

ENTRADA Y SALIDA DE DATOS CON RASPBERRY PI

Frías Rashel, Guanoluisa Fernanda, Gallegos Jorge

mrfrias@espe.edu.ec, cfguanoluisa@espe.edu.ec, jgallegos@espe.edu.ec

Resumen – La Raspberry PI aparece con la necesidad de incentivar y promover la enseñanza de la programación para los estudiantes ya que causa interés y curiosidad, al ser un dispositivo del tamaño de la tarjeta de crédito, y debido a su capacidad prácticamente es un pequeño ordenador muy útil en proyectos de electrónica, por tanto, los campos de aplicación del Raspberry Pi son múltiples y posee numerosas posibilidades de uso generales por lo que es necesario cierto conocimiento para su programación. En base los puntos planteados se pretende realizar un circuito que permita visualizar ejemplos de entrada y salida de datos utilizando la Raspberry Pi para realizar los circuitos tanto de entrada y salida de datos utilizamos el simulador Wylidrin Studio ya que este nos permite utilizar sus plantillas y se trabaja en la programación del código para modificar el funcionamiento de cada circuito para el presente trabajo de investigación se hizo uso del lenguaje de programación JavaScript.

Palabras clave: Raspberry Pi, circuitos, entrada y salida de datos, Wylidrin Studio, JavaScript.

I. INTRODUCCIÓN

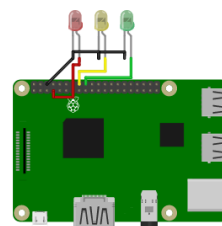
En este trabajo se realizó el desarrollo de un par de circuitos que permiten visualizar la entrada y salida de datos con la Raspberry Pi mediante el simulador Wylidrin Studio.

El primer circuito que se presenta es el de un pulsador de entrada con salida en led titilante y el segundo esta aplicado a un pequeño semáforo que combina 3 leds de salida.

Ambos circuitos propuestos fueron desarrollados por medio de plantillas existentes en el repositorio del simulador Wylidrin Studio junto con el uso de la librería GPIO en la Raspberry Pi.

II. PROCEDIMIENTO

Circuito de Entrada de Datos



| Pins | Name | Color |
|------|------|--------|
| 23 | LED | YELLOW |
| 24 | LED | GREEN |
| 4 | LED | RED |

Este circuito nos permite ver un Diodo LED parpadeando indefinidamente en intervalos de 1 segundo hasta cuando presionamos el botón del pulsador.

Creamos un objeto al cual le asignamos el paquete que vamos a usar, para poder controlar los pines de la raspberry debemos usar la librería "onoff".

Creamos tantos objetos como elementos electrónicos tengamos e inicializamos los mismos mandando como parámetro el pin GPIO al cual está conectado el elemento además debemos indicar el flujo de datos de cada elemento (entrada - in, salida - out), para nuestro ejemplo el LED representa el flujo de salida de datos y el pulsador el flujo de entrada de datos.

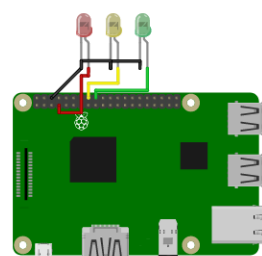
CICLO WHILE

Como sabemos el ciclo while es una estructura repetitiva que se ejecuta indefinidamente mientras una cierta condición sea verdadera, cuando esta condición sea falsa se rompe el ciclo.

La condición que definimos es que mientras no oprimamos el botón del pulsador, es decir la variable botón tiene guardado un valor de "0", el ciclo se sigue ejecutando y el LED sigue parpadeando indefinidamente, cuando pulsamos el botón le asignamos el valor de "1" a la variable

botón haciendo que el ciclo while se rompa y termine la ejecución del programa.

Circuito de salida de datos



| Pins | Name | Color |
|------|------|--------|
| 23 | LED | YELLOW |
| 24 | LED | GREEN |
| 4 | LED | RED |

En circuito fue configurado con la idea de simular el funcionamiento de un semáforo en estado normal y en estado preventivo. Los diodos Led se encienden de igual forma que un semáforo y cada LED permanece encendido durante 3 segundos, luego se apaga y se enciende el siguiente LED. Este proceso se repite 2 veces. Posteriormente luego de haber funcionado como un semáforo en estado normal el circuito el circuito simula el comportamiento de un semáforo en estado preventivo, es decir, parpadea solo el led de color amarillo 5 veces y finaliza la ejecución del programa.

