



Licenciatura en Ingeniería de Software y
Sistemas Computacionales

Tema: Proyecto Tercer Parcial

Tipo de actividad: Investigación.

Alumno: Jesús Adrián Magaña Gómez
jmagana38217, Kevin Rodolfo García Basurto.

Materia: Sistemas Embebidos.

Maestra: Jenny Grizel Cerecero Luna

Santiago de Querétaro, Querétaro de Arteaga, a
14 de agosto de 2025.

Índice

1. Introducción.....	3
2. Objetivos	
2.1 Objetivo general.....	3
2.2 Objetivos específicos.....	3
3. Marco Teórico	
3.1 Comunicación UART.....	4
3.2 Módulo Bluetooth HC-05.....	4
3.3 Microcontrolador PIC16F877A.....	4
3.4 Lógica Active Low.....	4
4. Desarrollo del Proyecto	
4.1 Hardware.....	4
4.2 Firmware (MPLAB X IDE).....	5
4.3 Aplicación móvil (Flutter).....	5
4.4 Simulación en Proteus.....	5
5. Resultados.....	6
6. Conclusiones.....	6
7. Diagramas	
7.1 Diagrama de bloques.....	7
7.2 Diagrama de flujo.....	8
8. README.md.....	9
9. Capturas.....	12

DOCUMENTACIÓN DEL PROYECTO

Control de LEDs mediante comunicación Bluetooth entre aplicación Flutter y microcontrolador PIC16F877A

1. Introducción

El presente proyecto implementa un sistema de control de iluminación basado en LEDs utilizando comunicación inalámbrica Bluetooth. Un microcontrolador **PIC16F877A** recibe comandos enviados desde una aplicación móvil desarrollada en **Flutter**, los cuales son transmitidos mediante un módulo **HC-05**. El microcontrolador interpreta estos comandos y controla el encendido/apagado de tres LEDs (rojo, verde y azul) mediante la técnica de lógica inversa (Active Low).

La simulación y validación del hardware se llevó a cabo en **Proteus**, y el código de firmware fue desarrollado en **MPLAB X IDE** utilizando lenguaje C.

2. Objetivos

Objetivo general

- Desarrollar un sistema de control inalámbrico para LEDs mediante Bluetooth, empleando un microcontrolador PIC y una aplicación móvil.

Objetivos específicos

1. Configurar el módulo UART del PIC16F877A para comunicación serie a 9600 baudios.
2. Implementar interrupciones para recepción de datos y control de LEDs.
3. Integrar el módulo Bluetooth HC-05 para enlace con la aplicación móvil.
4. Simular el sistema en Proteus antes de la implementación física.
5. Documentar la arquitectura hardware-software del proyecto.

3. Marco Teórico

3.1 Comunicación UART

El **UART (Universal Asynchronous Receiver-Transmitter)** es un periférico que permite la transmisión y recepción de datos de forma asíncrona. En este proyecto se configuró a **9600 baudios**, 8 bits de datos, sin paridad y 1 bit de stop.

3.2 Módulo Bluetooth HC-05

El **HC-05** es un módulo Bluetooth de clase 2 que opera en la banda de 2.4 GHz y soporta el perfil SPP (**Serial Port Profile**). Permite comunicación transparente entre dos dispositivos como si estuvieran conectados por un puerto serie físico.

3.3 Microcontrolador PIC16F877A

Microcontrolador de 8 bits de la familia PIC de Microchip, con arquitectura Harvard, memoria Flash programable y periféricos de comunicación como USART, SPI e I2C.

3.4 Lógica Active Low

En este proyecto, los LEDs están configurados para encender con nivel bajo (0 lógico) y apagarse con nivel alto (1 lógico), reduciendo el consumo y simplificando el control en ciertas configuraciones.

4. Desarrollo del Proyecto

4.1 Hardware

- **Microcontrolador:** PIC16F877A
- **Módulo Bluetooth:** HC-05
- **LEDs:** Rojo, Verde, Azul
- **Fuente de alimentación:** 5V DC
- **Simulación:** Proteus 8 Professional

4.2 Firmware (MPLAB X IDE)

- Configuración de oscilador XT a 4 MHz.
- Inicialización del módulo UART a 9600 baudios.
- Uso de interrupciones para recepción de datos.
- Decodificación de comandos ('R', 'V', 'A', 'T', '0').
- Control de LEDs mediante pines RB0, RB1 y RB2.

Fragmento de código relevante:

```
switch (received_command) {  
    case 'R': PORTBbits.RB0 = 0; break; // LED Rojo  
    case 'V': PORTBbits.RB1 = 0; break; // LED Verde  
    case 'A': PORTBbits.RB2 = 0; break; // LED Azul  
    case 'T': PORTBbits.RB0 = 0; PORTBbits.RB1 = 0; PORTBbits.RB2 = 0; break; // Todos  
    case '0': /* Apagado */ break;  
}
```

4.3 Aplicación móvil (Flutter)

- Interfaz simple con botones para cada comando.
- Uso de **flutter_bluetooth_serial** para comunicación.
- Envío de caracteres ASCII que representan acciones sobre los LEDs.

