Universidad del Valle de Guatemala
Organización de Computadores y Assembler
Sección no. 30
Docente: Kimberly Barrera

Proyecto No.3: Diseño programas en lenguaje Ensamblador para el control de circuitos

Alejandro Gomez 20347
Marco Jurado 20308

Índice

Introducción:

Como parte del curso de Organización de Computadores y Assembler, se presenta este proyecto, basado en el diseño de programas en lenguaje ensamblador para el control de circuitos. Como parte de la ejecución de este proyecto se utilizaron los conocimientos del lenguaje de programación ARM Assembly en donde se utilizaba WiringPi para poder utilizar los puertos GPIO de la consola Raspberry Pi 4b. Así mismo se utilizó el conocimiento adquirido para organizar el protoboard y tener el equivalente a un display de 7 segmentos con leds y un dip switch. Se manejó un control de versiones en la plataforma de GitHub, para un mejor manejo del programa.

Resolución de temario:

Para la resolución del temario, primero que todo se definió los puertos GPIO a utilizar. Se tomó como referencia el laboratorio anterior, lo cual facilitó bastante el trabajo. Debido a esto, solamente fue necesario incluir un pin más, correspondiente al dip-switch implementado en este proyecto. Ahora, en cuanto a la resolución del temario. Primero que todo, se definió los símbolos a representar en los leds. En este caso, las letras: A, b, C, d, E, F. Después de organizar los pines para cada uno de los segmentos de leds, se procedió a realizar las modificaciones pertinentes al código del laboratorio 10. Esto consistió en la adición de un menú el cual implementa el dip-switch. Indicándole así al programa, cuando ejecutarse. Se tomó en cuenta la programación defensiva para evitar cualquier inconveniente al ingresar una letra errónea. Cumpliendo con los requerimientos del proyecto, se realizó el montaje físico, agregando el dip-switch al montaje realizado para el laboratorio 10. Se verificó varias veces el funcionamiento para asegurarse que no existiera ningún tipo de error.

Información de diseño:

En cuanto a decisiones de diseño, se optó por usar un montaje bastante similar al realizado para el laboratorio 10. En vez de utilizar un segmento de 7 leds, se optó por, nuevamente, usar 8 leds por separado. Sin embargo, para este proyecto, se agregó, al circuito ya realizado, un dip-switch. Cumpliendo así, con los requerimientos planteados en la rúbrica.

Tabla No.1: Organización de puertos GPIO

	Puertos GPIO	PIN	Segmento de LED
Puertos 0-9	0	11	D
	1	12	E
	3	15	G
	4	16	F
	5	18	B
	6	22	DP
Puertos 20-29	21	29	A
	22	31	C
	25	37	Dip Switch
GROUND	GND	9	Ground

Pseudocódigo:

1. Se definen puertos GPIO a utilizar al inicio del programa.
    1.1. A cada puerto GPIO se le asigna un un segmento correspondiente del display de 7 segmentos. Para esto se utiliza el comando gpio readall en la consola para determinar los pines. Se puede observar la fig. 3 el cómo se encuentran ubicados dichos pines.

    1.2. Por cada segmento, se define una variable correspondiente a su pin. Esta tiene el nombre de “pinX”
2. Se define el align y se inicia el data para comenzar a definir las variables a utilizar en el código.
    2.1. Se definen los pines como int y están acompañados del valor numérico del pin en la raspberry.
    2.2. Se define el delay a utilizar
    2.3. Se definen las variables necesarias para el uso del dip switch.
3. Se declara el main:
    3.1. Para inicializar wiringPI se llama a librería.
    3.2. Se verifica si existe algún error.
        3.2.1. En caso de existir un error, el programa no se inicializa.
        3.2.2. En caso no exista error, se continúa con la corrida del programa.
4. Se declara el init:

