

Laboratorio #4

GPIO - Salida

– Temario B –

CC4010 – Taller de Assembler

Semestre 2, 2015

Facultad de Ingeniería

Departamento de Ciencias de la Computación

Competencias a desarrollar

En este laboratorio los estudiantes manejarán la salida de datos mediante los puertos GPIO utilizando una conexión a un circuito con LEDs. El presente laboratorio se realizará en **PAREJAS** según los grupos indicados en blackboard y durante el período de clase.

Materiales

- Raspberry Pi
- Cargador con microUSB. 5 voltios. 700 a 800 mA para la Raspberry Pi
- Tarjeta SD
- 1 protoboard
- 6 LEDs
- 6 resistencias de 270Ω
- 7 jumpers macho-hembra
- Jumpers macho-macho

Circuitos Básicos

El Protoboard es una herramienta que permite armar y modificar circuitos rápida y prácticamente. Esta herramienta se encuentra en diversos tamaños y consiste de una matriz de agujeros cuyas filas tienen continuidad entre sí tal como lo muestra la Figura No.1. Comúnmente, en un protoboard se utilizan resistencias, capacitores, LEDs, botones, circuitos integrados, entre otros para armar circuitos más complejos. Una de las mayores ventajas de los protoboards es que su estructura de agujeros permite probar circuitos sin necesidad de que éste sea permanente, pues se pueden conectar muchos dispositivos e intercambiarlos fácilmente.

Figura No.1 Continuidad en un Protoboard. **Fuente:**

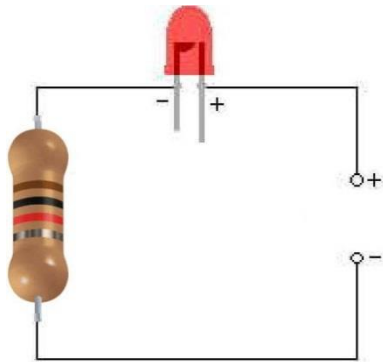
<http://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/turing-machine/two.html>



Un LED (Light Emitting Diode por sus siglas en inglés), es un dispositivo electrónico que emite luz cuando se conecta una fuente de voltaje en sus terminales. Es importante notar que el LED, por ser un diodo, debe conectarse en la orientación adecuada para que funcione. La Figura No.2 muestra la forma correcta de conectar un diodo LED.

Figura No.2 Conexión de un LED. **Fuente:**

<http://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/turing-machine/two.html>



	Lógica Inversa	Lógica Directa
Terminal +	3V	Puerto GPIO
Terminal -	Puerto GPIO	0V (GND)
Enciende cuando:	Puerto GPIO = 0	Puerto GPIO = 1
Se apaga cuando:	Puerto GPIO = 1	Puerto GPIO = 0

Instrucciones

Su tarea será implementar una pequeña **serie navideña** en lenguaje ensamblador de ARM:

- En un Protoboard conecte 6 LEDs con su resistencia como se indica en la Figura No.2.
 - Revise en blackboard qué número de grupo le fue asignado. Los grupos pares deberán implementar la serie navideña utilizando una **lógica inversa** y los impares utilizando una **lógica directa**.
- Conecte las terminales positivas y negativas de cada LED según lo indicado en la Figura No.2. No olvide que cada LED deberá ser conectado a un puerto GPIO diferente. Puede utilizar los puertos GPIO que considere convenientes, pero indique en su código cuál está utilizando para cada LED¹.
- Su programa debe manejar una serie navideña siguiendo una secuencia predeterminada. Tome en cuenta los siguientes requerimientos que debe cumplir su programa:
 - Su programa debe implementar subrutinas o macros propios.
 - Utilice systemTimer.s para controlar el tiempo de encendido y apagado de los LEDs del semáforo.
 - Utilice el archivo gpio.s para encender y apagar los LEDs correspondientes.
 - Cree un archivo **main.s** que configure los puertos GPIO como salidas utilizando el archivo gpio.s y utilice sus subrutinas o macros para implementar la funcionalidad solicitada.
 -

¹ Algunos puertos que puede utilizar sin problema son los siguientes: 7, 8, 9, 10, 11, 14, 15, 17, 18, 22, 23, 24, 25

4. Ejecute de forma infinita la siguiente secuencia². Utilice un delay de 0.5s entre cada uno de los estados:

Serie	Duración (s)
○ ○ ● ● ○ ○	0.5
○ ● ○ ○ ● ○	0.5
● ○ ○ ○ ○ ●	0.5
○ ● ○ ● ○ ●	0.5
● ○ ● ○ ● ○	0.5
○ ● ○ ● ○ ●	0.5
● ○ ● ○ ● ○	0.5
○ ○ ○ ○ ○ ○	0.5
● ● ● ● ● ●	0.5
○ ○ ○ ○ ○ ○	0.5
● ● ● ● ● ●	0.5
○ ● ● ● ● ○	0.5

Figura No.3 Pines de la Raspberry Pi B



² En amarillo se muestran los LEDs que deben estar encendidos en cada estado.

Material a entregar

- Todos los archivos de su carpeta *source*
- *kernel.img*
- **Video:** URL del video, donde se observe a los dos integrantes del grupo explicando:
 - el funcionamiento de su programa en la Raspberry y,
 - la lógica de las conexiones asignadas (directa o inversa).

Rúbrica del Laboratorio 4

Criterios de evaluación	Nivel Distinguido
Funcionamiento del Programa (grupal) 30%	El programa funciona con todos los requerimientos solicitados.
Orden y cableado del Circuito (grupal) 10%	El orden del circuito permite la fácil identificación de las rutas de datos; todas las conexiones se están completamente ordenadas y se encuentran sobre la superficie del <i>protoboard</i> . Si utilizan <i>jumpers</i> , están colocados de forma cuidadosa y utilizando código de colores.
Documentación y orden del programa (grupal) 5%	La documentación del programa incluye encabezado y comentarios representativos en los bloques de código más importantes. Los nombres de las variables son significativos, utilizando alias para los registros. La presentación del programa es muy clara y ordenada, y utiliza una tabulación adecuada.
Uso de subrutinas y/o macros (grupal) 15%	El programa es 100% estructurado con subrutinas y/o macros que hacen que el código sea más eficiente, comprensible y ordenado.