

REPORTE PRACTICA:
“ CONTADOR BINARIO CON GLADE Y SALIDA EN LEDS”

Omar I. Vizcaino Alarcon 24000207

Axel Arriola Fonseca 5802023

Karime García Xalteno 24100349

Microprocesadores

Puebla, Pue. A 21 de mayo de 2019

Introducción:

En esta práctica se realizó un programa en Glade capaz de utilizarse como contador binario, en este caso se tenía en el protoboard con 4 Leds, representando un código binario de 4 bits. Por otro lado en el programa Glade se tenía que introducir una entrada binaria de máximo 3 bits. De este modo la entrada seria en el Glade y la salida los Leds, el programa es capaz de duplicar la cantidad, es decir si se mete un 7 saldria un 14.

Material:

- Raspberry Pi 3 (cables)
- Computadora o algún visualizador para la Raspberry
- Protoboard
- Cables para Protoboard
- 4 Leds
- Deep Swich de 4

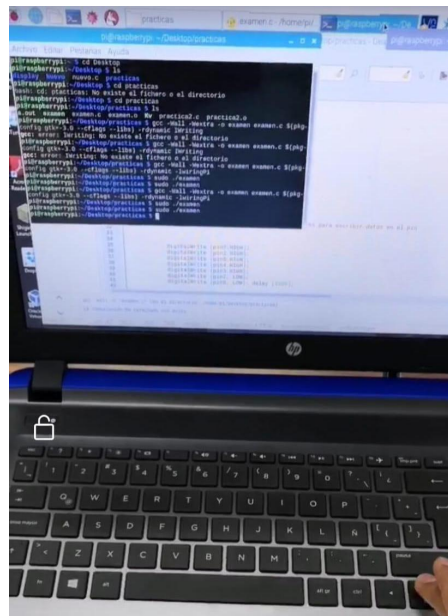
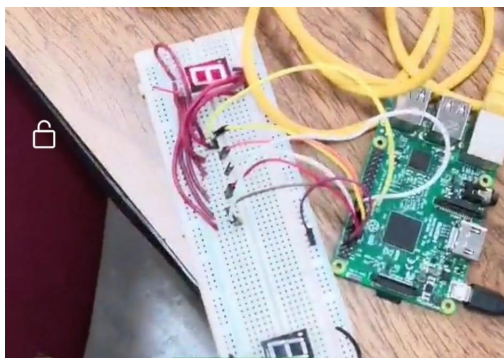
Procedimiento:

- Primero se creó la ventana en Glade con un botón de “duplicar número”, una etiqueta “imprimir número duplicado” y una entrada “introducir número a duplicar menor a 8”.
- Después se realizó el programa en Geany, lo más importante fue introducir la función para convertir el número introducido a binario.
- Posteriormente se utilizo la funcion Wiring.Pi para definir las señales de salida en la Raspberry con los GPIOs
- También se asimilaron tanto el botón, como la etiqueta y la entrada del Glade para que se pudiera establecer en el programa.
- Finalmente se se asiló la entrada del número binario de 3 bits y se asiló con la salida de 4 bits (multiplicado).

- El código e imágenes se encuentran en anexos.

Resultados:

Se obtuvieron los resultados esperados al final, aunque tuvimos inconvenientes al principio para correr el programa, no se había vinculado con el Glade y algunos datos estaban mal asimilados. Finalmente funcionó correctamente. Se anexan imágenes.



Conclusión:

Se puede definir que con esta práctica se logró entender mejor el programa Glade así como otras funciones en Geany y cómo asimilar diferentes funciones de botones, salidas y entradas, además de la implementación del código binario en el lenguaje C++.

Anexos:

```
#include <stdio.h>
```

```
#include <wiringPi.h>
```

```
int main (void)
```

```
{
```

```
    printf ("Led working\n");
```

```
    if(wiringPiSetup() == -1)
```

```
        return 1;
```

```
    pinMode(8, INPUT);
```

```
    pinMode(9, INPUT);
```

```
    pinMode(7, INPUT);
```

```
    pinMode(0, OUTPUT);
```

```
    pinMode(2, OUTPUT);
```

```
    pinMode(3, OUTPUT);
```

```
    pinMode(12, OUTPUT);
```

```
    for(;;)
```

```
    {
```

```
        if(digitalRead(8) == LOW)
```

```
        {
```

```
            if(digitalRead(9) == LOW)
```

```
            {
```

```
                if(digitalRead(7) == LOW)
```

```
                {
```

```
                    printf("Resultado: 0\n\r");
```

```
                }
```

```
        if(digitalRead(7) == HIGH)
```

```
        {
```

```
            printf("Resultado: 4\n");
```

```
        }
```

```

    }

    if(digitalRead(9) == HIGH)
    {
        if(digitalRead(7) == LOW)
        {
            printf("Resultado: 2\n\r");
        }
        if(digitalRead(7) == HIGH)
        {
            printf("Resultado: 6\n");
        }
    }

}

if(digitalRead(8) == HIGH)
{
    if(digitalRead(9) == LOW)
    {
        if(digitalRead(7) == LOW)
        {
            printf("Resultado: 1\n\r");
        }
        if(digitalRead(7) == HIGH)
        {
            printf("Resultado: 5\n");
        }
    }
}

```

```
if(digitalRead(8) == HIGH)
{
    if(digitalRead(7) == LOW)
    {
        printf("Resultado: 3\n\r");
    }
    if(digitalRead(7) == HIGH)
    {
        printf("Resultado: 7\n");
    }
}
}
}
}
```