



Proyecto

Ingeniería en Software y Sistemas
Computacionales
8vo cuatrimestre

Alumno: Luz Andrea Piña Roa, Tadeo Mauricio
Escobar Rivera,
Lorena Regina Garcia Felix, Jesús Adrián
Magaña Gomez

Correo: tescobar38642@ucq.edu.mx,
lpina31748@ucq.edu.mx,
lgarcia32381@ucq.edu.mx,

Materia: Sistemas Embebidos

Resumen

Este documento describe el desarrollo de un sistema embebido basado en el microcontrolador PIC16F877A que integra:

- Lectura de entradas analógicas (sensor de temperatura LM35 y LDR)
- Entradas digitales
- Salida PWM para control de intensidad lumínica
- Comunicación UART con PC
- Visualización en LCD 16x2
- Control automático basado en condiciones ambientales

El sistema fue implementado tanto en simulación Proteus

Introducción

Justificación

Este proyecto integra múltiples conceptos fundamentales de sistemas embebidos:

- Procesamiento de señales analógicas/digitales
- Generación de señales PWM
- Comunicación serial
- Implementación de lógicas de control

Su desarrollo sirve como base para aplicaciones más complejas en domótica, automatización industrial y sistemas de monitoreo.

Marco Teórico

PIC16F877A

Microcontrolador de 8 bits con:

- 8KB de memoria Flash
- 368 bytes de RAM
- 256 bytes de EEPROM
- 5 puertos I/O (33 pines digitales)
- Módulos ADC, PWM, USART integrados

ADC (Conversor Analógico-Digital)

- Resolución de 10 bits (1024 niveles)
- Tiempo de conversión $\sim 20\mu s$
- 8 canales multiplexados

PWM (Modulación por Ancho de Pulso)

- Frecuencia configurable (1Hz-20kHz típico)
- Resolución de 10 bits
- 2 canales independientes (CCP1 y CCP2)

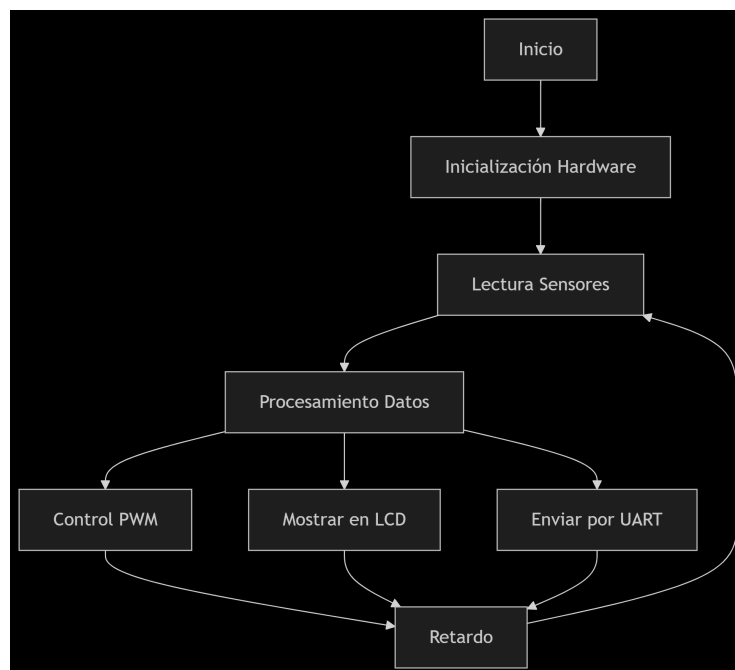
UART (Comunicación Serial)

- Protocolo asíncrono
- Velocidad configurable (300-115200 baudios)
- Soporte para comunicación full-duplex

Diseño del Sistema

Componentes principales:

- PIC16F877A (microcontrolador)
- LM35 (sensor temperatura)
- LDR + resistencia 10k Ω (sensor luz)
- LCD 16x2 (visualización)
- LED con driver PWM





Implementación del Código

Estructura Principal

```
void main() {  
    // 1. Inicializaciones  
    init_ADC();  
    init_PWM();  
    init_UART(9600);  
    init_LCD();  
    init_buttons();  
  
    // 2. Bucle principal  
    while(1) {  
        read_sensors();  
        process_data();  
        control_outputs();  
        display_data();  
        send_uart();  
        __delay_ms(100);  
    }  
}
```

Lectura de Sensores

ADC (LM35 y LDR)

```
float read_temperature() {  
    ADCON0 = 0b00000001; // Canal AN0, ADC ON  
    __delay_us(20);  
    GO = 1;  
    while(GO);  
    return ((ADRESH<<8)+ADRESL)*0.488; // Conversión a °C  
}
```

Entradas Digitales

```
#define BTN1 RB0
```

```
void check_buttons() {  
    if(!BTN1) { // Lógica pull-up  
        // Acción para botón presionado  
    }  
}
```

Generación PWM

```
void set_pwm_duty(unsigned int duty) {  
    CCPR1L = duty>>2; // 8 MSB  
    CCP1CONbits.DC1B = duty&3; // 2 LSB  
}
```

```
}
```

Comunicación UART

```
void uart_send_str(const char *str) {  
    while(*str) {  
        while(!TXIF);  
        TXREG = *str++;  
    }  
}
```

Control Lógico

```
void control_outputs() {  
    if(temperature > 30.0) {  
        set_pwm_duty(1023); // Máxima ventilación  
    }  
    else if(light < 50) {  
        set_pwm_duty(512); // Media iluminación  
    }  
    else {  
        set_pwm_duty(0); // Apagado  
    }  
}
```

Pruebas y Resultados

Pruebas en Proteus

- Funcionalidad de ldr
- No se pudo comprobar la funcionalidad del dht11 por falta de librería lcd
- El bluetooth fue eliminado del proyecto ya que no se supo como implementar a proteus y que se comprobara su funcionalidad
- El gps no se pudo comprar la funcionalidad por una advertencia de proteus con google

Conclusión

Limitaciones Encontradas

1. Sensor DHT11
 - a. Problema: No se pudo comprobar su funcionalidad por falta de librería compatible en Proteus
2. Módulo Bluetooth
 - a. Problema: Eliminado del proyecto por dificultades de implementación en Proteus
3. Módulo GPS
 - a. Problema: Advertencia de Proteus sobre integración con servicios de Google

Dificultades y Soluciones Implementadas

Limitaciones de Proteus:

- Problema: Falta de soporte para algunos componentes como DHT11
- Componentes no implementados:
- Bluetooth y GPS fueron descartados por complejidad de simulación

Recomendaciones para Futuras Mejoras

- Implementar pruebas físicas para componentes no soportados en simulación
- Considerar alternativas de simulación más completas (como LabCenter VSM)
- Desarrollar módulos de prueba independientes para cada componente
- Documentar claramente las limitaciones de la simulación vs. implementación real