Hands on session 3: Twilight Switch

Session Introduction: This hands-on session would introduce you to the concept of analog Inputs and working principle of potentiometer. In this session we will make a twilight switch using photoresistor and demonstrate how increase in resistance effect the input values.

Session Objectives:

- **Reading and mapping analog values.**
- **Understanding Principle behind Photoresistor.**
- **Understanding principle behind potentiometer.**
- **To make a twilight switch.**

Analog Input

An analog signal is one that can take on any number of values, unlike a digital signal which has only two values: HIGH and LOW. To measure the value of analog signals, the Arduino, esp32 or other dev modules has a built-in analog-to-digital converter (ADC). The ADC turns the analog voltage into a digital value. The function that you use to obtain the value of an analog signal is analogRead(pin).

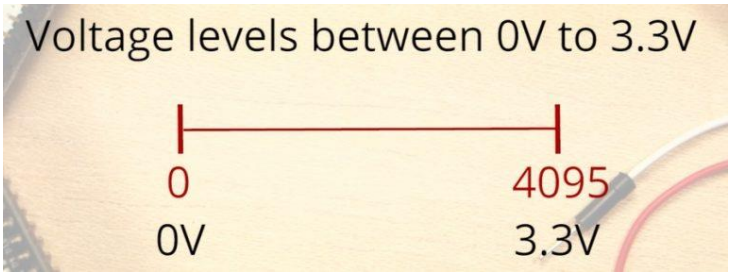
The Analog to digital converter returns a range of values from 0 to 1023 or 0 to 4095 or some other range depending upon the resolution of the analog to digital converter, relative to the reference value.

The default reference voltage is 3.3 V (for 3.3 V dev boards) or 5 V (for 5 V dev boards)

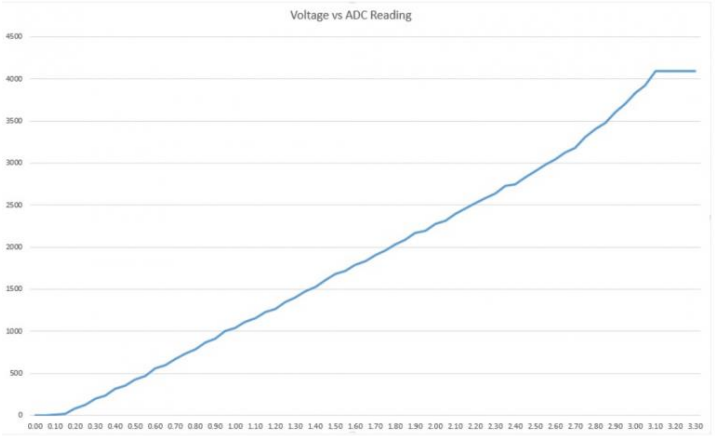
- A 10-bit analog to digital converter gives values in the range of 2^{10}->1024 values
- A 12-bit analog to digital converter gives values in the range of 2^{12}->4096 values

Reading analog values

Reading an analog value with the ESP32 means you can measure varying voltage levels between 0 V and 3.3 V.
The voltage measured is then assigned to a value between 0 and 4095, in which 0 V corresponds to 0, and 3.3 V corresponds to 4095. Any voltage between 0 V and 3.3 V will be given the corresponding value in between.



Ideally, you would expect a linear behaviour when using the ESP32 ADC pins. However, that's not true.



This behaviour means that your ESP32 is not able to distinguish 3.3 V from 3.2 V. You'll get the same value for both voltages: 4095.

