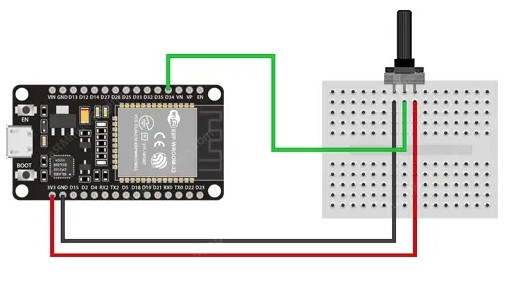
9.ADC Calibration with ESP 32(Practical)

Circuit Diagram:



9.(A).ADC Calibration with ESP 32(Practical)

Source code:

#include "esp\_adc\_cal.h"

#define AN\_Pot1    35

int AN\_Pot1\_Result = 0;

float Voltage = 0.0;

void setup()

{

  Serial.begin(115200);

}

void loop()

{

  AN\_Pot1\_Result = analogRead(AN\_Pot1);

  Voltage = readADC\_Cal(AN\_Pot1\_Result);

  Serial.println(Voltage/1000.0); // Print Voltage (in V)

  //Serial.println(Voltage);      // Print Voltage (in mV)

  delay(1000);

}

uint32\_t readADC\_Cal(int ADC\_Raw)

{

  esp\_adc\_cal\_characteristics\_t adc\_chars;

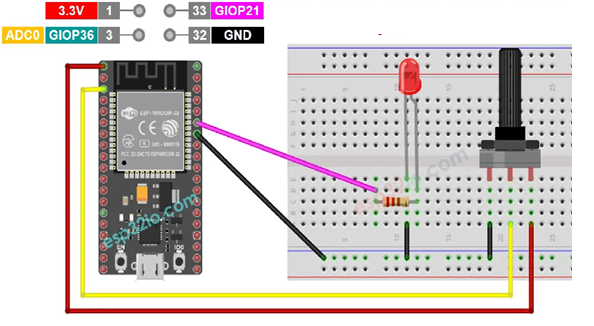
  esp\_adc\_cal\_characterize(ADC\_UNIT\_1, ADC\_ATTEN\_DB\_11, ADC\_WIDTH\_BIT\_12, 1100, &adc\_chars);

  return(esp\_adc\_cal\_raw\_to\_voltage(ADC\_Raw, &adc\_chars));

}

9.ADC read using LED dimmer using ESP 32(Skilling)

Circuit Diagram:



Source code:

// Define constants

const int potPin = 35;   // Analog input pin for potentiometer

const int ledPin = 21;    // PWM output pin for LED

void setup() {

  // Initialize serial communication

  Serial.begin(115200);

  // Configure LED pin as an output

  pinMode(ledPin, OUTPUT);

}

void loop() {

  // Read the analog value from the potentiometer

  int potValue = analogRead(potPin);

  // Map the potentiometer value (0-4095) to the LED brightness (0-255)

  int brightness = map(potValue, 0, 4095, 0, 255);

  // Print the potentiometer value and set the LED brightness

  Serial.print("Potentiometer Value: ");

  Serial.print(potValue);

  Serial.print("   Brightness: ");

  Serial.println(brightness);

  // Set the LED brightness using PWM

  analogWrite(ledPin, brightness);

  // Add a delay to make the changes noticeable

  delay(5000);

}