

Presentación de resultados de la integración de comunicación

Problemas presentados y su solución

Problema: El ESP32 no se conecta al dispositivo Bluetooth

Descripción: El ESP32 está visible en el dispositivo móvil pero no logra establecer una conexión.

Causas Potenciales:

- El código del ESP32 no está correctamente configurado para aceptar conexiones.
- Modo Bluetooth incorrecto (BLE vs. Classic).
- Dispositivos fuera de alcance.

Solución:

- **Verificar el modo Bluetooth:** Asegúrate de que ambos dispositivos estén usando el mismo protocolo (Bluetooth clásico o BLE). Si usas BLE, asegúrate de que el ESP32 esté configurado para manejar este tipo de comunicación.

// Para Bluetooth clásico:

```
BluetoothSerial SerialBT;
```

```
SerialBT.begin("ESP32_BT");
```

- **Revisión del alcance:** Acerca los dispositivos para asegurarte de que estén dentro del rango adecuado (hasta 10 metros en línea recta para Bluetooth).
- **Pruebas con otros dispositivos:** Intenta conectar el ESP32 a diferentes dispositivos para verificar si el problema está en el dispositivo Bluetooth o en el ESP32.

Problema: El ESP32 se desconecta aleatoriamente de Bluetooth

Descripción: El ESP32 se conecta correctamente, pero pierde la conexión de manera intermitente.

Causas Potenciales:

- interferencia de otros dispositivos Bluetooth o Wi-Fi (en la banda de 2.4 GHz).
- Exceso de carga en el ESP32 (falta de recursos de memoria).
- Problemas de señal o alcance.

Solución:

- Reducción de interferencias: Asegúrate de que no haya otros dispositivos Bluetooth o Wi-Fi que causen interferencias. Si estás usando Wi-Fi y Bluetooth al mismo tiempo, intenta cambiar el canal Wi-Fi.
- Optimización de memoria: Usa `ESP.getFreeHeap()` para verificar cuánta memoria disponible tiene el ESP32. Si es baja, libera memoria no utilizada o reinicia periódicamente.
- Ajustar el intervalo de conexión: Si la desconexión ocurre frecuentemente, intenta ajustar los intervalos de conexión Bluetooth para reducir el consumo de energía.

```
BLEDevice::setPower(ESP_PWR_LVL_P7); // Ajustar la potencia de transmisión BLE
```

Problema: No se puede enviar ni recibir datos

Descripción: El ESP32 se conecta, pero no puede enviar ni recibir datos correctamente desde el dispositivo conectado.

Causas Potenciales:

- El dispositivo Bluetooth remoto no está configurado para aceptar o enviar datos.
- Problemas de emparejamiento (pairing) o sincronización de protocolos.
- El ESP32 está usando servicios y características incorrectas en Bluetooth BLE.

Solución:

- Revisar el emparejamiento: Verifica que los dispositivos estén correctamente emparejados. Si hay algún conflicto de emparejamiento anterior, elimina el dispositivo de la lista de dispositivos Bluetooth y vuelve a emparejarlo.
- Revisión de características BLE: Si estás usando BLE, asegúrate de que las características y servicios configurados sean correctos y estén habilitados para el envío y recepción de datos:

```
BLECharacteristic *pCharacteristic = pService->createCharacteristic(
```

```
    CHARACTERISTIC_UUID_TX,
```

```
    BLECharacteristic::PROPERTY_NOTIFY);
```

```
pCharacteristic->addDescriptor(new BLE2902());
```

- **Uso de Serial Bluetooth:** Si estás usando Bluetooth clásico, asegúrate de que estés utilizando el objeto Serial correctamente:

```
if (SerialBT.available()) {  
  
char incomingChar = SerialBT.read();  
  
Serial.print(incomingChar);  
  
}
```

Problema: Errores de emparejamiento Bluetooth

Descripción: El ESP32 solicita emparejamiento, pero no logra completarlo con éxito (PIN incorrecto o problemas de seguridad).

Causas Potenciales:

- PIN incorrecto.
- Problemas de seguridad o versión Bluetooth incompatibles.
- Fallos en la sincronización de emparejamiento.