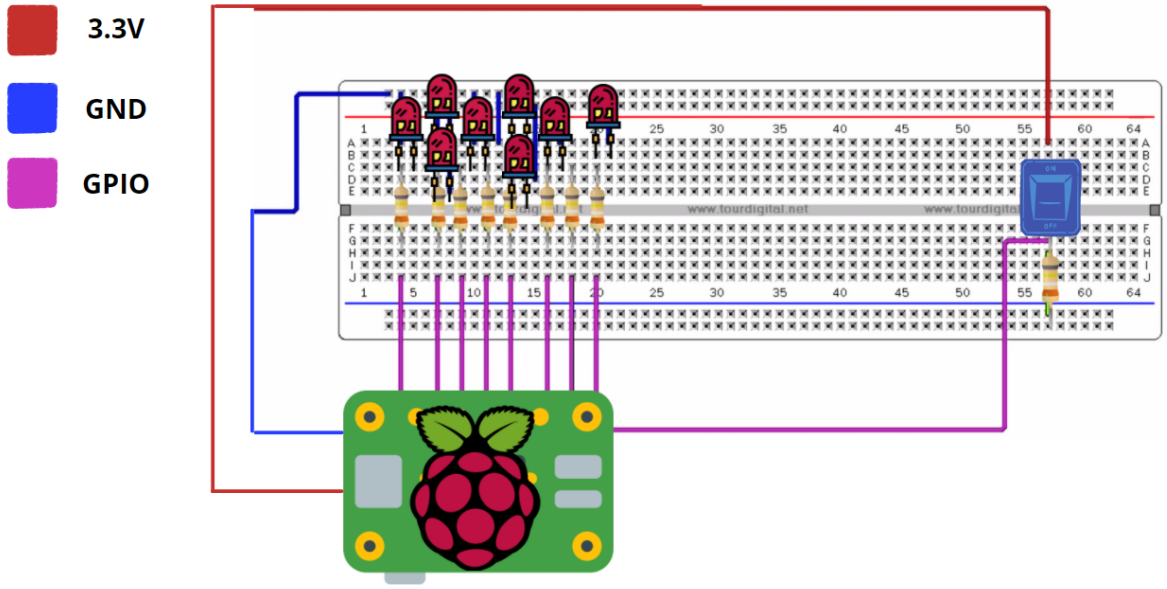
4.1. Por cada pin a utilizar, se le asigna el modo de salida o entrada y se llama a la función de WiringPI para la configuración del pin.
4.2. Se repite proceso para cada uno de los pines.

5. For loop donde se corre la secuencia de encender y apagar los leds
   5.1. Condición inicial para que al momento de ingresar la letra Y se inicie el loop siempre verificando si el valor de este cambia a Q para terminar la secuencia.
   5.2. Se da la instrucción de encendido a los pines correspondientes para formar la letra necesaria para siempre teniendo en cuenta que el dip switch debe de estar activo para permitir que la secuencia corra.
   5.3. Luego de formar cada una de las letras, se apagan todos los leds y se corre la subrutina del delay. Posteriormente se inicializa la siguiente secuencia para formar la letra que corresponde.
   5.4. Se repite el proceso hasta finalizar con las vocales.
6. DONE
   6.1. se hace el pop correspondiente para terminar el programa.

Repositorio en GitHub:

Para el desarrollo de este proyecto, se implementó GitHub para llevar un control de versiones ordenado y mantener el historial de los cambios realizados. Esta herramienta facilitó bastante el trabajo grupal a distancia, debido a que, mientras se iba trabajando por una sesión colaborativa en Visual Studio Code, cada miembro iba realizando los cambios que creía pertinentes, logrando así un trabajo colaborativo y equitativo. Cabe destacar que cada uno de los cambios realizados fue hablado a tiempo real en Zoom mientras se programaba. El enlace para ingresar al repositorio de GitHub se adjunta a continuación: https://github.com/aleg001/Proyecto3Assembler

Diagrama de circuito:



Listado de componentes:

Cantidad	Componente
8	Leds
1	Dip Switch
1	Protoboard
?	Jumpers Hembra-Macho
1	Raspberry PI 4B
?	Jumpers Macho-Macho
8	Resistencias 330
1	Resistencias 10K

Figuras:

Figura. 1: Temario seleccionado para el proyecto

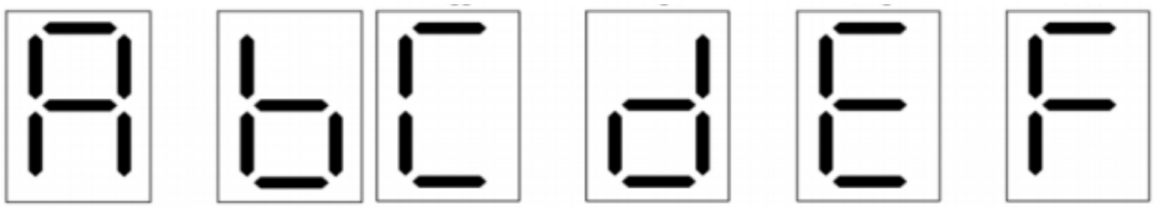


Figura. 2: Representación del temario en el display de 7 segmentos.

