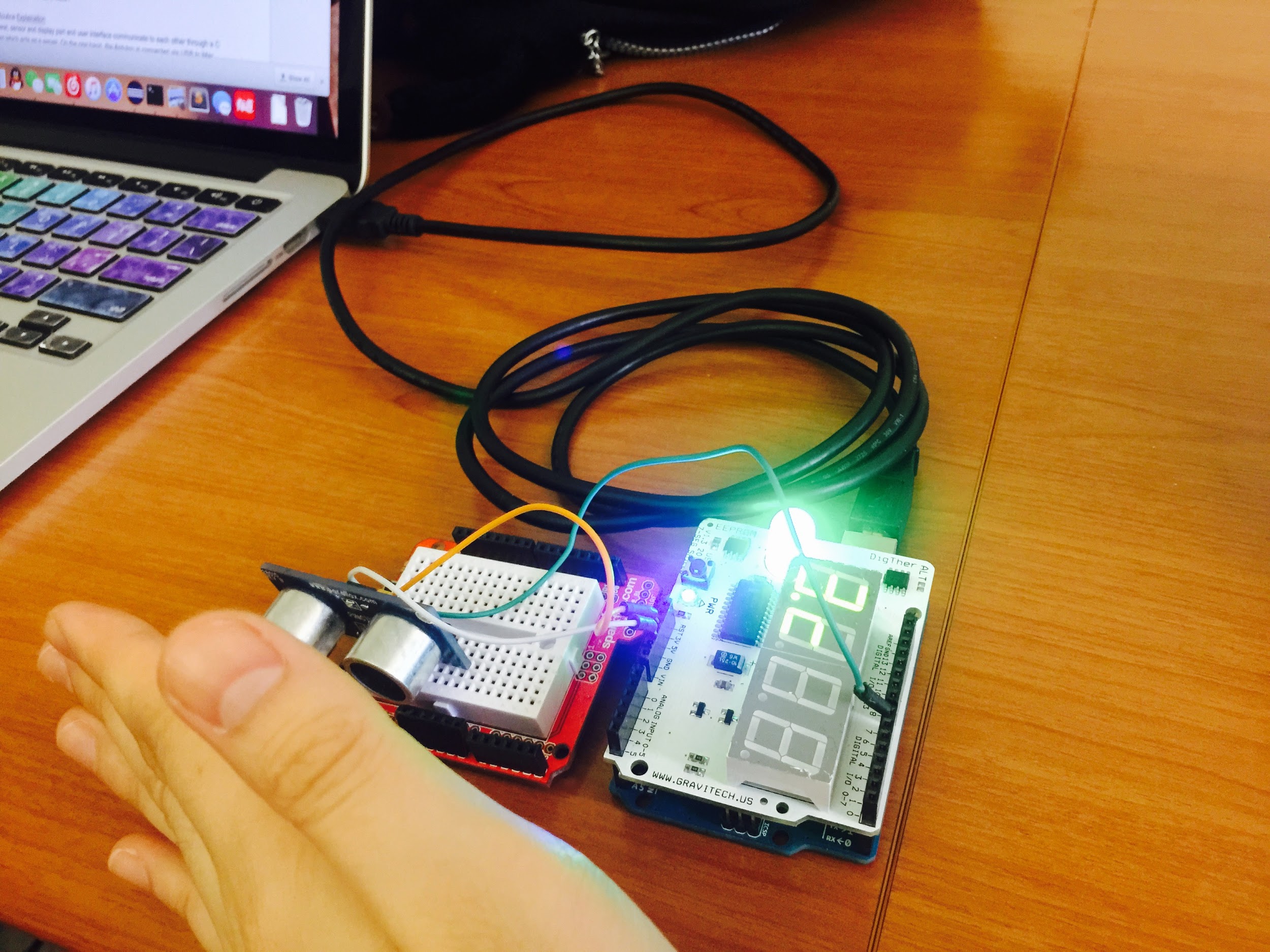


Figure 6 Hardware Configuration

For software configuration, follow the next steps:

1. open Arduino IDE and select File → Open File, select arduino.ino file.
2. Then click Tools → Port and identify the connected port.
3. Next, click Verify button, after it compiles successfully, click Upload button to run the file.

If it uploads successfully, the code has been loaded to the Arduino MCU after these steps. Now if you select Tools → Serial Monitor, you could see the printed lines about temperature or distance data. The serial monitor is only for you to understand how it works. If you are running the cpp server program, please close the this window, otherwise the cpp server may not work properly.

* If the code is uploaded successfully, you will see it displays temperature that is read from temperature sensor in Celsius or Fahrenheit depends on the command in temperature mode, or displays distance in ping mode. Or it will display the current temperature in Celsius sent from user interface if it is now in weather mode. The RGB light is used to monitor the temperature value. If it is below 23 degrees Celsius, the blue light will turn on indicating it is cold. If it is above 26 degrees Celsius, the right light will turn on indicating it is hot. Otherwise, the green light is on. If you see that there is no display, then it must be that the server sent a stand-by mode command so that the arduino stops displaying any data. However, it is still writing the data to serial port.