

Práctica 1: Laboratorio de Principios de Mecatrónica

Rebeca Baños García, Leonardo García Osuna

Resumen—En esta práctica se aprendió a usar las operaciones básicas de la terminal de GNU/Linux, así como a usar el programa de Arduino para realizar funciones en el microcontrolador y por último lo realizado se respaldó en Github usando los comandos necesarios.

1. INTRODUCCIÓN

El aprender a usar la terminal de un sistema operativo es conveniente ya que todos los procesos se ejecutan de ~~manera directa, sin necesidad de pasar por un compilador~~. Para Arduino, los problemas presentados en la práctica nos ayudaron como introducción para poder conocer el lenguaje y el uso del microprocesador en cuanto a entradas y salidas. Como ingenieros, nos conviene hacer un respaldo de todos los códigos y experimentos que realizamos y Github nos permite hacer respaldos de códigos de tal manera que no haya problema al abrir el código en cualquier otra computadora.

2. MARCO TEÓRICO

Los dispositivos utilizados en esta práctica fueron:

- **Terminal de Linux:** La terminal de Linux, que es una forma de acceder al sistema sin utilizar la interfaz gráfica, es decir, realizar todo tipo de tareas en formato texto. La forma de utilizar el sistema de este modo es mediante ~~órdenes~~.
- **Programa Arduino y microcontrolador:** Arduino es un Autómata programable, por esto es más fácil realizar proyectos como recolección de datos, supervisión del entorno, envío de alarmas, accionar motores, y otros más. Los microcontroladores pueden adquirir variados y diferentes espacios de implementación, por ejemplo, los microprocesadores se han desarrollado fundamentalmente orientados al mercado de los ordenadores personales y las estaciones de trabajo, pues allí se requiere una elevada potencia de cálculo, el manejo de gran cantidad de memoria y una gran velocidad de procesamiento.
- **Github:** Github es una plataforma de desarrollo colaborativo para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de computadora. El código de los proyectos alojados en Github se almacena típicamente de forma pública, aunque utilizando una cuenta de pago, también permite hospedar repositorios privados.

3. DESARROLLO

Para iniciar la práctica se aprendió a generar directorios y archivos en diferentes ubicaciones de la computadora, utilizando la terminal del sistema.

Como segunda parte, se realizaron los programas en el IDE de Arduino. Primero, para que un LED respondiera al valor de una fotoresistencia, para lo cual decidimos que se adaptara dicho LED a la intensidad de luz de la siguiente manera:

```
if (ena) {  
    int sensorValue = analogRead(in_frst);  
    int ledValue=(sensorValue-2)*6;  
    analogWrite(LED,ledValue);  
} else {  
    analogWrite(LED,0);  
}
```

Dependiendo del switch, se enciende o apaga el LED

Después, debimos –en otro circuito– definir los pins para un display de 7 segmentos de LED, y en el código, asociar cada caracter a una combinación de encendido/apagado para mostrar adecuadamente los caracteres.

```
int lectura = Serial.read();  
if(lectura>0){  
    Serial.println(lectura);  
    if(lectura>47 && lectura<58){  
        Num_Write(lectura-48);  
    } else if(lectura>96 && lectura<103){  
        Num_Write(lectura-87);  
    } else {  
        Serial.println("Comando no soportado.");  
    }  
}
```

Se escribe un valor en la pantalla sólo si es un valor 0-f

Los valores ASCII de los caracteres 0-9 y a-f son 48-57 y 97-102, respectivamente.

Por último, nos aseguramos de tener todo en la misma carpeta para respaldarlo en Github y así hacer más fácil y rápida la modificación de nuestro código si es necesario.

