

# 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.

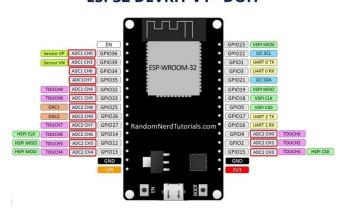
```
readme.md
 blog
  C3.1 TituloActividad.md
   C3.2 TituloActividad.md
  | C3.3_TituloActividad.md
   C3.4_TituloActividad.md
   C3.5 TituloActividad.md
   C3.6_TituloActividad.md
   img
   docs
 A3.1_TituloActividad.md
```

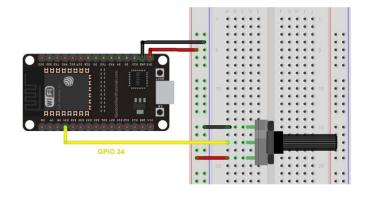


## Nesarrollo 🍳

1. Ensamble el circuito mostrado en la figura siguiente.

#### **ESP32 DEVKIT V1 - DOIT**





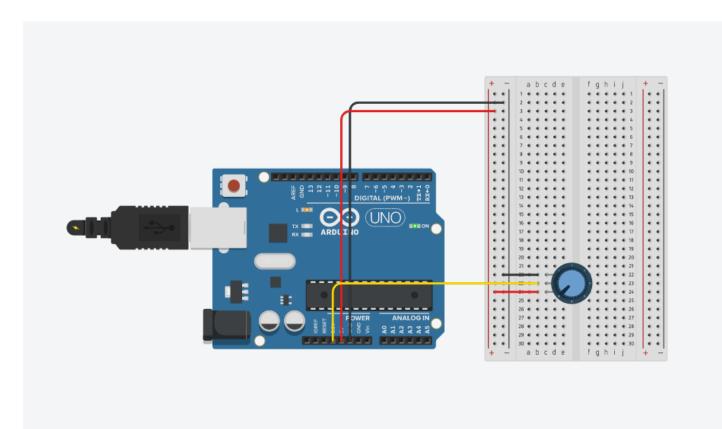
2. Analice y escriba el programa que se muestra a continuación.

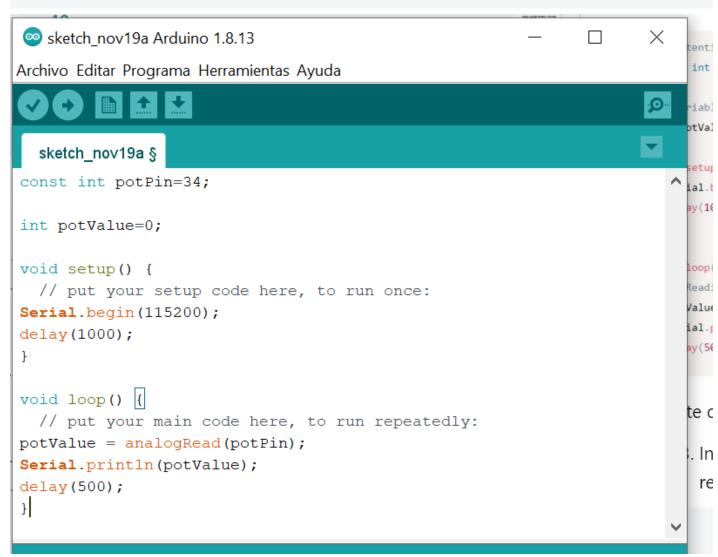
```
// Potentiometer is connected to GPIO 34 (Analog ADC1_CH6)
const int potPin = 34;
// variable for storing the potentiometer value
int potValue = 0;
void setup() {
 Serial.begin(115200);
 delay(1000);
void loop() {
  // Reading potentiometer value
  potValue = analogRead(potPin);
  Serial.println(potValue);
  delay(500);
```

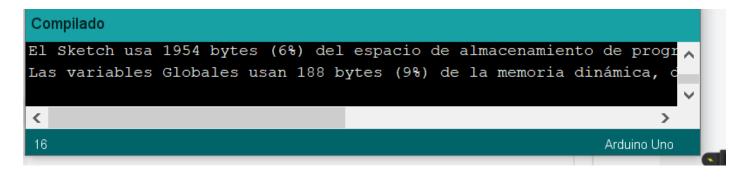
```
2000
4095
4095
4095
4095
4095
4095
4095
4095
2495
507
0
                             Voltage levels between 0V to 3.3V
0
                                                    409
                                   0
                                   0V
☑ Autoscroll
```

Fuente de consulta: Random Nerd Tutorials

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

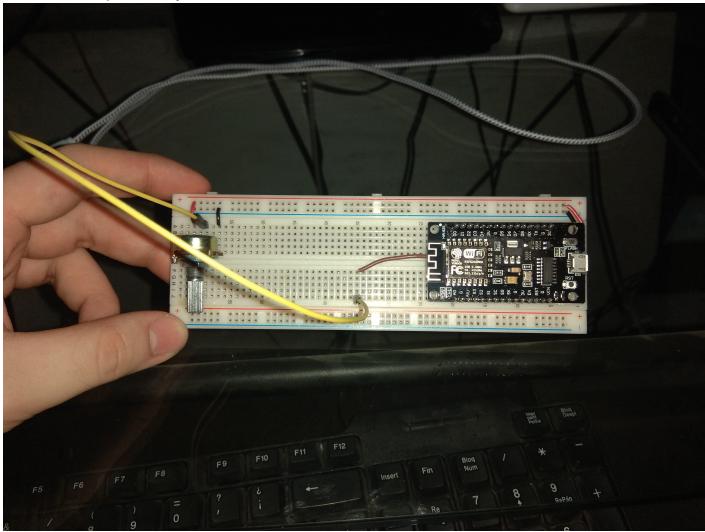






4. Evidencias de ALEJANDRO en ESP32

#### Foto de mi compañero Alejandro



Video: Explicacion realizada por mi compañero Alejandro



Criterios Descripción Puntaje

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



Repositorio en GitHub