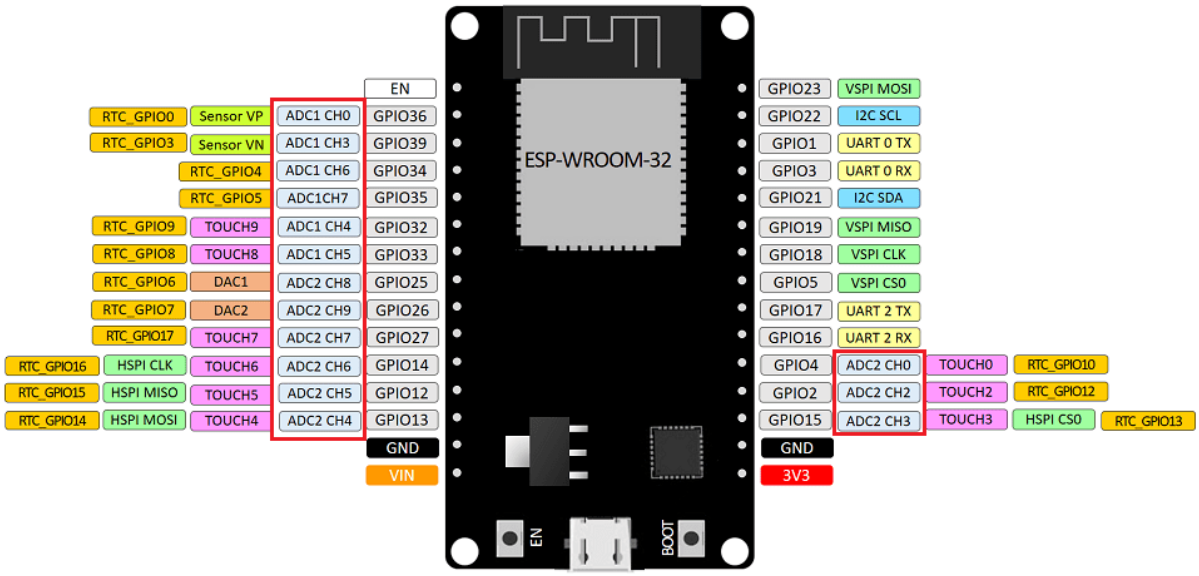
The same happens for very low voltage values: for 0 V and 0.1 V you'll get the same value: 0. You need to keep this in mind when using the ESP32 ADC pins.

analogRead() Function

Reading an analog input with the ESP32 using the Arduino IDE is as simple as using the **analogRead()** function. It accepts as argument, the **GPIO** you want to read:

- analogRead(GPIO);

The ESP32 supports measurements in 18 different channels. Only 15 are available in the espwroom32 board (version with 30 GPIOs).
The highlighted ones below can be used for ADC conversion in our esp32 dev module.



These analog input pins have 12-bit resolution. This means that when you read an analog input, its range may vary from 0 to 4095.

ADC2 pins cannot be used when Wi-Fi is used. So, if you're using Wi-Fi and you're having trouble getting the value from an ADC2 GPIO, you may consider using an ADC1 GPIO instead, that should solve your problem.

Mapping Values

The analog values we get can be remapped to a different range using inbuilt function **map().**

map() function

Description
Re-maps a number from one range to another. That is, a value of **fromLow** would get mapped to **toLow**, a value of **fromHigh** to **toHigh**, values in-between to values in-between, etc.
Does not constrain values to within the range, because out-of-range values are sometimes intended and useful. The constrain() function may be used either before or after this function, if limits to the ranges are desired.
Note that the "lower bounds" of either range may be larger or smaller than the "upper bounds" so the map() function may be used to reverse a range of numbers, for example
y = map(x, 1, 50, 50, 1);
The function also handles negative numbers well, so that this example
y = map(x, 1, 50, 50, -100);
is also valid and works well.
The map() function uses integer math so will not generate fractions, when the math might indicate that it should do so. Fractional remainders are truncated, and are not rounded or averaged.

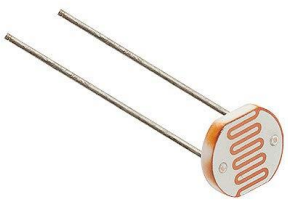
Syntax

map(value, fromLow, fromHigh, toLow, toHigh);

Parameters
value: the number to map.
fromLow: the lower bound of the value's current range.
fromHigh: the upper bound of the value's current range.
toLow: the lower bound of the value's target range.
toHigh: the upper bound of the value's target range.

Returns
The mapped value.

Photoresistor
A photoresistor (also known as a light-dependent resistor, LDR, or photo-conductive cell) is a passive component that decreases resistance with respect to receiving luminosity (light) on the component's sensitive surface.

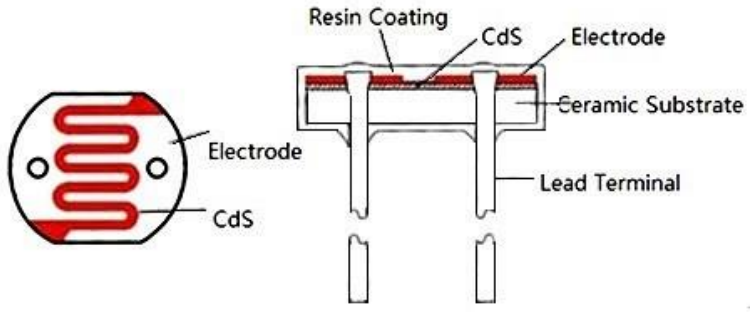


Photoresistor connections:

- **VCC -> 5V**
- **GND -> Variable Resistance 10k**
- **AI1 -> Sensor VP / GPIO 36**

Working Principle

The working principle of the photoresistor is based on the internal photoelectric effect. Photosensitive resistors are formed by mounting electrode leads at both ends of the semiconductor photosensitive material and encapsulating them in a tube case with a transparent window. In order to increase the sensitivity, the two electrodes are often made into a comb shape. The materials used to make photoresistors are mainly semiconductors such as metal sulphides, selenides, and tellurides. Coating, spraying, sintering, and other methods are used to make a very thin photoresistor and a comb-shaped ohmic electrode on an insulating substrate. The leads are connected and sealed in a sealed housing with a light-transmitting mirror to prevent its sensitivity from being affected by moisture. After the incident light disappears, the electron-hole pairs generated by the photon excitation will recombine, and the resistance of the photoresistor will return to its original value. When a voltage is applied to the metal electrodes at both ends of the photoresistor, a current pass through it. When the photoresistor is irradiated by the light with a certain wavelength, the current will increase with the light intensity, thereby achieving photoelectric conversion. The photoresistor has no polarity and is purely a resistive device. It can be used with both DC voltage and AC voltage. The conductivity of a semiconductor depends on the number of carriers in the semiconductor's conduction band.



More about Photoresistors: https://www.utmel.com/blog/categories/resistor/photoresistor-basics-types-principles-and-applications

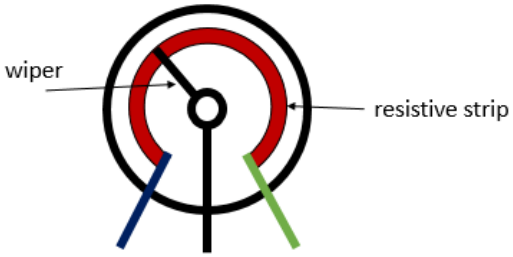
- in the dark and at low light levels, the resistance of an LDR is high and little current can flow through it
- in bright light, the resistance of an LDR is low and more current can flow through it

Potentiometer

A potentiometer is a three-terminal resistor with a sliding or rotating contact that forms an adjustable voltage divider. If only two terminals are used, one end and the wiper, it acts as a variable resistor or rheostat.

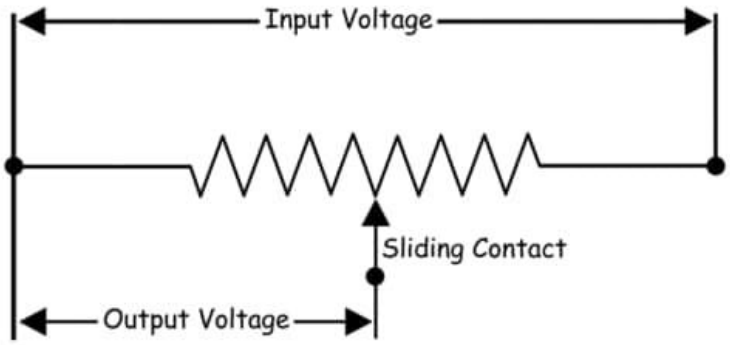
Potentiometer connections:

A potentiometer has 3 pins. Two terminals (the blue and green) are connected to a resistive element and the third terminal (the black one) is connected to an adjustable wiper.



Working principle

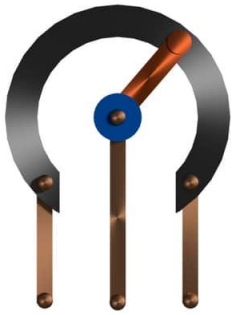
A potentiometer is a passive electronic component. Potentiometers work by varying the position of a sliding contact across a uniform resistance. In a potentiometer, the entire input voltage is applied across the whole length of the resistor, and the output voltage is the voltage drop between the fixed and sliding contact as shown below.



A potentiometer has the two terminals of the input source fixed to the end of the resistor. To adjust the output voltage the sliding contact gets moved along the resistor on the output side.

There are mainly two types of potentiometers rotatory and linear we will be using the rotatory one here.

The rotary type potentiometers are used mainly for obtaining adjustable supply voltage to a part of electronic circuits and electrical circuits. The volume controller of a radio transistor is a popular example of a rotary potentiometer where the rotary knob of the potentiometer controls the supply to the amplifier.



