

## CSC530 Final Part 2 – Building a Mesh

### Background

A Mesh Network (Mesh) is a collection of devices/nodes that are connected for communication. A Mesh can be full or partial (all nodes know about all other nodes, or not). A Mesh is a robust way to create an interconnected set of nodes, and can be created in an *ad hoc* manner, making them useful in situations where networks need to be realized on the fly.

Nodes in a Mesh can communicate with each other by routing messages to specific individual nodes, or by flooding the Mesh with messages. The addressing scheme is dependent on the protocol of Mesh. There are many kinds of Meshes (WiFi, Bluetooth, etc). We will be exploring building an *ad hoc* mesh using a proprietary wireless protocol from Espressif that runs on ESP32 devices: ESP-NOW. API documentation can be found here:

[https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp\\_now.html](https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp_now.html)

### Assignment

Your assignment will be to build an *ad hoc* Mesh using ESP-NOW on your 2 devices (M5Core2 and Seeed Studio XIAO).

Write a single program to be deployed on both your ESP32 devices to do the following:

- a) Initial Send: Broadcast an ESP-NOW message at 1 second intervals for 5 seconds announcing the presence of the device.
- b) Receive: For each message from a new device, record the device MAC in an internal data structure. NOTE: You will receive 2 kinds of messages. Broadcast messages sent by new devices, and welcome messages after you broadcast (also from new devices). You only need to send a response back to the broadcast messages. Response messages should be directed to specific MAC addresses.
- c) Messages: Your broadcast message should say “Hello”. Your response messages should say “Welcome”.
- d) Output: Whenever you discover a new device, print the MAC addresses of all the peers you currently know about on the Serial terminal (format it nicely!).
- e) Run the Mesh code on the second core (core 0) of your devices.