

Diego Estrada (19264); Jonathan Menéndez (18023); Ingeniería Mecatrónica, UVG

## Dispensador de Bebida y Alimentos

Proyecto 1

**Resumen**—*El siguiente reporte resume brevemente la fabricación de un dispensador de bebidas y alimentos por medio de una red de sensores programados utilizando tres PIC16F887, un maestro y dos esclavos con comunicación I2C. Los sensores utilizados son un sensor de luz, uno de temperatura y la comunicación con un LCD con I2C por medio de un módulo con un PC8574AT para desplegar valores variados de los sensores y motores utilizados. Además, se incluye un motor DC y un motor Servo para ejecutar las acciones de dispensar los productos.*

### I. Descripción de la Función de cada PIC

#### Maestro:

El funcionamiento del microcontrolador maestro consiste en recibir los datos de un contador y de un sensor de temperatura, los cuales se encuentran en los esclavos 1 y 2, por medio de la comunicación I2C. También se implementó un motor sumergible DC para el dispensador de agua, el cual, es controlado por medio de un botón para encenderse y apagarse. Además, también se implementó una LCD con un módulo de comunicación I2C para el control de caracteres que esta muestra, los cuales, son los correspondientes a: el contador que muestra las porciones disponibles de comida, el estado del motor sumergible para el dispensador de agua (encendido o apagado) y el valor del sensor de temperatura. Por último, se implementó la comunicación UART por medio de una librería para poder enviar los datos recolectados por los sensores hacia la computadora.

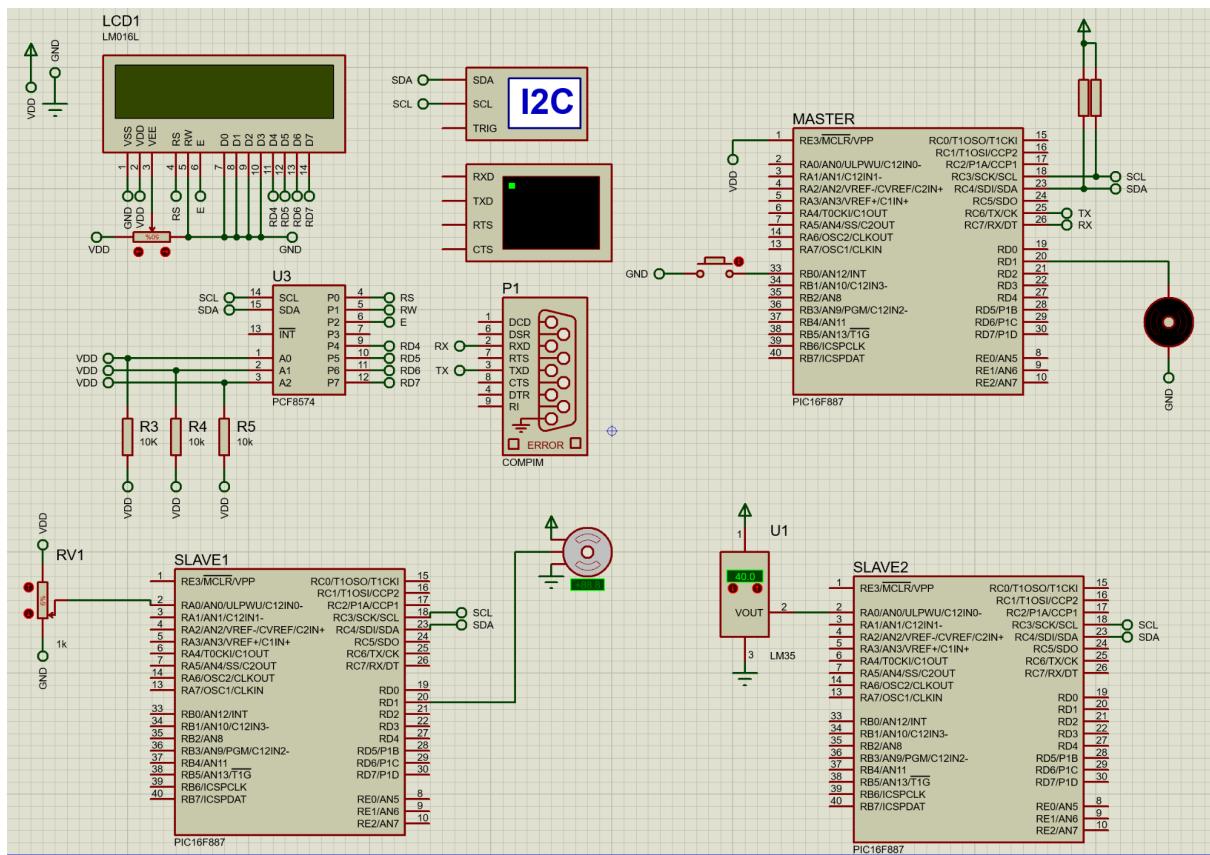
#### Esclavo 1:

El propósito del primer esclavo era controlar el dispensador de comida por medio de un sensor de luz el cual brinda una señal analógica capaz de mover el servo motor. Para esto se utilizó bit banging para que el servo motor tuviera únicamente dos posiciones las cuales permitirían la salida de alimentos por porciones. Luego, se implementó un contador que empieza en cero y cada vez que se utiliza va sumando uno, cuyo resultado es enviado por medio de la comunicación I2C hacia el microcontrolador maestro cuya programación se encuentra dada por una librería para llevar a cabo esta comunicación. El resultado de este contador luego es restado a una variable creada en el microcontrolador maestro la cual empieza con un valor de 10 y, de esta forma, poder llevar un control en cuanto a cuántas porciones de comida hay disponibles.