Solución:

- Verificar el PIN: Si estás utilizando un PIN para emparejar dispositivos, asegúrate de que coincida en ambos dispositivos. Puedes establecer un PIN manualmente:

```
bool pinResult = esp_bt_gap_set_pin(ESP_BT_PIN_TYPE_FIXED, 4, (uint8_t *)"1234");
```

- **Eliminar dispositivos emparejados:** A veces, limpiar los dispositivos emparejados previamente y volver a emparejar puede solucionar el problema:
 - En el dispositivo móvil, ve a la configuración de Bluetooth y elimina el ESP32 de la lista de dispositivos emparejados.
 - Reinicia el proceso de emparejamiento.
- **Comprobar la versión Bluetooth:** Asegúrate de que el ESP32 y el dispositivo emparejado sean compatibles con la misma versión de Bluetooth.

Problema: Interferencia entre Wi-Fi y Bluetooth

Descripción: Si se usa Wi-Fi y Bluetooth al mismo tiempo en el ESP32, la conexión Bluetooth se vuelve inestable o lenta.

Causas Potenciales:

- Interferencia de las frecuencias en la banda de 2.4 GHz.
- Recursos limitados del ESP32 al manejar Wi-Fi y Bluetooth simultáneamente.

Solución:

- **Optimización de coexistencia:** Utiliza la coexistencia de Wi-Fi y Bluetooth en el ESP32 para evitar interferencias:

```
esp_bt_controller_mem_release(ESP_BT_MODE_CLASSIC_BT);
```

Esto libera memoria Bluetooth clásica cuando se usa BLE y Wi-Fi al mismo tiempo.

- **Reducción de uso simultáneo:** Si no es estrictamente necesario usar Wi-Fi y Bluetooth simultáneamente, intenta desactivar uno mientras usas el otro.

Problema: Problemas de velocidad y latencia en Bluetooth

Descripción: La velocidad de transmisión de datos en Bluetooth es lenta o los datos se transmiten con retraso.

Causas Potenciales:

- El ESP32 está en modo de baja energía (BLE) y la tasa de transmisión es baja.
- Interferencia de otros dispositivos Bluetooth o Wi-Fi.
- Configuración de tamaño de paquetes ineficiente.

Solución:

- **Ajustar la tasa de transmisión:** Si estás usando BLE, ajusta los intervalos de conexión y los parámetros de la MTU (Maximum Transmission Unit) para mejorar la velocidad:

```
BLEDevice::setMTU(512); // Establecer un tamaño de paquete mayor
```

- **Mejorar la gestión de paquetes:** Si estás enviando grandes cantidades de datos, intenta dividir los paquetes en tamaños más pequeños y optimizar la gestión del buffer para reducir el tiempo de latencia.
- **Uso de Bluetooth clásico:** Si la velocidad es crítica, considera usar Bluetooth clásico en lugar de BLE, ya que tiene una mayor tasa de transferencia de datos.

Problema: El ESP32 se cuelga o se reinicia al intentar conectarse vía Bluetooth

Descripción: El ESP32 se reinicia o se congela cuando intenta establecer una conexión Bluetooth.

Causas Potenciales:

- Sobrecarga de memoria o mal manejo de recursos.
- Problemas con el controlador Bluetooth del ESP32.

Solución:

- **Liberar recursos no utilizados:** Asegúrate de que tu código no esté consumiendo demasiada memoria. Utiliza `ESP.getFreeHeap()` para monitorear el estado de la memoria. Si es necesario, reinicia el Bluetooth o desactiva funcionalidades no esenciales.
- **Actualizar la versión de firmware:** Verifica si hay actualizaciones de firmware disponibles para el ESP32, ya que algunas versiones anteriores pueden tener bugs en el manejo de Bluetooth.
- **Manejo correcto de desconexiones:** Asegúrate de que tu código maneje correctamente las desconexiones y reconexiones para evitar que el ESP32 se bloquee.

Problema: No se detecta el ESP32 en la lista de dispositivos Bluetooth

Descripción: El ESP32 está configurado para ser visible, pero no aparece en la lista de dispositivos Bluetooth del teléfono u otro dispositivo.

Causas Potenciales:

- El ESP32 no está en modo visible (advertising mode en BLE).
- Problemas con la señal Bluetooth.
- Uso de un nombre o servicio no compatible.

Solución:

- **Verificar modo visible:** Si estás usando BLE, asegúrate de que el ESP32 esté en modo de advertising:

```
pServer->getAdvertising()->start();
```

```
Serial.println("Advertising started");
```

- **Reiniciar el dispositivo:** A veces, reiniciar el ESP32 o el dispositivo móvil puede solucionar problemas temporales de visibilidad.

- **Comprobar configuración de nombre y servicio:** Verifica que el nombre del dispositivo Bluetooth esté correctamente configurado y que los servicios estén habilitados.