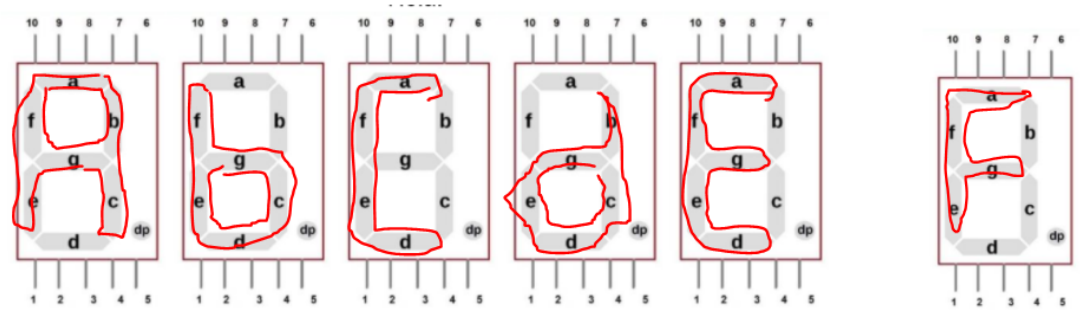
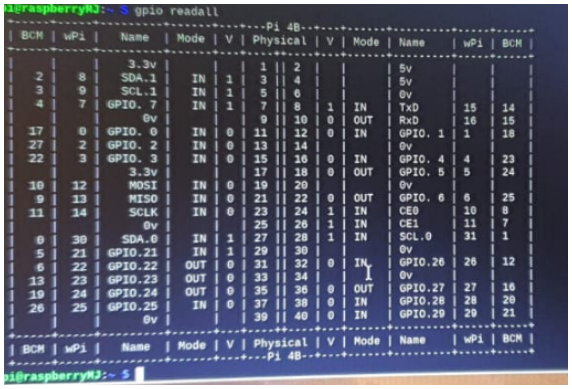


Figura. 3: GPIO Readall



Conceptos técnicos:

- GPIO:
  - GPIO es definido como Entrada y Salida de propósito general. Básicamente corresponde a los pines que se encuentran en la placa, en este caso, la RaspBerry PI.

- Dip-Switch:

    - Es un conjunto de interruptores utilizados para modificar o mandar una señal al hardware.

- Diodo:
    - Es definido como un dispositivo que permite el paso de la corriente en un solo sentido. Al recibir corriente eléctrica muestra luz, del color correspondiente al dispositivo semiconductor.

- Secuencia:
    - En términos generales podemos definir una secuencia como el orden que se lleva a cabo con una serie de pasos que suceden unos a otros.

- Señal:
    - Por definición, una señal es el diferencial de potencial entre dos puntos que se cargan de forma eléctrica en un determinado tiempo.

- Raspberry Pi:
    - Esta es una consola de pequeño tamaño y bajo costo capaz de soportar el sistema operativo de Linux y ser controlada desde las salidas de video mediante un monitor y de entrada con mouse y teclado. Esta permite el uso de sensores para poder interactuar con el mundo externo de una manera sencilla y efectiva.

Conclusiones:

- Por medio de la programación en el lenguaje ensamblador, es posible controlar un circuito físico hecho en una protoboard. Por lo tanto, se cumple con los objetivos del curso respecto al diseño de programas en el lenguaje ensamblador.

- Se verifica que, empleando los conceptos vistos en clase y la investigación realizada, la utilización de librerías externas como lo es WiringPI facilita bastante el trabajo de códigos en un lenguaje de programación de bajo nivel.

- Se cumplió con los objetivos propuestos para este proyecto debido a que el temario asignado se completó sin ningún inconveniente. Así mismo, se recomienda tener precaución al momento de realizar las conexiones físicas.

Agradecimientos:

Se agradece primero que todo, a nuestras familias, por darnos la oportunidad de estudiar la carrera de Ingeniería en Ciencias de la Computación y Tecnologías de la Información en la Universidad del Valle de Guatemala. Por animarnos y apoyarnos durante cada etapa de nuestras vidas. Así mismo, se agradece profundamente a Kimberly Barrera, por su dedicación al curso y cada uno de los alumnos.

En cada oportunidad que se requirió de apoyo extra, Kimberly estuvo presente para apoyarnos y ayudarnos. También, muchas gracias a Ayleen, por estar siempre escuchando cada uno de nuestros conflictos dentro de Assembler y darnos el apoyo necesario.

Bibliografía:

¿Qué es Raspberry Pi? Raspberry Pi. (2019, September 27).https://raspberrypi.cl/que-es-raspberry/

RPI. (s. f.). GPIO - Raspberry Pi Documentation. GPIO. Recuperado 27 de mayo de 2021, de https://www.raspberrypi.org/documentation/usage/gpio/

Área Tecnología. (s. f.). Diodo LED Cómo funciona Características y Ventajas. DIODO LED. Recuperado 27 de mayo de 2021, de https://www.areatecnologia.com/electronica/como-es-un-led.html

- Asale, R. (s. f.-b). secuencia | Diccionario de la lengua española. «Diccionario de la lengua española» - Edición del Tricentenario. Recuperado 27 de mayo de 2021, de https://dle.rae.es/secuencia

SMITH, Bruce.Raspberry Pi Assembly Language Beginners. Hands on Guide. 2013. Recuperado 2 de junio de 2021.

Computer Hope. (2017, 15 septiembre). What is a DIP Switch? What Is a DIP Switch? https://www.computerhope.com/jargon/d/dipswitc.htm