Empezamos creando un objeto que nos permite controlar los pines de la raspberry al cual le asignamos el paquete "onoff" como se hizo en el ejemplo anterior.

Creamos 3 objetos de tipo Gpio uno para cada LED y le asignamos los pines de la raspberry a los cuales están contados. En este caso particular todos los elementos del circuito cumplen la función de salida de datos.

CICLO FOR como sabemos el ciclo for es una estructura repetitiva controlada. En el primer for nuestro semáforo está configurado en estado normal.

Imprimimos un mensaje que nos indica que la simulación comenzó. Encendemos el Led rojo. Con la función sleep hacemos que el Led permanezca encendido durante 3 segundos. Apagamos el Led rojo.

Repetimos el mismo procedimiento con los otros dos diodos LED.

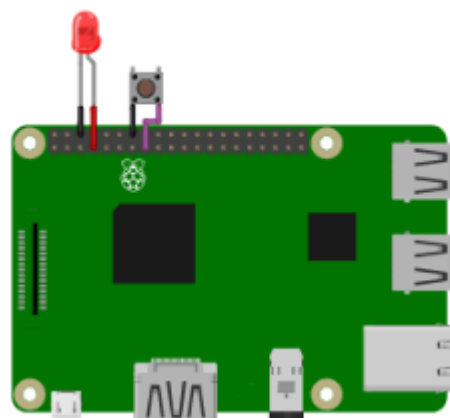
Al tener este bloque de código dentro del ciclo for programamos la raspberry para que realice esta configuración de nuestro semáforo 2 veces.

Luego de haber ejecutado el bloque de código 2 veces el primer ciclo for finaliza y se ejecuta el segundo ciclo for en el cual configuramos a nuestro semáforo en estado preventivo. Imprimimos un mensaje que nos permite identificar que el ciclo for inició. Encendemos el Led amarillo. Con la función sleep hacemos que el Led permanezca encendido durante 1 segundos. Apagamos el Led.

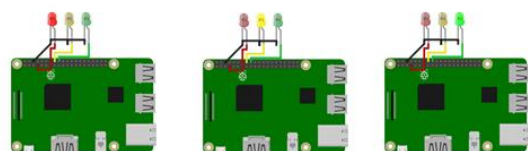
De nuevo con la función sleep hacemos que el Led permanezca apagado durante 1 segundo. Gracias al ciclo for este bloque de código se repite 5 veces. Posteriormente finaliza la ejecución del programa.

III. RESULTADOS

Circuito de entrada de datos con Pulsador y un LED.



Circuito de salida de datos con tres Diodos LED



IV. CONCLUSIONES

- La Raspberry posee un gran campo de aplicaciones con distintos niveles de dificultad, pero accesibles a múltiples usos predeterminados.
- Raspberry Pi permite la simulación de varios componentes electrónicos mediante la

programación y adaptación a circuitos establecidos.

- La visualización de entrada y salida de datos es posible y sencilla de explicar por medio de un circuito de prueba junto con diodos leds y dispositivos de ingreso de datos como lo es un pulsador.

V. REFERENCIAS

- Asociación Programo Ergo Sum. (s.f.). Programo Ergo Sum. Obtenido de <https://www.programoergosum.com/cursos-online/raspberry-pi/232-curso-de-introduccion-a-raspberry-pi/que-es-raspberry-pi>
- ComputerHoy. (23 de enero de 2014). CH. Obtenido de <https://computerhoy.com/noticias/hardware/que-es-raspberry-pi-donde-comprarla-como-usarla-8614>
- Mozilla web docs. (17 de Agosto de 2020). Mozilla web docs. Obtenido de https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics
- OpenJS Foundation. (s.f.). OpenJS Foundation. Obtenido de <https://nodejs.org/es/about/>

Autores

Frías Cajas Melanie Rashel
Guanoluisa Anoluisa Cynthia Fernanda
Gallegos Medrano Jorge Antonio
Estudiantes de pregrado
Ing. Telecomunicaciones