Problema: No se establece la comunicación entre los módulos LoRa

Descripción: Los módulos LoRa no logran comunicarse entre sí, a pesar de que están encendidos y configurados.

Causas Potenciales:

- Diferentes frecuencias de operación configuradas en los módulos.
- Distancias demasiado largas para el rango soportado.
- Antenas mal conectadas o dañadas.

Solución:

- **Verificar frecuencias:** Asegúrate de que ambos módulos estén configurados en la misma frecuencia de operación (433 MHz o 868 MHz, según el modelo y la región).

`LoRa.setFrequency(433E6);` // Establecer la frecuencia en 433 MHz

- **Revisión de la distancia:** Asegúrate de que los módulos estén dentro del rango de comunicación LoRa (varía según las condiciones, pero generalmente puede llegar a varios kilómetros en campo abierto).
- **Revisar antenas:** Verifica que las antenas estén correctamente conectadas y no dañadas. Un mal contacto en la antena puede reducir el rango de comunicación.

Problema: Comunicación intermitente o pérdida de paquetes

Descripción: La comunicación entre los módulos LoRa se interrumpe o los mensajes no se reciben consistentemente.

Causas Potenciales:

- Interferencia de señales cercanas.
- Colisiones de paquetes por múltiples transmisiones simultáneas.
- Falta de sincronización entre los tiempos de transmisión y recepción.

Solución:

- **Reducir interferencias:** Asegúrate de que no haya interferencias en la frecuencia usada. En áreas urbanas, el espectro puede estar congestionado.

- **Evitar colisiones:** Implementa un esquema de control de acceso, como Time Division Multiple Access (TDMA) o Carrier Sense Multiple Access (CSMA), para evitar que los módulos transmitan simultáneamente.
- **Sincronización:** Si usas una comunicación bidireccional, asegúrate de que los módulos estén sincronizados en cuanto a los tiempos de transmisión y recepción. Implementa un protocolo simple para coordinar la comunicación.

Problema: El rango de comunicación es muy corto

Descripción: Los módulos LoRa SX1278 no alcanzan el rango esperado (puede ser de varios kilómetros en condiciones ideales).

Causas Potenciales:

- Baja potencia de transmisión configurada.
- Obstáculos físicos entre los módulos (edificios, árboles, paredes).
- Mal ajuste del ancho de banda, factor de difusión (spreading factor) o tasa de transmisión.

Solución:

- **Aumentar la potencia de transmisión:** Configura la potencia de transmisión al máximo (dentro de los límites permitidos por las regulaciones locales).

`LoRa.setTxPower(20); // Potencia máxima de transmisión`

- **Optimizar la configuración:** Ajusta los parámetros de LoRa para optimizar el alcance:
 - o Ancho de banda: Un ancho de banda más bajo (p. ej., 125 kHz) mejora el alcance.
 - o Spreading Factor: Un mayor Spreading Factor (SF) mejora la sensibilidad y aumenta el alcance.

`LoRa.setSpreadingFactor(12); // Mayor alcance pero menor velocidad de transmisión`

- **Minimizar obstáculos:** Si es posible, evita colocar los módulos en áreas con muchos obstáculos que puedan interferir con la señal (paredes gruesas, edificios).

Problema: Baja velocidad de transmisión de datos

Descripción: Los datos se envían a una velocidad muy lenta, lo que afecta la eficiencia del sistema.

Causas Potenciales:

- El Spreading Factor (SF) está configurado muy alto, lo que aumenta la sensibilidad, pero reduce la tasa de datos.
- Ancho de banda de transmisión bajo.

Solución:

- **Optimización de SF:** Reduce el Spreading Factor si la velocidad de transmisión es más importante que el rango.

`LoRa.setSpreadingFactor(7); // Mayor velocidad, menor alcance`

- **Ajustar el ancho de banda:** Aumenta el ancho de banda si es posible para mejorar la velocidad de transmisión.

`LoRa.setSignalBandwidth(250E3); // Aumenta el ancho de banda`

Problema: No se recibe la señal a largas distancias

Descripción: Aunque los módulos deberían tener un largo alcance, los mensajes no se reciben cuando se intenta comunicar a largas distancias.

Causas Potenciales:

- Factores ambientales que afectan la propagación de la señal.
- Incorrecta alineación de las antenas.
- Potencia de transmisión insuficiente o pérdida de señal por mal diseño del hardware.

