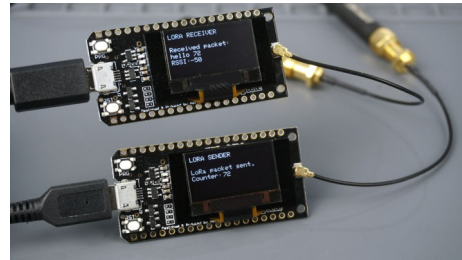


Trabajo LoRa

Con el dispositivo TTGO ESP32 OLED, hemos instalado Arduino ideg versión 2.1 y posteriormente hemos descargado las especificaciones de la placa que hemos usado que en este caso a sido la TTGO LoRa32-OLED



Le hemos realizado un reset a los dispositivos

Seguidamente hemos buscado una librería de LORA y OLED y la hemos intslado, la librería a sido la siguiente:

SSD1306

<https://github.com/sandeepmistry/arduino-LoRa>

Posteriormente le hemos instalado un código para enviar paquetes a distancia que posterior mente le hemos instalado a otro Arduino otro código para recibir los paquetes que estábamos enviado con el primer dispositivo Arduino

Es es el codigo para enviar los paquetes

```
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET 4
Adafruit_SSD1306 display(OLED_RESET);

int ky013 = A0; // Valores de entrada analógicos por el termistor
int valor; // Variable que guardará los datos de entrada del KY-013

// Variables para realizar la ecuación Steinhart-Hart
float R1 = 10000; // Valor de R1 de la PCB 10k
float logR2, R2, TK, TC; // Elementos para realizar la ecuación
float A = 0.001129148, B = 0.000234125, C = 0.0000000876741; // Valores constantes considerando NTC de 10K y trabajando a una temperatura entre -55°C a +125°C

void setup() {
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setTextSize(1);
  display.setCursor(0, 0);
  display.display();
}

void loop() {
  valor = analogRead(ky013); // Se asigna los valores leídos a la variable valor

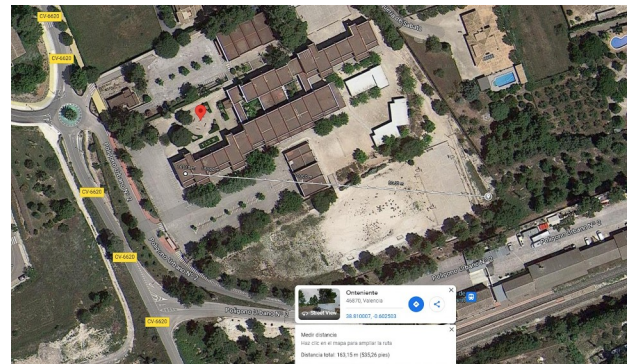
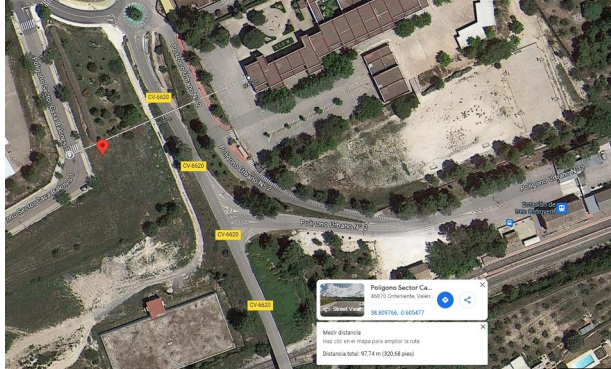
