

Manejo de Datos en el Código client.py

1. Módulos Importados

- **network:** Proporciona la funcionalidad para conectarse a redes Wi-Fi. Permite establecer una conexión a una red específica utilizando un SSID y una contraseña.
- **socket:** Permite la creación de conexiones de red mediante el uso de sockets. Es fundamental para la comunicación TCP con el servidor.
- **machine:** Proporciona acceso a las funciones de hardware de la Raspberry Pi Pico W, como la configuración de pines y la comunicación I2C.
- **time:** Permite la gestión de temporizadores y la creación de pausas en la ejecución del programa.
- **sh1106:** Biblioteca específica para manejar pantallas OLED basadas en el controlador SH1106. Permite la inicialización y manipulación de la pantalla para mostrar texto y gráficos.

2. Definición de Constantes

- **SSID:** Define el nombre de la red Wi-Fi a la que se desea conectar la Raspberry Pi Pico W.
- **PASSWORD:** Define la contraseña de la red Wi-Fi.
- **TCP_IP:** Define la dirección IP del servidor TCP al que se enviará el conteo de interrupciones.
- **TCP_PORT:** Define el puerto en el que el servidor TCP está escuchando (3001).

3. Conexión a la Red Wi-Fi

- Se crea una instancia de `network.WLAN` y se activa como cliente (`STA_IF`).
- Se conecta a la red Wi-Fi utilizando el SSID y la contraseña proporcionados.
- Se utiliza un bucle `while` para esperar hasta que la conexión sea exitosa, imprimiendo la dirección IP asignada una vez que se conecta.

4. Configuración del Botón

- **button_pin:** Se configura un pin GPIO (en este caso, el pin 15) como entrada con una resistencia pull-up. Este pin se utilizará para detectar las interrupciones cuando se presione un botón.
- **count:** Se inicializa un contador en 0 para llevar un registro de cuántas veces se ha presionado el botón.

5. Configuración de la Pantalla I2C

- Se inicializa la interfaz I2C utilizando dos pines (SCL y SDA). Estos pines deben estar conectados correctamente a la pantalla OLED.
- Se crea una instancia de SH1106_I2C, especificando las dimensiones de la pantalla (128x64) y el objeto I2C.

6. Función de Interrupción

- `button_handler(pin)`: Esta función se ejecuta cada vez que se detecta una interrupción en el pin del botón. Incrementa el contador y muestra un mensaje en la consola.

7. Configuración de la Interrupción

- Se utiliza `button_pin.irq()` para configurar una interrupción que se activará en el flanco de bajada (cuando se presiona el botón).

8. Función para Enviar el Conteo al Servidor

- `send_count()`: Esta función se encarga de enviar el conteo actual al servidor TCP. Se establece una conexión TCP y se envía el conteo cada segundo.
- Dentro del bucle while, se limpia la pantalla, se muestra el texto "Conteo:" y el valor actual del contador en la pantalla OLED, y luego se envía el conteo al servidor.

9. Cierre del Socket

- Aunque el cierre del socket se menciona, el bucle infinito no permite que se alcance esta línea. Sin embargo, es importante recordar que en un programa bien diseñado, el socket debería cerrarse adecuadamente al finalizar.

Resumen del Manejo de Datos

- **Conexión Wi-Fi:** El código establece una conexión a una red Wi-Fi, lo que permite que la Raspberry Pi Pico W se comuniquen con un servidor TCP a través de Internet.
- **Interrupciones:** Se utiliza un botón para incrementar un contador, lo que permite la interacción física con el dispositivo.
- **Comunicación TCP:** El conteo se envía a un servidor TCP en intervalos regulares, lo que permite el monitoreo en tiempo real.
- **Pantalla OLED:** Se utiliza la pantalla I2C SH1106 para mostrar el conteo actual, proporcionando una interfaz visual al usuario.

Carga de la Biblioteca SH1106 en Thonny

Para utilizar la biblioteca SH1106 en tu proyecto, sigue estos pasos:

1. Descargar la Biblioteca:

- Ve al [repositorio de SH1106 en GitHub](https://github.com/micropython/micropython-sh1106) y descarga el archivo sh1106.py.
2. Abrir Thonny:
 - Inicia el entorno de desarrollo Thonny en tu computadora.
 3. Conectar la Raspberry Pi Pico W:
 - Conecta tu Raspberry Pi Pico W a la computadora mediante un cable USB. Asegúrate de que Thonny esté configurado para usar la Raspberry Pi Pico W como el intérprete.
 4. Cargar la Biblioteca:
 - En Thonny, ve a Archivo > Guardar como....
 - Selecciona la opción para guardar en la Raspberry Pi Pico W.
 - Nombra el archivo como sh1106.py y haz clic en Guardar.
 5. Verificar la Carga:
 - Puedes verificar que la biblioteca se haya guardado correctamente abriendo el sistema de archivos de la Raspberry Pi Pico W en Thonny y buscando el archivo sh1106.py.

Consideraciones Finales

- Asegúrate de que la biblioteca sh1106.py esté en el mismo directorio que tu archivo client.py para que pueda ser importada correctamente.
- Si experimentas problemas al cargar la biblioteca, verifica que la conexión entre la Raspberry Pi Pico W y la pantalla OLED esté configurada correctamente y que los pines SCL y SDA estén conectados a los pines correctos en la placa.

Este enfoque te permitirá utilizar la pantalla I2C SH1106 en tu proyecto y visualizar el conteo de interrupciones de manera efectiva.