Solución:

- **Optimizar condiciones ambientales:** Asegúrate de que los módulos LoRa estén en un entorno libre de interferencias y obstáculos. En áreas urbanas, los edificios altos pueden bloquear las señales.
- **Mejorar antenas:** Usa antenas de mayor ganancia o más adecuadas para largas distancias. También puedes probar con una mejor ubicación de las antenas.
- **Subir potencia de transmisión:** Incrementa la potencia de transmisión, siempre y cuando cumpla con las regulaciones locales, para mejorar el alcance.

Problema: Errores en la recepción de datos (CRC errors)

Descripción: Los datos recibidos tienen errores o están corrompidos, causando fallos en la comunicación.

Causas Potenciales:

- Señal débil o interferencias que corrompen los paquetes de datos.
- Desajuste de los parámetros LoRa entre los módulos emisor y receptor.

Solución:

- **Habilitar la verificación de CRC:** Activa la comprobación de errores CRC en la configuración del LoRa para detectar y corregir posibles errores en los datos.

cpp

Copiar código

```
LoRa.enableCrc();
```

- **Ajustar parámetros de comunicación:** Asegúrate de que los parámetros de comunicación (frecuencia, ancho de banda, Spreading Factor) sean los mismos en ambos módulos para evitar desajustes.

Problema: El ESP32/ESP8266 se reinicia o se bloquea al usar LoRa

Descripción: El microcontrolador (ESP32 o ESP8266) se reinicia o se bloquea durante la transmisión o recepción de datos LoRa.

Causas Potenciales:

- Sobrecarga de memoria o problemas de energía.
- Problemas en la configuración del hardware.

Solución:

- **Optimización del uso de memoria:** Verifica que el código no esté consumiendo demasiada memoria y libera recursos innecesarios. Utiliza `ESP.getFreeHeap()` para monitorear la memoria libre.
- **Asegurar alimentación adecuada:** Asegúrate de que el módulo LoRa y el ESP32/ESP8266 estén alimentados correctamente. Los módulos LoRa pueden requerir más corriente durante la transmisión, lo que podría provocar reinicios si la fuente de alimentación no es suficiente.
- **Revisar la conexión física:** Asegúrate de que el módulo LoRa esté bien conectado al ESP32/ESP8266 y que no haya cortocircuitos o conexiones sueltas.

Problema: Fallo en la inicialización del módulo LoRa

Descripción: El módulo LoRa no se inicializa correctamente o falla al comenzar a operar.

Causas Potenciales:

- Conexión incorrecta de los pines SPI.
- El módulo no está correctamente alimentado.

Solución:

- **Verificar conexiones SPI:** Asegúrate de que los pines SPI (MOSI, MISO, SCK, NSS) estén correctamente conectados al microcontrolador.

cpp

Copiar código

```
LoRa.setPins(SS, RST, DIO0); // Asegura que los pines correctos estén configurados
```

- **Comprobar alimentación:** Asegúrate de que el módulo LoRa esté recibiendo la tensión de alimentación correcta (3.3V generalmente).

Problema: Diferencias en la calidad de la señal (RSSI)

Descripción: Las mediciones de la intensidad de la señal (RSSI) varían mucho entre transmisiones, incluso sin mover los módulos.

Causas Potenciales:

- Interferencias en la frecuencia de operación.
- Cambios en el entorno (interferencias temporales o multitrayectoria de la señal).

Solución:

- **Cambiar la frecuencia:** Si estás operando en una banda congestionada, intenta cambiar a una frecuencia diferente dentro del rango permitido.

```
LoRa.setFrequency(868E6); // Cambia a 868 MHz
```

- **Mejorar el entorno:** Asegúrate de que no haya grandes superficies metálicas, fuentes de ruido electromagnético o grandes obstáculos cerca de los módulos.

Problema: El ESP32 no se conecta a la red Wi-Fi

Descripción: El ESP32 no logra conectarse a la red Wi-Fi y el proceso de conexión nunca se completa.

Causas Potenciales:

- Credenciales incorrectas (SSID o contraseña).
- SSID oculto o seguridad no compatible (WPA3, por ejemplo).

- Canal Wi-Fi saturado o interferencia de otros dispositivos.
- Red Wi-Fi fuera de alcance.