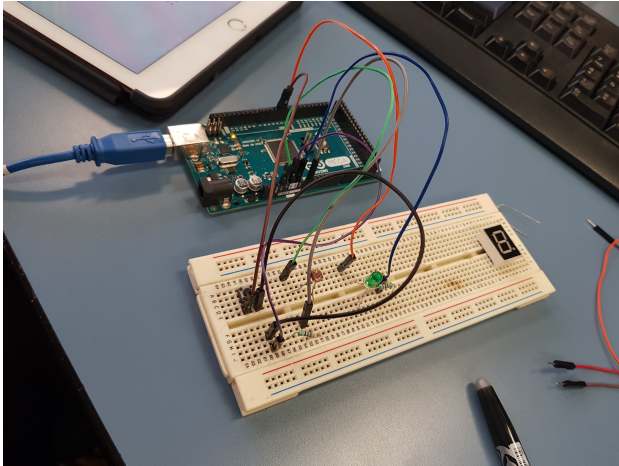


Figura 1. Problema 1.

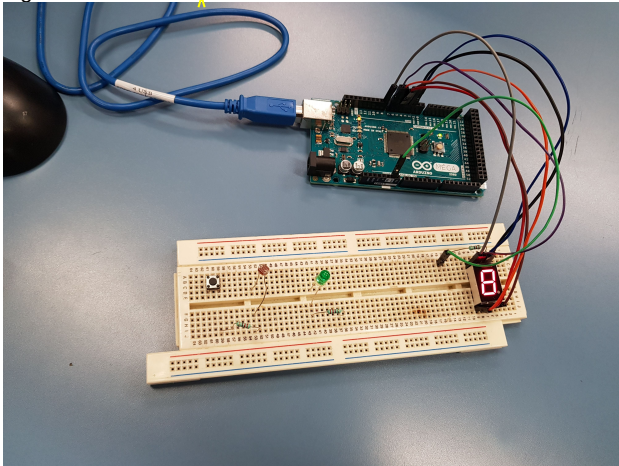


Figura 2. Problema 2.

4. RESULTADOS

Al realizar las consultas en la terminal de Linux, tuvimos algunas fallas en cambiar de directorio, ya que el diagrama que tuvimos que seguir fue algo complicado y nos equivocamos varias veces. Al programar el código necesario en Arduino, tuvimos que consultar varios ejemplos anteriores y revisar lo que el programa estaba arrojando como resultado para así leer de manera correcta lo necesario para encender el led y el display de 7 segmentos. Al realizar el problema 1, los resultados arrojados por el programa eran números, por lo tanto tuvimos que interpretarlos para ponerle la intensidad adecuada al led y que reaccionara con el sensor. En la segunda parte del problema 1, tuvimos un pequeño inconveniente al ingresar el botón que activa todo el programa, pero logramos solucionarlo codificándolo como un interruptor. El problema dos fue sencillo una vez que investigamos como se declaraban las variables necesarias para que se encendieran todos los leds del display de 7 segmentos. Utilizamos matrices para cada número y letra a representar y cada elemento de la matriz representaba el led a encender. Para guardar la carpeta en Github utilizamos los comandos git necesarios en la terminal.

5. CONCLUSIONES

- **Rebeca:** La práctica me enseñó a usar la terminal mucho más a fondo de lo que la utilizaba antes; aprendí comandos muy útiles que antes desconocía y a moverme más rápido en los directorios. En cuanto a Arduino, aprendí un lenguaje que antes desconocía totalmente y me di cuenta que es un poco más sencillo aprender un nuevo lenguaje de programación cuando conoces uno previamente.
- **Leonardo:** Con experiencia previa, pensar en las soluciones de la práctica resultó sencillo. La terminal de Linux es esencial conocerla y usarla bien, dado que para muchas plataformas de desarrollo, su principal acceso, configuración e interacción, es mediante dicha terminal.

6. ROL O PAPEL

- **Rebeca:** Se encargó de guiar los pasos de la práctica, investigar los apoyos para la realización de cada problema y escribir código.
- **Leonardo:** Se encargó de escribir lo solicitado en la terminal para llegar al directorio deseado, cablear la protoboard y escribir código.

7. FUENTES CONSULTADAS

https://www.ecured.cu/Terminal_de_GNU/Linux
<https://aprendiendoarduino.wordpress.com/category/microprocesador/> <https://es.wikipedia.org/wiki/GitHub>