

Mini Proyecto – I2C

Utilizando el microcontrolador PIC16F887 con el compilador XC8 y el microcontrolador ESP32 y servicio de Adafruit IO

Con este proyecto se busca que el estudiante experimente con el microcontrolador PIC16F887 para realizar una aplicación de una red de comunicación I2C.

El proyecto consiste en implementar una red de comunicación I2C entre un sensor I2C y el microcontrolador PIC16F887. El microcontrolador PIC obtendrá los valores del sensor, y los mandará mediante comunicación UART al Microcontrolador ESP32. El microcontrolador ESP32 se conectará por medio de WIFI al servidor de Adafruit IO. Deberá crear un dashboard en donde se pueda visualizar las variables obtenidas por el PIC. Además, tendrá dos luces piloto las cuales puede accionar desde el dashboard. Estas luces piloto están conectadas al microcontrolador PIC, por lo que, si cambian de estado en el dashboard, el ESP32 deberá mandarle mediante comunicación UART al PIC dicho cambio de estado.

Deberá implementar librerías en sus códigos.

Parte 1 Microcontrolador PIC 16F887:

Diseñe e implemente el algoritmo de comunicación I2C con el sensor que escogió. **Investigue bien como funciona el sensor seleccionado (voltajes, velocidades, direcciones, comandos, etc.).**

Parte 2 Microcontrolador PIC 16F887:

Diseñe e implemente una rutina en donde los valores obtenidos los pueda mandar mediante comunicación UART.

Parte 3 Microcontrolador PIC 16F887:

Diseñe e implemente una rutina en donde pueda recibir mediante comunicación UART dos estados de las luces piloto y con estos active o desactive los leds (luces piloto).

Parte 4 Conexión PIC y ESP32:

Construya un convertidor bidireccional de 5V a 3.3V entre los pines de comunicación UART del Microcontrolador PIC16F887 y el ESP32. **TOME EN CUENTA QUE EL ESP32 SOLO PUEDE MANEJAR 3.3V.**

Parte 5 ESP32:

Implemente la comunicación UART para adquirir los valores obtenidos del sensor I2C.

Parte 6 Dashboard Adafruit IO:

Diseñe una interfaz en los servidores de Adafruit IO, en donde pueda mostrar el resultado de los valores obtenidos del sensor I2C. Además, de colocar dos luces piloto las cuales pueda cambiar el estado desde la interfaz.

Parte 7 ESP32 WIFI:

Implemente la comunicación entre el ESP32 y los servidores de Adafruit IO, utilizando WIFI. El ESP32 deberá ser capaz de mandar los valores obtenidos del sensor I2C. También deberá obtener los estados de las luces piloto del dashboard de Adafruit IO.

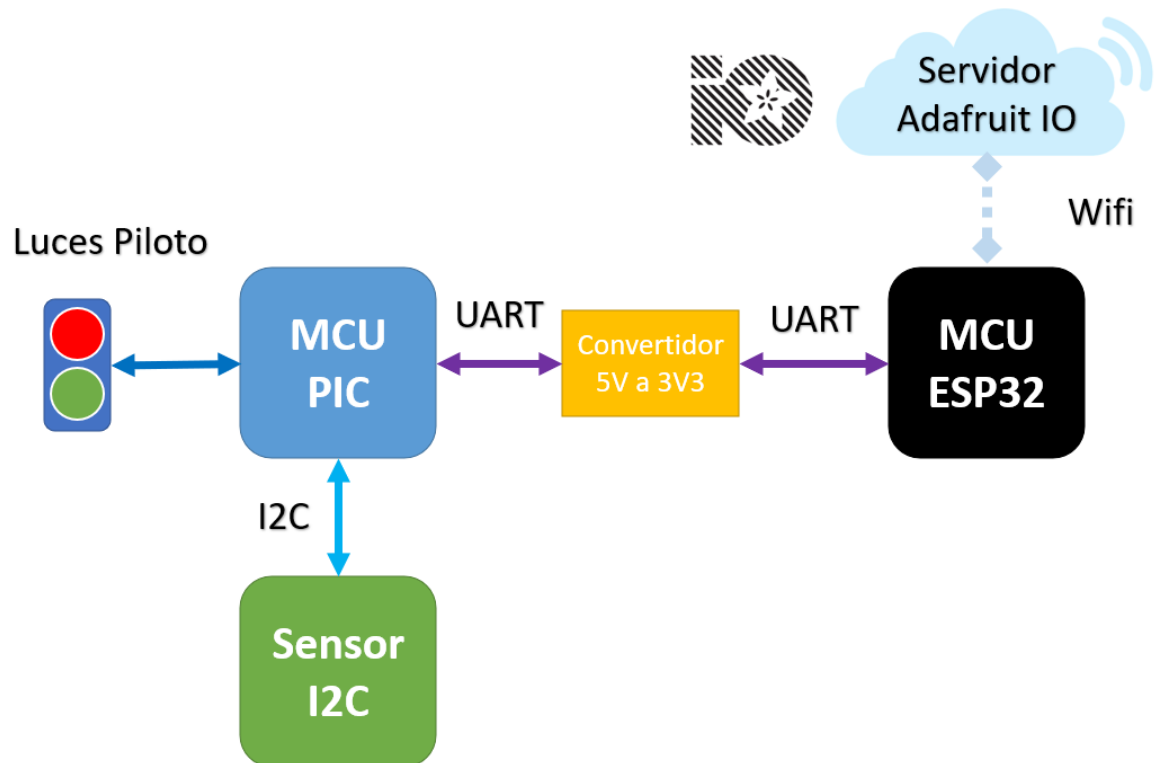
Parte 8 ESP32 y PIC16F887 (Luces piloto):

Diseñe e implemente un algoritmo en el cual a partir que exista un cambio en las luces piloto del servidor de Adafruit, el microcontrolador ESP32 le mande una notificación al PIC16F887 para que cambie de estado dicha luz piloto.

Entregables

- Diagrama de flujo general del proyecto y pseudocódigos para cada uno de los microcontroladores involucrados.
- Link de Repositorios de GitHub con códigos debidamente comentados, con múltiples commits, de forma que se pueda evidenciar el progreso paulatino del proyecto.
- Proyectos de MPLAB X del microcontrolador y Proyecto del ESP32
- Video donde se explique el funcionamiento del proyecto
 - o Explicación de código
 - o Deberá salir explicando el circuito físico

Figura 1. Circuito



Ponderación

- | | |
|---|------------|
| • Funcionamiento | 80% |
| ○ Parte 1: Comunicación I2C | 30% |
| ○ Parte 3: Luces piloto | 10% |
| ○ Parte 2,4 y 5: Comunicación UART | 10% |
| ○ Parte 6: Dashboard Adafruit IO | 10% |
| ○ Parte 7 y 8: ESP32 comunicación Adafruit IO | 20% |
| • Video y Documentación | 20% |

TOTAL 100%

Punteo

15 puntos netos