Solución:

- **Verificar credenciales:** Asegúrate de que el SSID y la contraseña en el código coincidan con la red a la que te intentas conectar.
- **Verificación del canal Wi-Fi:** Si es posible, accede a la configuración del enrutador y cambia el canal Wi-Fi a uno menos saturado (como el canal 1, 6, o 11 para redes de 2.4GHz).
- **Red de seguridad compatible:** Revisa si la red Wi-Fi utiliza seguridad WPA3, que no es compatible con el ESP32. Intenta cambiar la configuración del enrutador a WPA2.
- **Prueba de distancia:** Coloca el ESP32 más cerca del enrutador para verificar si el problema es la distancia.

Mejora en el Código:

- Añade un temporizador de espera máxima para que el ESP32 no quede en un bucle de reconexión infinito:

```
int attemptCount = 0;

while (WiFi.status() != WL_CONNECTED && attemptCount < 10) {

  delay(500);

  Serial.print(".");

  attemptCount++;

}

if (WiFi.status() != WL_CONNECTED) {

  Serial.println("Error: No se pudo conectar a la red Wi-Fi.");

}
```

Problema: El ESP32 se desconecta de la red Wi-Fi constantemente

Descripción: El ESP32 se conecta a la red Wi-Fi pero se desconecta después de un corto período.

Causas Potenciales:

- Señal Wi-Fi débil o interferencias (otros dispositivos en la misma frecuencia, como microondas).
- Configuración de "auto desconexión" en el enrutador.
- Software del ESP32 con gestión ineficiente de la conexión.

Solución:

- **Fortalecer la señal:** Intenta mover el ESP32 más cerca del enrutador o usa un repetidor de Wi-Fi.
- **Desactivar ahorro de energía:** El ESP32 entra en un modo de bajo consumo por defecto, lo que puede causar desconexiones:

WiFi.setSleep(false);

- **Configuración del enrutador:** Verifica la configuración del enrutador para asegurarte de que no tiene un límite de tiempo de conexión o reglas de desconexión automática por inactividad.
- **Desconexiones por DHCP:** Cambia el ESP32 a una IP estática para evitar problemas con la asignación dinámica de IP (DHCP).

IPAddress local_IP(192, 168, 1, 184);

IPAddress gateway(192, 168, 1, 1);

IPAddress subnet(255, 255, 255, 0);

WiFi.config(local_IP, gateway, subnet);

Problema: Latencia Alta o Lento en la Transferencia de Datos

Descripción: Los datos enviados desde el ESP32 al servidor tardan mucho tiempo en llegar, o la latencia es alta durante las comunicaciones.

Causas Potenciales:

- Red Wi-Fi sobrecargada con muchos dispositivos conectados.
- Uso de conexiones HTTP en lugar de conexiones más eficientes como WebSockets o MQTT.
- Interferencias en la banda de 2.4GHz.

Solución:

- **Optimización de la red:** Desconecta otros dispositivos que no sean necesarios de la red Wi-Fi para reducir la carga.
- **Uso de WebSockets o MQTT:** Considera cambiar la implementación de HTTP a WebSockets o MQTT para reducir la latencia y mejorar la velocidad de transmisión en tiempo real.
- **Uso de HTTPS:** Cambia a HTTPS para conexiones más seguras y eficientes.
- **Reubicación de dispositivos:** Si hay muchos dispositivos o interferencias, intenta cambiar el enrutador a la banda de 5GHz (si es compatible).

Problema: Errores de HTTP al Enviar Datos al Servidor

Descripción: El ESP32 envía datos al servidor, pero obtiene códigos de error como 404, 500, o ningún código de respuesta.

Causas Potenciales:

- URL incorrecta o servidor no disponible.
- Error en la solicitud HTTP (falta de encabezados o formato incorrecto de los datos).
- Problemas de seguridad del servidor (certificado SSL/TLS).

Solución:

- **Verificación de la URL:** Asegúrate de que la URL sea correcta y esté activa (prueba accediendo desde un navegador).
- **Revisión de encabezados HTTP:** Verifica que los encabezados y el formato de la solicitud sean correctos.

```
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
```

- **Usar HTTPS:** Si el servidor requiere una conexión segura, asegúrate de usar HTTPS y añadir los certificados necesarios al código.

Problema: Direcciones IP Duplicadas o Conflictos de DHCP

Descripción: El ESP32 parece conectarse correctamente a la red, pero luego pierde la conexión o causa conflictos de IP con otros dispositivos.

Causas Potenciales:

- El enrutador asigna la misma dirección IP a más de un dispositivo.
- El ESP32 tiene una IP estática que ya está asignada a otro dispositivo.