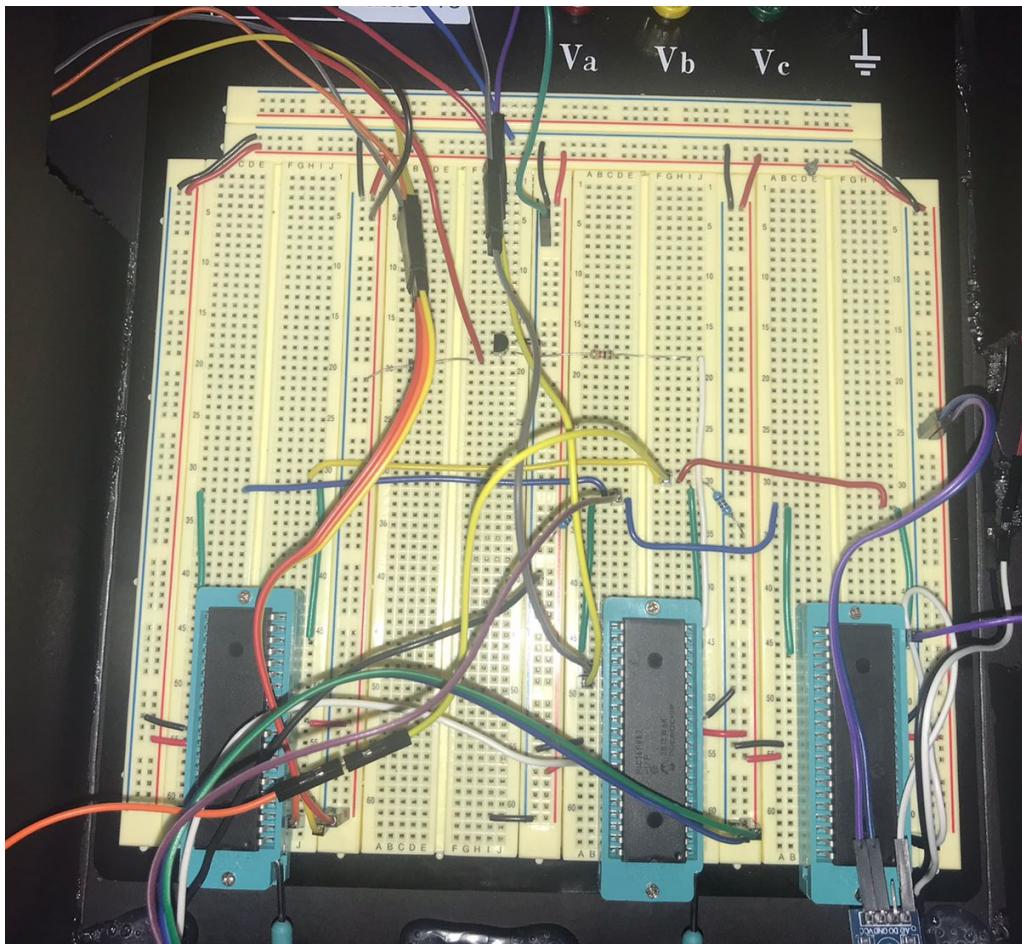
#### Esclavo 2:

El segundo esclavo tiene como función leer el valor entregado por un sensor de temperatura colocado en el contenedor de almacenamiento de la bebida, esto es para poder controlar que el contenido se encuentra a una temperatura deseada. El sensor es utilizado como una entrada analógica, por lo que se da uso del módulo ADC del microcontrolador. El valor leído por la entrada analógica seleccionada y luego es presentada en el puerto D, cuyo valor luego es transferido a una variable en el microcontrolador maestro a través de la comunicación I2C, cuya aplicación se encuentra dada por una librería con la programación necesaria para implementarla.

### Simulación en Proteus:



Círcuito en Físico:



Dispensador armado:



## II. Descripción de programa Python y plataforma Adafruit IO

### Programa de Python:

Para poder realizar la comunicación serial con el módulo FTDI con el PIC16F887 maestro se utilizan las librerías serial y time. Además, para poder transmitir los datos a la plataforma de Adafruit IO se utiliza la librería Adafruit\_IO instalada previamente. En la primera parte del código se dan las configuraciones generales de la comunicación serial, donde se selecciona el puerto a utilizar y el baud rate deseado. Luego se tiene la configuración con la cuenta de Adafruit con la que se desea comunicar.

Para poder mantener una comunicación serial constante se coloca un loop por medio de un while. Dentro de este se inicia la comunicación serial, se lee y asigna en tres diferentes variables los valores obtenidos. Luego por medio de la instrucción de aio.feeds se asigna el nombre para el programa y la contraseña de cada sensor utilizado. Después, se utiliza la instrucción de aio.send\_data para enviar los valores de las variables con las lecturas de la comunicación serial hacia los feeds y dashboard correspondientes en la plataforma de Adafruit IO.

### Interfaz del Dashboard de Adafruit IO:



## III. Enlaces

- Video de YouTube:

<https://www.youtube.com/watch?v=A0g9J4PIOmQ>

- Enlace de Repositorio en Github:

[https://github.com/men18023/Proyecto1\\_Digital2](https://github.com/men18023/Proyecto1_Digital2)

- Enlace de plataforma Adafruit IO:

[https://io.adafruit.com/diego\\_estrada/public](https://io.adafruit.com/diego_estrada/public)