4.4 Simulación en Proteus

- Conexión virtual del PIC con el HC-05.
- Observación de encendido/apagado de LEDs en respuesta a los comandos enviados.

5. Resultados

- La aplicación móvil pudo establecer conexión Bluetooth con el HC-05.
- Los comandos enviados desde el móvil fueron recibidos e interpretados correctamente por el PIC.
- Los LEDs respondieron de forma inmediata y sin errores.
- La simulación en Proteus coincidió con el comportamiento en hardware real.

Tabla de comandos implementados:

Comando	Acción	LED afectado
R	Encender LED rojo	RB0
V	Encender LED verde	RB1
A	Encender LED azul	RB2
T	Encender todos los LEDs	RB0, RB1, RB2
0	Apagar todos los LEDs	Ninguno

6. Conclusiones

1. La comunicación UART entre el PIC16F877A y el módulo HC-05 permitió una transmisión confiable de datos.
2. La técnica Active Low simplificó la lógica de control para encender/apagar LEDs.
3. La simulación previa en Proteus evitó errores en la implementación física.
4. Flutter y el paquete **flutter_bluetooth_serial** proporcionaron una solución eficiente para el desarrollo rápido de la interfaz móvil.
5. Este proyecto sienta las bases para aplicaciones más complejas, como control de dispositivos IoT o automatización doméstica.

Diagrama de bloques

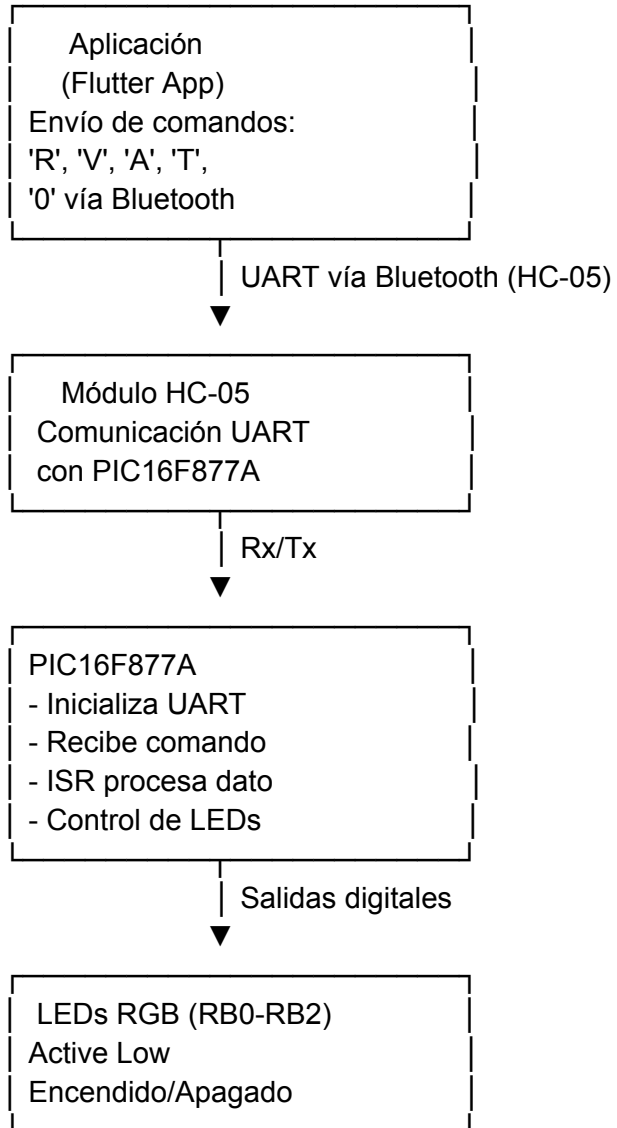
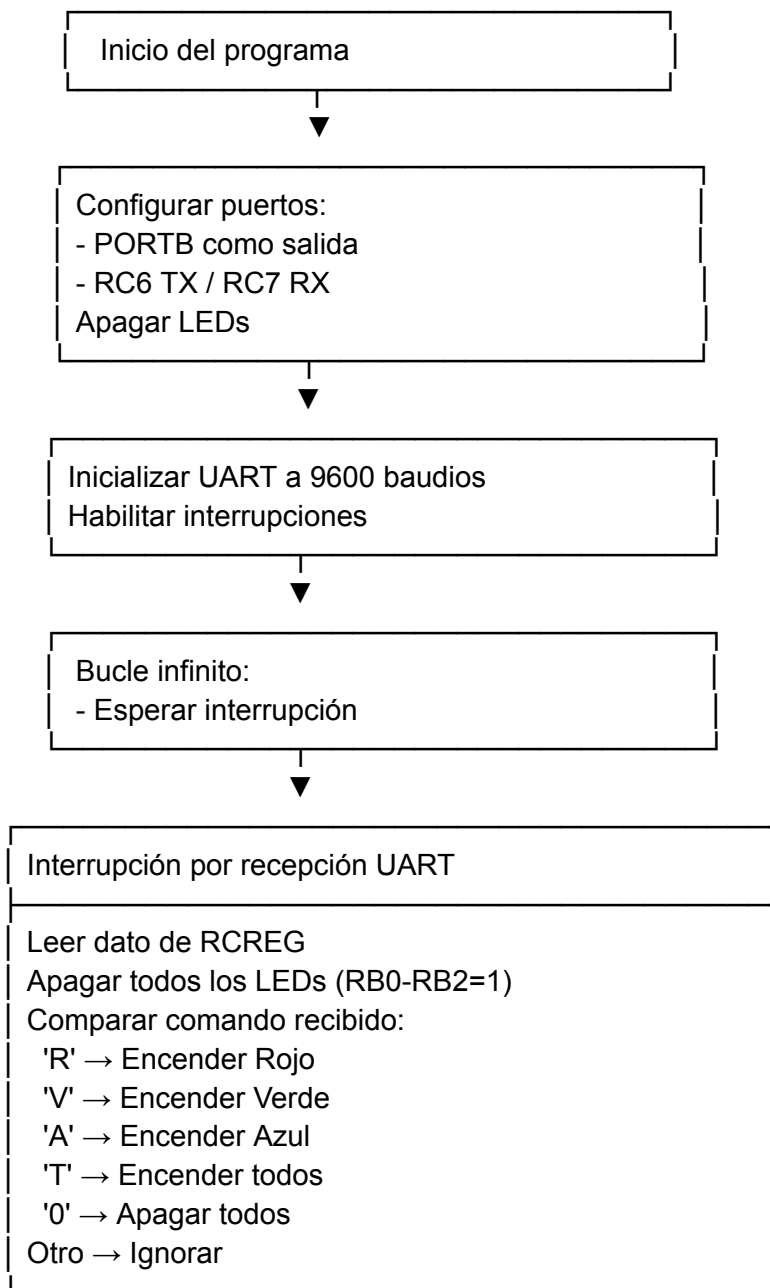