Solución:

- Revisión del DHCP del enrutador: Configura el enrutador para manejar direcciones IP dinámicas (DHCP) correctamente y evitar conflictos.
- Asignación de IP estática: Establece una IP estática para el ESP32 dentro del rango permitido por el enrutador.

```
IPAddress staticIP(192,168,1,184); // IP estática para el ESP32
```

```
IPAddress gateway(192,168,1,1);
```

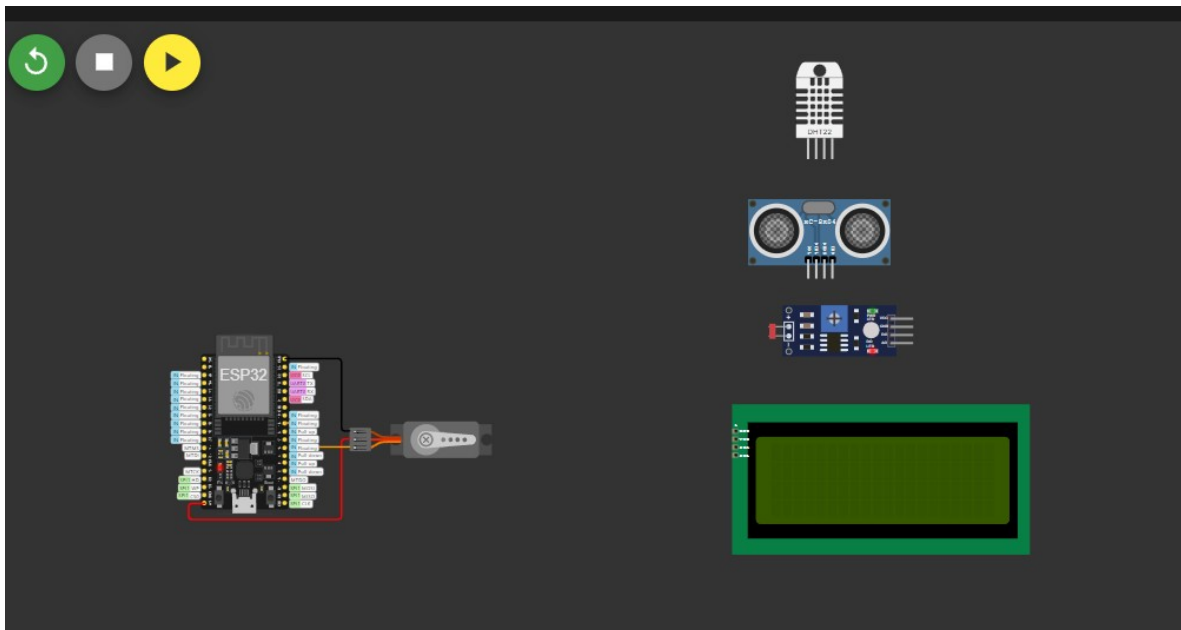
```
IPAddress subnet(255,255,255,0);
```

```
WiFi.config(staticIP, gateway, subnet);
```

Pruebas de prototipos

Ath10

Nos va a permitir obtener lecturas de temperatura y humedad, además es de bajo costo y excelente rendimiento.



