Lesson 19 LED Bar Graph

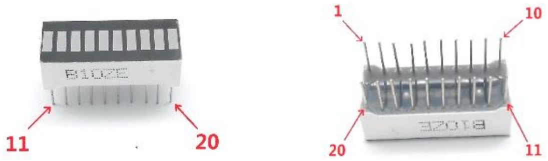
Overview

In this lesson, we will learn how to control an LED bar graph by programming the Raspberry Pi.

Requirement

- 1* Raspberry Pi
- 1* ADC0832
- 1* LED bar graph
- 10* 220Ω Resistor
- 1* 10KΩ Potentiometer
- 1* Breadboard
- Several Jumper wires

Principle



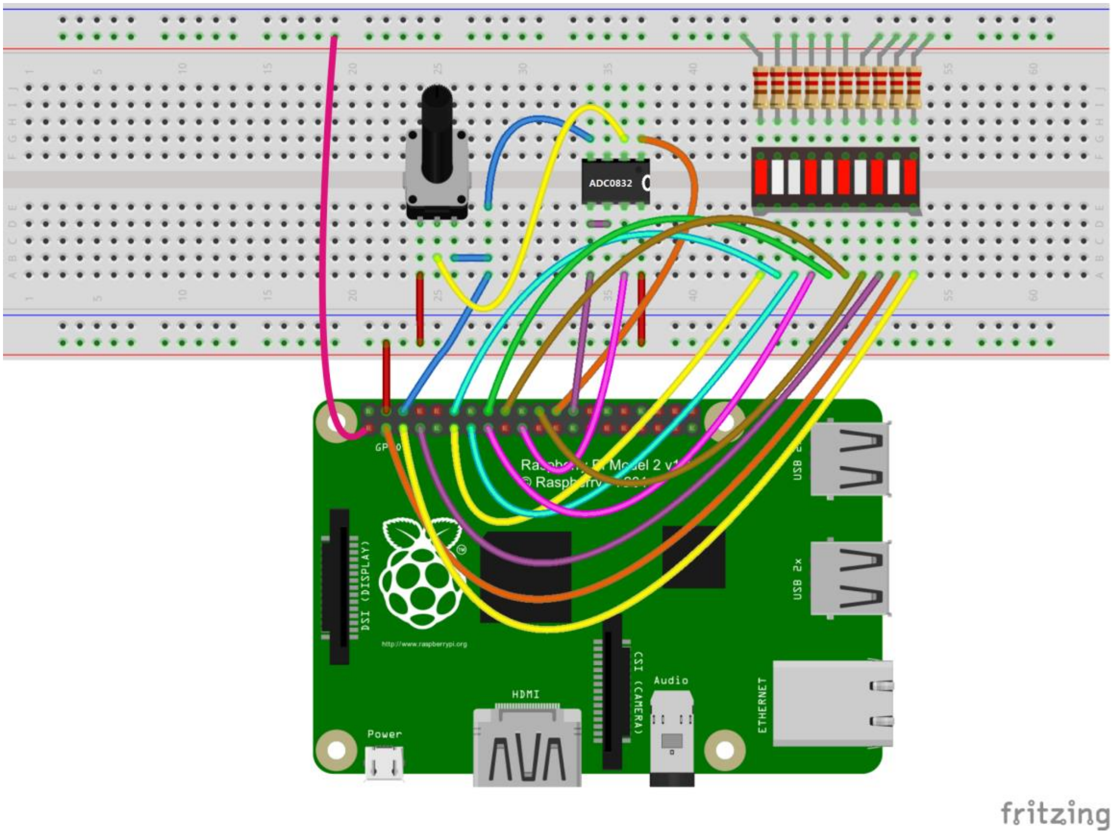
The bar graph - a series of LEDs in a line, such as you see on an audio display - is a common hardware display for analog sensors. It's made up of a series of LEDs in a row, an analog input like a potentiometer, and a little code in between. You can buy multi-LED bar graph displays fairly cheaply. This tutorial demonstrates how to control a series of LEDs in a row, but can be applied to any series of digital outputs.

This tutorial borrows from the For Loop and Arrays tutorial as well as the Analog Input tutorial.

The sketch works like this: first you read the input. You map the input value to the output range, in this case ten LEDs. Then you set up a *for* loop to iterate over the outputs. If the output's number in the series is lower than the mapped input range, you turn it on. If not, you turn it off.

Procedures

1. Build the circuit



C user:

2.1 Edit and save the code with vim or nano.

(Code path: /home/Adeept_Ultimate_Starter_Kit_C_Code_for_RPi/19_ledBar/ledBar.c)

2.2 Compile the program

```
$ gcc ledBar.c -o ledBar -lwiringPi
```

2.3 Run the program

```
$ sudo ./ledBar
```

Python user:

2.1 Edit and save the code with vim or nano.

(Code path: /home/Adeept_Ultimate_Starter_Kit_Python_Code_for_RPi/19_ledBar/ledBar.py)

2.2 Run the program

```
$ sudo python ledBar.py
```

Now, when you turn the knob of the potentiometer, you will see that the number of LED in the LED bar graph will be changed.