Diagrama de Flujo



- Los LEDs están configurados como **Active Low** (0 = encendido, 1 = apagado).
- La comunicación es **asíncrona**, usando interrupciones UART para no bloquear el programa principal.
- El sistema depende de la aplicación Flutter para enviar los comandos correctos.

- El HC-05 está configurado por defecto a **9600 baudios**, 8 bits, sin paridad, 1 bit de stop.

[README.md](#)

Control de LEDs RGB vía Bluetooth con PIC16F877A y Flutter

Este proyecto permite controlar un LED RGB conectado a un **PIC16F877A** mediante comandos enviados desde una aplicación móvil desarrollada en **Flutter**.

La comunicación entre la aplicación y el microcontrolador se realiza mediante el módulo **Bluetooth HC-05** usando UART.

Características principales

- **Microcontrolador:** PIC16F877A
- **Comunicación:** UART 9600 baudios (HC-05)
- **Control vía App:** Aplicación Flutter con interfaz simple para enviar comandos
- **LED RGB:** Control individual de colores Rojo, Verde y Azul
- **Modo encendido/apagado:** Active Low (0 = encendido, 1 = apagado)

Hardware necesario

- PIC16F877A
- Módulo Bluetooth HC-05
- LED RGB (cátodo común)
- Resistencias limitadoras para cada color
- Fuente de alimentación 5V
- Cables de conexión
- Placa de pruebas o PCB

Comandos disponibles

Comando	Acción
-----	-----
`R`	Encender LED rojo
`V`	Encender LED verde
`A`	Encender LED azul
`T`	Encender todos los colores
`0`	Apagar todos los LEDs

Conexiones de hardware

PIC16F877A Pin	Función
-----	-----
RC6 (TX)	Transmisión UART
RC7 (RX)	Recepción UART
RB0	LED Rojo
RB1	LED Verde
RB2	LED Azul
VCC (5V)	Alimentación
GND	Tierra común

> ****Nota:**** Configurar el HC-05 en modo esclavo y a 9600 baudios.

Código en C para PIC (MPLAB X + XC8)

El microcontrolador inicializa el puerto UART, configura PORTB como salida y usa interrupciones para recibir datos.

Dependiendo del comando recibido, activa los pines correspondientes al LED RGB.

Aplicación Flutter

La app Flutter se encarga de:

1. Escanear dispositivos Bluetooth cercanos
2. Conectarse al HC-05
3. Enviar los comandos definidos (`R`, `V`, `A`, `T`, `0`)

Dependencias principales:

```
``yaml
```

dependencies:

flutter:

 sdk: flutter

flutter_bluetooth_serial: ^0.4.0