Valores: humedad y temperatura

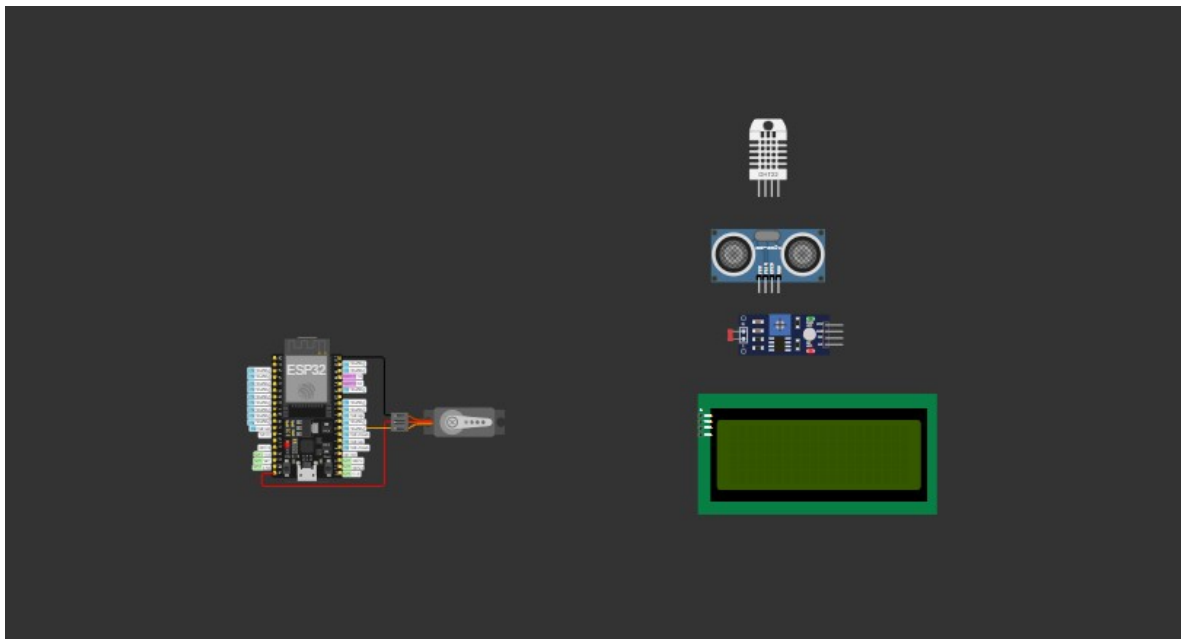
```
Temperatura: -0.00 °C, Humedad: 0.00  
Temperatura: -0.00 °C, Humedad: 0.00  
Temperatura: -0.00 °C, Humedad: 0.00
```

Los valores van a ser 0 ya que no tenemos nada incorporado

Bh1750/falta

DHT11

Nos va a permitir obtener lecturas de temperatura y humedad, como el Ath10



```
Temperatura: nan °C, Humedad: nan  
Temperatura: nan °C, Humedad: nan  
Temperatura: nan °C, Humedad: nan  
Temperatura: nan °C, Humedad: nan
```

Hc-sr04

Lo vamos utilizar para medir el crecimiento de nuestras plantas en cm(centímetros)

Valores de la distancia

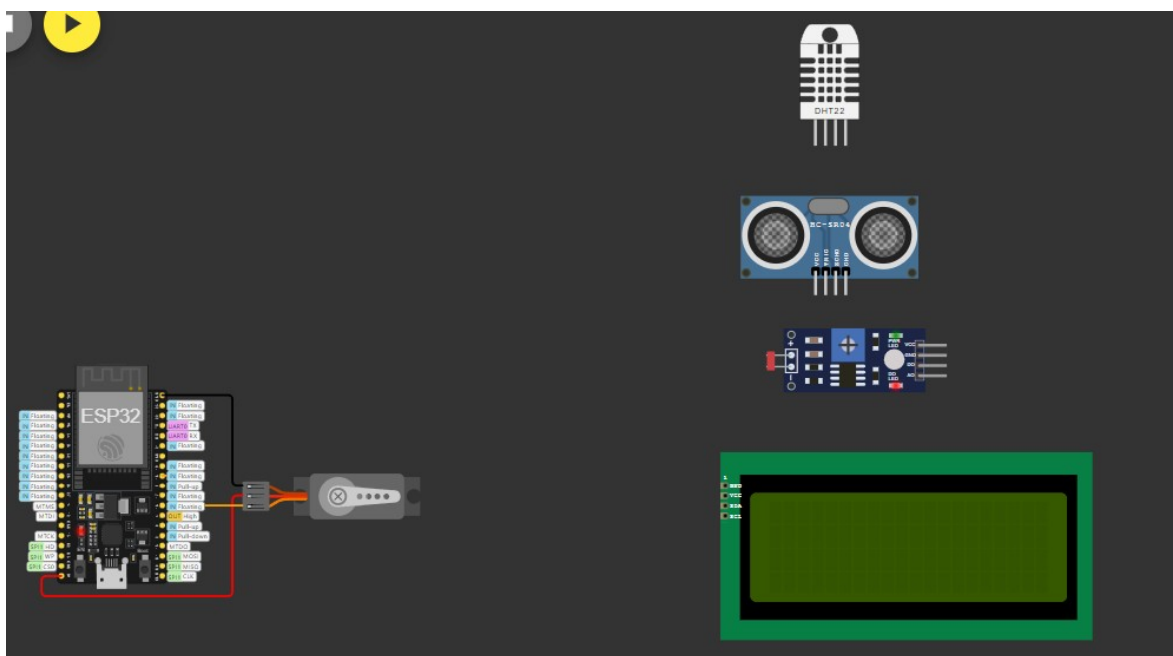
El valor de la distancia va a ser 0cm, ya que todavía no creció la planta.

