

C3.6 Programación Microcontrolador NodeMCU ESP32

Arduino y entrada analógica, utilizando un potenciómetro



Instrucciones

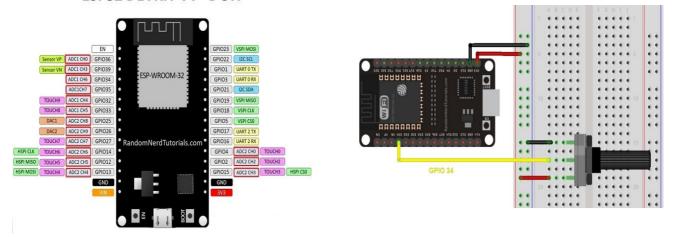
- De acuerdo con la información presentada por el asesor referente al tema, desarrollar lo que se indica dentro del apartado siguiente.
- Toda actividad o reto se deberá realizar utilizando el estilo MarkDown con extension .md y el entorno de desarrollo VSCode, debiendo ser elaborado como un documento single page, es decir si el documento cuanta con imágenes, enlaces o cualquier documento externo debe ser accedido desde etiquetas y enlaces.
- Es requisito que el archivo .md contenga una etiqueta del enlace al repositorio de su documento en Github, por ejemplo Enlace a mi GitHub
- Al concluir el reto el reto se deberá subir a github el archivo .md creado.
- Desde el archivo .md se debe exportar un archivo .pdf con la nomenclatura C3.6_NombreAlumno_Equipo.pdf, el cual deberá subirse a classroom dentro de su apartado correspondiente, para que sirva como evidencia de su entrega; siendo esta plataforma oficial aquí se recibirá la calificación de su actividad por individual.
- Considerando que el archivo .pdf, fue obtenido desde archivo .md, ambos deben ser idénticos y mostrar el mismo contenido.
- Su repositorio ademas de que debe contar con un archivo **readme**.md dentro de su directorio raíz, con la información como datos del estudiante, equipo de trabajo, materia, carrera, datos del asesor, e incluso logotipo o imágenes, debe tener un apartado de contenidos o indice, los cuales realmente son ligas o **enlaces a sus documentos .md**, evite utilizar texto para indicar enlaces internos o externo.
- Se propone una estructura tal como esta indicada abajo, sin embargo puede utilizarse cualquier otra que le apoye para organizar su repositorio.

```
readme.md
blog
| C3.1 TituloActividad.md
| C3.2_TituloActividad.md
| C3.3 TituloActividad.md
  C3.4 TituloActividad.md
  C3.5_TituloActividad.md
  C3.6 TituloActividad.md
| img
docs
| A3.1 TituloActividad.md
A3.2_TituloActividad.md
```



1. Ensamble el circuito mostrado en la figura siguiente.

ESP32 DEVKIT V1 - DOIT



2. Analice y escriba el programa que se muestra a continuación.

```
COM4
// Potentiometer is connected to GPIO 34 (Analog ADC1_CH6)
                                                                  ZOOJ
const int potPin = 34;
                                                                  4095
                                                                  4095
                                                                  4095
// variable for storing the potentiometer value
                                                                  4095
                                                                  4095
int potValue = 0;
                                                                  4095
                                                                  4095
void setup() {
                                                                  4095
                                                                  2495
 Serial.begin(115200);
                                                                  507
 delay(1000);
                                                                  0
                                                                  0
                                                                                          Voltage levels between 0V to 3.3V
                                                                  0
void loop() {
                                                                                                             4095
 // Reading potentiometer value
                                                                  Autoscrol
  potValue = analogRead(potPin);
  Serial.println(potValue);
  delay(500);
```

Fuente de consulta: Random Nerd Tutorials

3. Inserte aquí las imágenes que considere como evidencias para demostrar el resultado obtenido.

CODIGO:

```
//Entrada GPI34 del NodeMCU
const int potPin = 34;
//Variable para calcular el valor del potenciometro
int potValue = 0;
```

```
void setup() {
    //Se utilizara el monitor en serie
    Serial.begin(115200);
    //Delay de 1 segundo
    delay(1000);
}

void loop() {
    // Leer valor de potenciometro
    potValue = analogRead(potPin);
    //Imprime el valor del potenciometro
    Serial.println(potValue);
    //Delay de medio segundo
    delay(500);
}
```



00 C3.6 Arduino 1.8.12

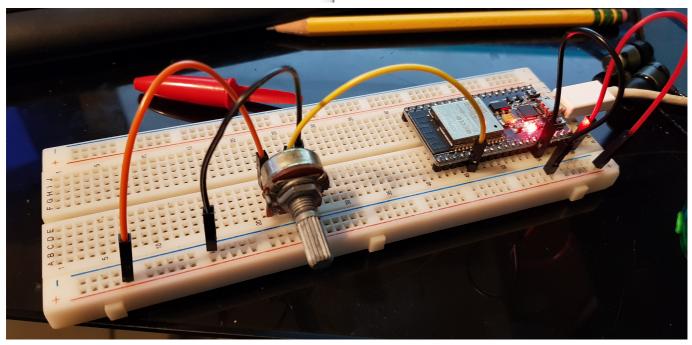
Archivo Editar Programa Herramientas Ayuda

C3.6

```
const int potPin = 34;
//Variable para calcular el valor del potenciometro
int potValue = 0;
void setup() {
Serial.begin(115200);
delay(1000);
}
void loop() {
 // Leer valor de potenciometro
 potValue = analogRead(potPin);
 Serial.println(potValue);
 delay(500);
```



17:14:49.332 -> 4095 17:14:49.842 -> 4095 17:14:50.352 -> 4095 17:14:50.830 -> 4095 17:14:51.341 -> 4095 17:14:51.850 -> 4095 17:14:52.329 -> 4095 17:14:52.841 -> 3959 17:14:53.353 -> 2791 17:14:53.828 -> 2022 17:14:54.341 -> 1232 17:14:54.853 -> 1066 17:14:55.331 -> 592 17:14:55.842 -> 332 17:14:56.353 -> 320 17:14:56.830 -> 0 17:14:57.338 -> 0





Criterios	Descripción	Puntaje
Instrucciones	Se cumple con cada uno de los puntos indicados dentro del apartado Instrucciones?	20
Desarrollo	Se respondió a cada uno de los puntos solicitados dentro del desarrollo de la actividad?	80

