

Descripción del Proyecto

Juan Pablo Vargas Córdoba

2 de septiembre de 2025

Descripción del Proyecto

El presente proyecto implementa un sistema de control básico utilizando una placa de desarrollo STM32. El programa combina la comunicación serial por UART, el manejo de interrupciones externas y el control de salidas digitales mediante LEDs.

Comunicación UART

El microcontrolador está configurado para comunicarse con el PC a través de la interfaz USART2. Cada vez que el sistema recibe un byte por el puerto serial, se procesa en una rutina de interrupción. En lugar de enviar de vuelta el mismo carácter (eco), el programa responde con una letra distinta, lo cual permite comprobar la correcta transmisión y recepción de datos. Adicionalmente, el sistema envía mensajes informativos por UART cuando ocurren ciertos eventos, como la pulsación del botón o el apagado del LED externo.

Heartbeat LED

Para indicar que el sistema se encuentra en ejecución, se utiliza el LED integrado de la placa (LD2) como un *heartbeat*. Este LED cambia de estado automáticamente cada 2 segundos, simulando un “latido” que confirma el funcionamiento normal del microcontrolador.

Interrupción por Botón

La placa incluye un botón azul (B1) que genera una interrupción externa al ser presionado. Cuando se detecta este evento:

- Se enciende un LED externo conectado a un pin de salida.

- Se establece un temporizador de 5 segundos para apagarlo automáticamente.
- Se envía un mensaje por UART notificando la acción al usuario.

Para evitar falsas detecciones, se implementa un pequeño antirrebote por software.

LED Externo con Temporizador

El LED externo permanece encendido durante 5 segundos después de la pulsación del botón. Una vez transcurrido este tiempo, el LED se apaga y se transmite un mensaje por UART indicando que el apagado se produjo por *timeout*. Este mecanismo combina el uso del temporizador del sistema (`SysTick`) con el control digital de salidas.

Conclusión

El proyecto integra de manera práctica conceptos fundamentales de los sistemas embebidos: interrupciones, comunicación serial, control de periféricos y temporización. Gracias a estas funcionalidades, el sistema puede servir como base para aplicaciones de control más avanzadas que requieran interacción entre hardware y comunicación con un dispositivo externo.