### **Rfu-328 Progress report:**

# Currently implemented:

The system can currently use radio frequency to send and receive LLAP Messages to another device to perform basic functionality.

The LLAP messaging library makes use of 12 byte message packet. The first byte is the character “a” and signifies the start of a message. The second two bytes are used for the device ID, this will be then used to determine what device the message is for. The last 9 bytes are used for the command and values that the receiver will use.

Due to systems potential of being messages being interfered by other devices, the systems PAN ID has been changed to 1066. This will result in the devices only being able to see messages using sent using the same ID.

The system also makes use of the LCD16x2 library to control the LCD screen.

Extra Notes;

Pins; d2, d11, d12 and d13 do not work correctly, avoid using them if possible.

Device 0 has a bad antenna

Chip settings cannot be changed unless using an empty sketch is used

Use the “+++” command to enter command mode

# Further Plans:

Continued work on the system will involve, sending data faked by using a LED light and a variable resistor from the controller and sending it to the hub. The hub will then receive this and output the data to the LCD screen. The hub should also be able to turn the LED light on the controller off and on by press of a button.

Currently we can send a message to a controller with a fixed ID. However this is not ideal as hard programing the id is not an option since it leads to potentially duplicate ID’s over multiple controllers.

This will mean we would be unable to send a message to a specific device. Vice versa the controllers also need to know the hub that they will be connected to so that a controller’s message is not interpreted by someone else’s e.g a neighbor’s hub. In order to fix this issue a handshake protocol needs to be created. Two potential method to do with is as follows

# One button sync method

**Controller**

**Hub**

Creates an ID

Sends Hubs ID

On Button press

If controller is not already linked to a Hub

Sends Controller ID Using the Hubs ID

Listens to requests

Creates an ID

Listens to requests

Adds to list of controllers

Pros

* Requires only one button, so easier for a user to sync.

Cons

* Controllers can become attached to nearby hubs requiring switching off and on to sync to correct Hub.
* Controller will use power waiting for hub to send their ID

# Two button sync method

Sends Controller ID Using the Hubs ID

On Button press

On Button press

**Hub**

**Controller**

Listens to requests

Sends Hubs ID

Creates an ID

Listens to requests

If controller is not already linked to a Hub

Adds to list of controllers

Creates an ID

Pros

* Saves power since the controller is not looking for an ID to be sent to the hub when the button is not pressed and can sleep.

Cons

* More effort for user to press a button on each of the controllers and the hub