This type of potentiometer has two terminal contacts between which a uniform resistance is placed in a semi-circular pattern. The device also has a middle terminal which is connected to the resistance through a sliding contact attached with a rotary knob. By rotating the knob one can move the sliding contact on the semi-circular resistance. The voltage is taken between a resistance end contact and the sliding contact. The potentiometer is also named as the POT in short. POT is also used in substation battery chargers to adjust the charging voltage of a battery. There are many more uses of rotary type potentiometer where smooth voltage control is required.

Twilight Switch

A **twilight switch** is an electronic component that allows the automatic activation of a lighting circuit when natural light drops in a given environment. Among a large number of uses, the most common is to enable automatic lighting of streets, roads, highways, roads, gardens, courtyards, etc., when sunlight drops below a certain level.

we have made a voltage divider circuit using LDR and Potentiometer. The voltage divider output is feed to the analog pin of the esp32. The analog Pin senses the voltage and gives some analog value to esp32. The analog value changes according to the resistance of LDR. So, as the light falls on the LDR the resistance of it gets decreased and hence the voltage value increase.

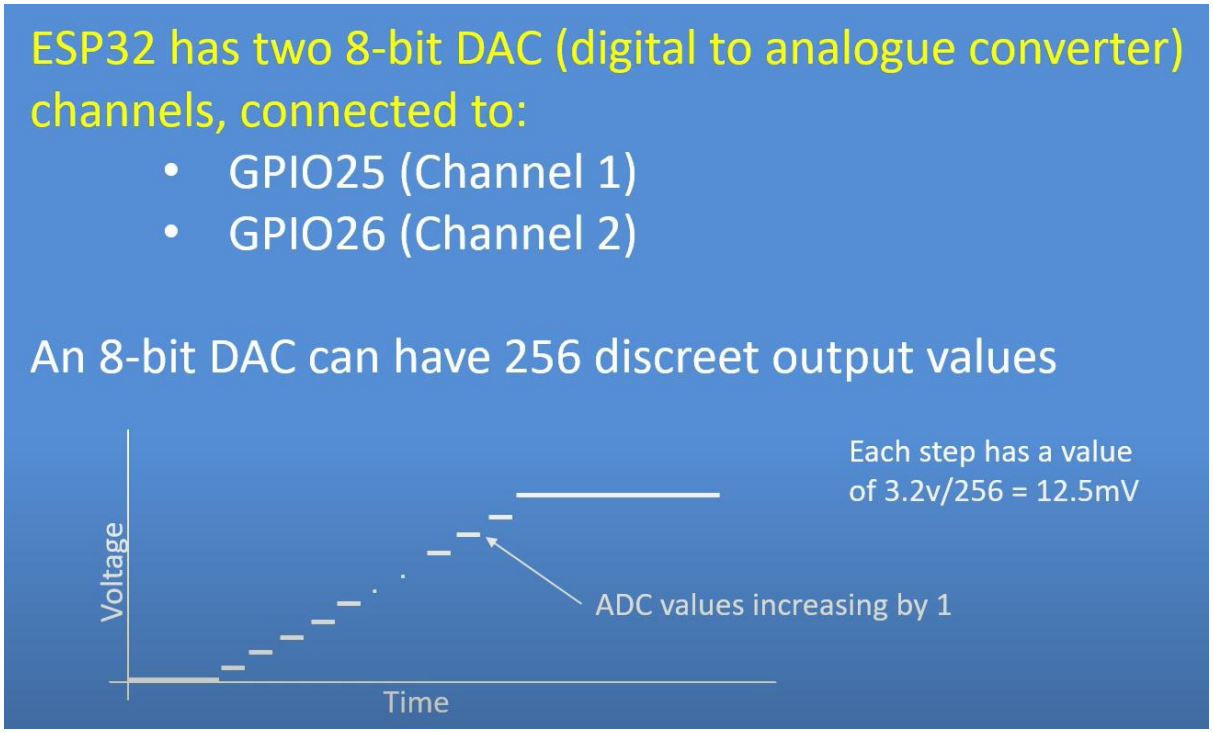
Intensity of light ↓ - Resistance↑ - Voltage at analog pin↓ - **Light turns ON**

As per our code, if the analog value falls below for example 700 we consider it as dark and the light turns ON. If the value comes above 700, we consider it as bright and the light turns OFF.

Digital to Analog Converter

Esp-32 features two digital to Analog converters:

DAC1 -> GPIO 25
DAC2 -> GPIO 26



To use DAC in Esp-32 we use function dacWrite:

dacWrite(DAC_channel,Value); // value ranges from 0-255