  valor = map(valor, 0, 650, -55, 125); // Para evitar la exponencial, usaremos la función map
  // en donde 0-650, serán los valores de entrada analógica vs -55-125 el valor de temperatura

  // Conversión de valores analógicos a grados Celsius
  R2 = R1 * (1023.0 / (float)valor - 1.0); // Cálculo de la resistencia del termistor
  logR2 = log(R2); // Cálculo del logaritmo de la resistencia del termistor
  TK = (1.0 / (A + B * logR2 + C * logR2 * logR2)); // Temperatura en Kelvin
  TC = TK - 273.15; // Conversión a Temperatura Celsius
  TC = TC * (-1);

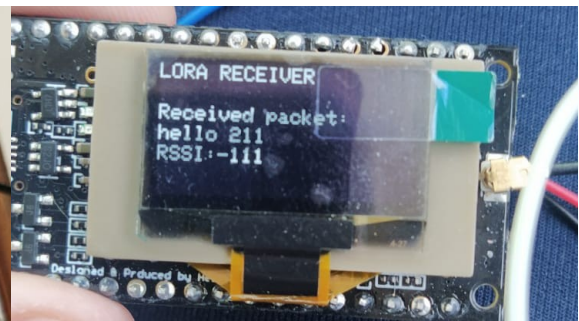
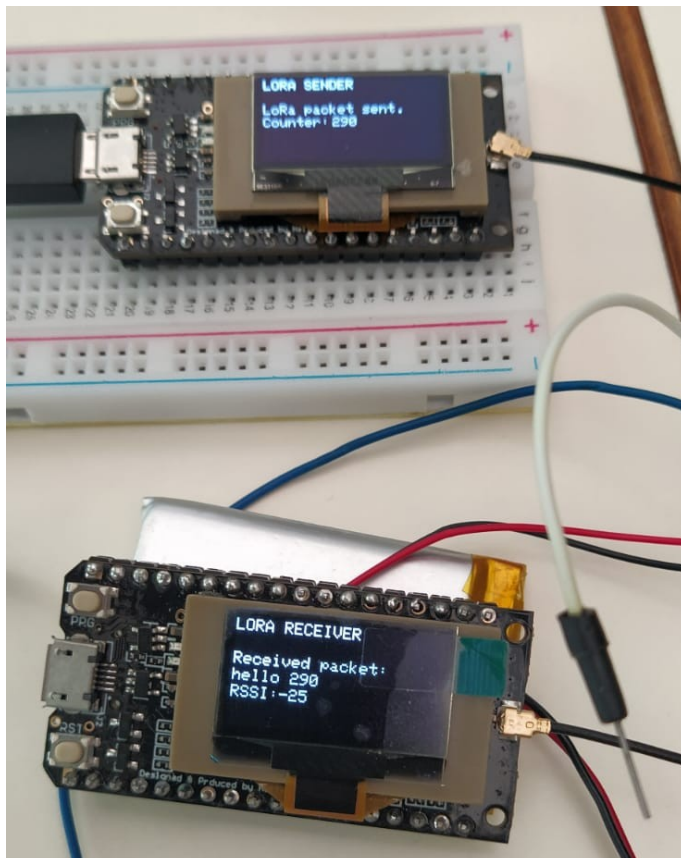
  display.clearDisplay();
  display.setCursor(0, 0);
  display.println("Temperatura:");
  display.print(TC);
  display.println(" C");
  display.display();

  delay(1000); // Espera de 1 segundo antes de actualizar la pantalla
}
```

Posterior mente para comprobar la distancia de envío han salido fuera dos compañeros del aula y han recorrido una distancia lejana al centro mientras que dos compañeros están en el aula comunicándose vía telefónica comprobando el envío de los paquetes enviados.



Llegando a la conclusión de que el RSSI (Indicado de fuerza recibida) que ha menor sea el numero mejor sera la recepción del paquete



Por aquí dejare los enlaces los cuales hemos utilizado para lludarnos en esta practica

<https://randomnerdtutorials.com/ttgo-lora32-sx1276-arduino-ide/>

<https://randomnerdtutorials-com.translate.goog/installing-the-esp32-board-in-arduino-ide-windows-instructions/? x tr sl=auto& x tr tl=es& x tr hl=es>

<https://blog.uelectronics.com/tarjetas-desarrollo/sensores-de-temperatura-ky-028-y-ky-013/>