Distancia: 0 cm
Distancia: 0 cm
Distancia: 0 cm

HL-69 / VER

Modulo Relé 2 Canales Optoacoplado 5v

Para encender o apagar una bomba de agua que suministre riego a la maceta, logrando una automatización en nuestro riego.

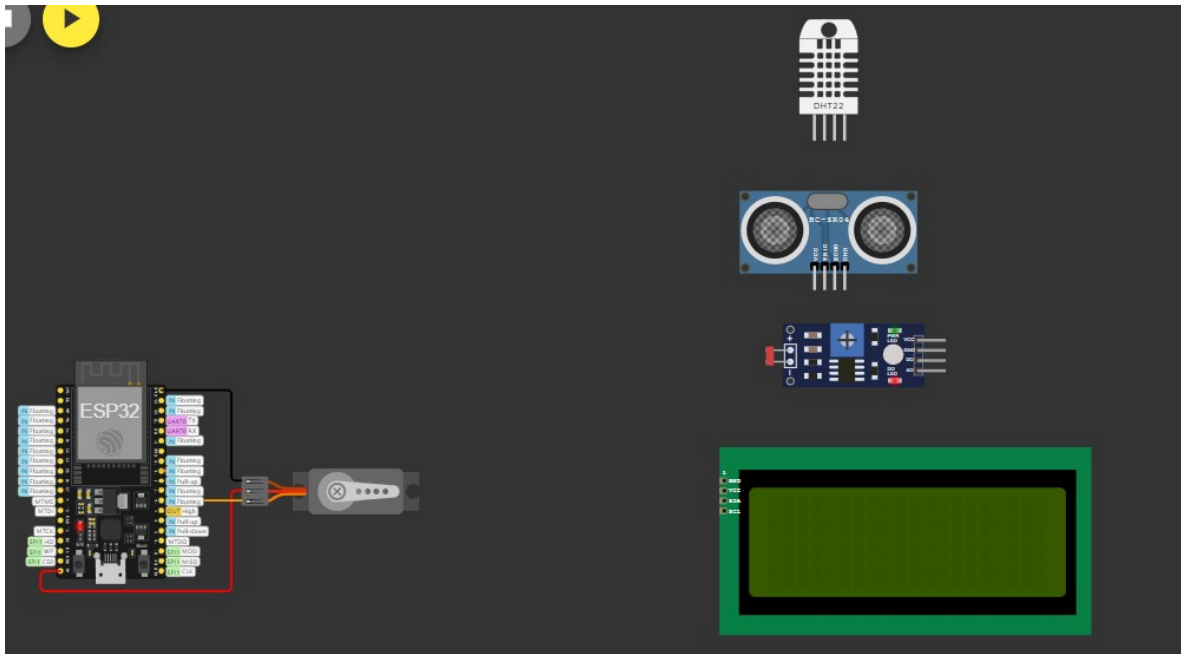


Valores: El rele se va a prender y apaga automáticamente.


```
Relé encendido.  
Relé apagado.  
Relé encendido.  
Relé apagado.
```

Modulo Sensor De Luz Con Ldr Fotoresistor

Va a ser útil, ya que las lechugas necesitan una cantidad adecuada de luz para crecer de manera saludable.



Valores: Los valores de luz nos va a ir cambiando según la iluminación que le pongamos, por ejemplo:

Con 500 lux

```
Nivel de luz: 2056  
Nivel de luz: 2185  
Nivel de luz: 2332  
Nivel de luz: 2432
```

Con 1000 lux

```
Nivel de luz: 1541  
Nivel de luz: 1713  
Nivel de luz: 1559  
Nivel de luz: 1405  
Nivel de luz: 1346  
Nivel de luz: 1511
```

Sensor lluvia

para evitar el exceso de agua. Esto es crucial para evitar el encharcamiento, que podría afectar negativamente a tus plantas de lechuga o usar el sensor para registrar la cantidad de lluvia en la maceta, además puede activar una alerta (visual o sonora) cuando empieza a llover, avisándote de las condiciones del clima para que tomes decisiones manuales.

Valores: como resultados vamos a obtener si va estar lloviendo o no, lo vamos a visualizar de la siguiente manera:

Si esta lloviendo

```
Está lloviendo.  
Está lloviendo.  
Está lloviendo.  
Está lloviendo.
```

En el caso contrario

```
No está lloviendo.  
No está lloviendo.  
No está lloviendo.
```