Linear Variable Resistance using Raspberry Pi

Introduction:

A Linear Variable Resistance (LVR) is a type of resistor whose resistance changes in a linear manner based on an external factor, such as pressure, temperature, or light. These resistances are used in applications like sensors and controllers to measure environmental changes.

In many cases, Raspberry Pi lacks an inbuilt Analog-to-Digital Converter (ADC), making it difficult to directly measure variable resistances. To overcome this, we use an ADC1115, which is a high-precision 16-bit ADC module. This module allows us to convert analog signals from the LVR into digital values that the Raspberry Pi can process.

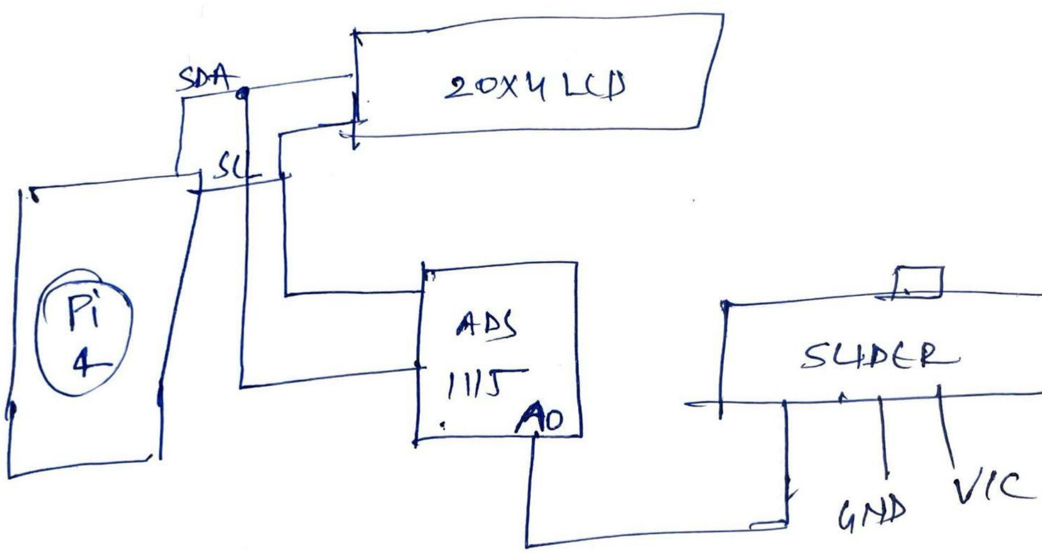
The ADC1115 operates on an I2C interface, making it easy to connect multiple sensors. By using this setup, we can measure resistance changes and interpret them for various applications, such as pressure sensing, light detection, and more. This manual provides a step-by-step guide to setting up and understanding the working of an LVR with ADC1115 and Raspberry Pi.

Components Required:

- Raspberry Pi (any model with I2C support)
- ADC1115 Module
- Linear Variable Resistor (e.g., LDR, Thermistor, or Force Sensor)
- Connecting Wires
- Breadboard
- Power Supply (5V for Raspberry Pi, 3.3V/5V for ADC1115)

Circuit Connection:

Component	**Pin Connection**
Raspberry Pi	GPIO (I2C SDA, SCL)
ADC1115 Module	VCC to 3.3V/5V
	GND to GND
	SDA to Raspberry Pi SDA
	SCL to Raspberry Pi SCL
LVR Sensor	One terminal to ADC input (A0)
	Other terminal to VCC/GND via pull-up/pull-down resistor



Applications:

- Temperature Sensors – Measuring temperature changes using thermistors.
- Light Sensors – Detecting light intensity with LDRs for automatic lighting systems.
- Force Sensors – Measuring pressure or force in industrial and consumer applications.
- Environmental Monitoring – Recording real-time environmental data for smart systems.

Learnings:

- Understanding the concept of Linear Variable Resistance and its real-world applications.
- How to interface an ADC1115 module with Raspberry Pi using the I2C protocol.
- The process of converting analog signals to digital values for further processing.
- The importance of using external ADCs with Raspberry Pi for analog sensor applications.

Conclusion:

Using an ADC1115 with Raspberry Pi enables us to measure variable resistances accurately, allowing for applications in sensors and automation. The I2C interface makes it simple to communicate with the ADC module, expanding the capabilities of Raspberry Pi in data acquisition and real-world monitoring. This setup is useful for applications like temperature monitoring, light detection, and force sensing, providing an efficient way to read analog signals digitally.

Program:

```
import time
import board
import busio
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn
from RPLCD.i2c import CharLCD

# Initialize I2C and ADS1115
i2c = busio.I2C(board.SCL, board.SDA)
ads = ADS.ADS1115(i2c)

# Initialize LCD
lcd = CharLCD('PCF8574', 0x27)

# ADC maximum value (16-bit ADS1115)
ADC_MAX = 32767

while True:
   # Read analog value from channel 0
   value = AnalogIn(ads, ADS.P0).value

   # Convert to percentage
   percentage = int((value / ADC_MAX) * 100)

   # Generate slider (10 blocks)
   slider_blocks = int((percentage / 10))
   slider = "[" + ("#" * slider_blocks).ljust(10) + "]"

   # Display on LCD
   lcd.clear()
```

```
lcd.cursor_pos = (0, 0)
lcd.write_string("ADS1115 Analog Value\n")
lcd.cursor_pos = (2, 0)
lcd.write_string(f"Slider: {percentage}%\n")
lcd.cursor_pos = (3, 0)
lcd.write_string(slider)

time.sleep(1)
```