ADC con ESP32

Un ADC es un dispositivo electrónico que sirve para convertir una señal analógica (de tensión o de corriente) en una señal digital, con el propósito de facilitar su procesamiento, así como hacer la señal resultante (digital) más inmune al ruido y otras interferencias a las que son más sensibles las señales analógicas.

La ESP32 cuenta con varios pines que funcionan como ADCs que podemos observar en el pinout de la tarjeta con la que contemos en la placa ESP32 DEVKIT V1 contamos con 18 canales de convertidores analogicos-digitales.

GPIO23 VSPI MOSI EN Sensor VP ADC1 CH0 GPIO36 GPIO22 I2C SCL Sensor VN ADC1 CH3 GPIO39 GPIO1 UART 0 TX ESP-WROOM-32 ADC1 CH6 GPIO34 GPIO3 UART 0 RX ADC1CH7 GPIO35 GPIO21 I2C SDA TOUCH9 ADC1 CH4 GPIO32 GPIO19 VSPI MISO TOUCH8 ADC1 CH5 GPIO33 GPIO18 VSPI CLK DAC1 ADC2 CH8 GPIO25 DAC2 ADC2 CH9 GPIO26 GPIO17 UART 2 TX TOUCH7 ADC2 CH7 GPIO27 GPIO16 UART 2 RX GPIO4 ADC2 CH0 TOUCH0 HSPI CLK TOUCH6 ADC2 CH6 GPIO14 HSPI MISO TOUCHS ADC2 CH5 GPIO12 GPIO2 ADC2 CH2 TOUCH2 HSPI MOSI TOUCH4 ADC2 CH4 GPIO13 GPIO15 ADC2 CH3 TOUCH3 HSPI CS0 GND GND ■ E

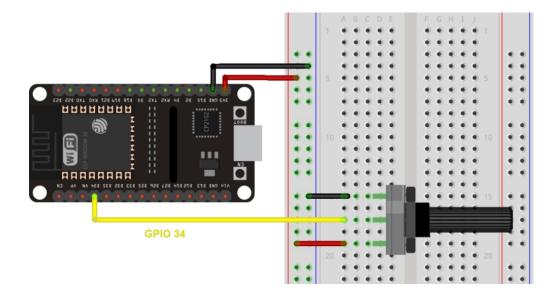
ESP32 DEVKIT V1 - DOIT

En esta práctica utilizaremos un ADC para convertir las señales Analógicas a digitales conectado un potenciómetro a un canal que funcione como un adc para su lectura en el monitor serial.

Materiales necesarios:

- ESP8266
- Cable USB
- Computadora con el IDE de Arduino instalado
- Potenciómetro
- Jumpers

Esquemático que se utilizara para la práctica:



```
// Declarar el pin 34 que es el GPIO 34 que funciona como adc (Analog
ADC1_CH6)
const int potPin = 34;

// Variable para almacenar el valor del potenciómetro
int potValue = 0;

void setup() {
   Serial.begin(115200);//Se inicia el puerto serial en 115200 baulio
   delay(1000);
}

void loop() {
   potValue = analogRead(potPin); // Leer el valor del potenciómetro
   Serial.println(potValue);// Imprimir el valor del potenciómetro en el
puerto serial
   delay(500);
}
```

Puedes encontrar el código completo y los pdf en la sig pagina de GitHub:

https://github.com/GabrielCorUs/PracticasESP32/tree/main/ESP32 Practica1