

Practica 5: lectura de potenciómetro y control de salida PWM

reporte

Universidad de Guadalajara

Seminario de programación de sistemas embebidos D01-I9893

equipo

Ángel de Jesús Vázquez Zarate

Naresh Satoshi López Ojeda

Daniel Esau Gonzalez Molina

Fernanda Hermosillo Gonzáles

Resumen- En esta práctica, el sistema se implementó utilizando la tarjeta de desarrollo ESP32 para leer valores de potenciómetro y generar salida PWM proporcional.

Materiais

- ESP32 microcontroller
- Potenciómetro
- Cables de conexión
- Proto board

Marco Teorico

ESP32 Microcontrolador

El ESP32 es un microcontrolador económico y de alto rendimiento desarrollado por Espressif Systems. Se utiliza ampliamente en proyectos de electrónica e IoT (Internet de las Cosas) gracias a sus avanzadas funciones y versatilidad. Se puede programar con Arduino IDE, MicroPython o Espressif IDF y se aplica en diversos proyectos, como controles remotos, domótica, sistemas de monitorización, sensores conectados a la nube, pantallas (como en su proyecto de contador con formato de reloj), robots, drones y otros dispositivos inteligentes..



Potenciómetro

resistor variable que permite ajustar manualmente la resistencia en un circuito eléctrico. Se usa para controlar voltaje y corriente, y su aplicación más común es en controles de volumen, dimmers de luz y sensores.



Conversión ADC

Un convertidor analógico-digital (ADC) es un dispositivo que convierte señales analógicas (como voltajes o corrientes) en datos digitales que una computadora o microcontrolador puede interpretar.

PWM (Modulación por Ancho de PWM

(Pulse Width Modulation o Modulación por Ancho de Pulso) es una técnica para controlar la cantidad de energía entregada a un dispositivo mediante la variación del ciclo de trabajo de una señal digital.

Code

```
const int ledPin = 26;
const int portPin = 34;
int potValor = 0;
float angulo = 0;
int perc = 0;

void setup() {
    // put your setup code here, to
    run once:
    Serial.begin(9600);
    delay(1000);
}

void loop() {
    // put your main code here, to
    run repeatedly:
    potValor = analogRead(portPin);
    angulo =
    (analogRead(portPin)*270)/4095;
    perc =
    (analogRead(portPin)*100)/4095;
    Serial.print("angulo:
    "+String(angulo));
    Serial.print("\tValor leído: " +
    String(potValor));
    Serial.println("\tPorcentaje: " +
    String(perc));
    analogWrite(ledPin, (potValor*255)
    /4095);
    delay(100);
}
```

Important functions

Loop()

Increments the seconds counter every 10 milliseconds, When seconds reach 60, they reset to 0, and minutes increase by 1, hen minutes reach 60, they reset to 0.

conclusion

En conclusión, la práctica demostró la capacidad del ESP32 para leer valores analógicos de un potenciómetro y generar una señal PWM proporcional en su salida. Esto resalta la versatilidad del microcontrolador para aplicaciones de control y automatización, donde la modulación de ancho de pulso es esencial para ajustar la potencia entregada a dispositivos como motores o luces.

References

- Arduino. (n.d.). *ESP32 Documentation*. Retrieved from <https://www.arduino.cc>
- IEEE Xplore. (2020). *Multiplexing Techniques in Digital Displays*. Retrieved from <https://ieeexplore.ieee.org>