ARDUINO™

LCD Control

Introduction

The goal of this project is to understand how potentiometers and LCDs work and to understand analogue inputs to microcontrollers.

Potentiometers, or "pots" for short, are electrical components that are used to adjust the resistance in a circuit. They consist of a resistive element and a sliding contact, which can be moved along the resistive element to change the resistance.

Pots are commonly used in electronic circuits for a variety of purposes, such as controlling the volume of an amplifier, adjust the brightness of a display, or setting the frequency of an oscillator. They come in various shapes and sizes, with different resistance values and power ratings, and can be either single-turn or multi-turn. Pots are a versatile and essential component in many electronic designs.

The LCD display 16x2c is a type of alphanumeric liquid crystal display that can display 16 characters per row and has 2 rows of characters.
This type of display is commonly used in various electronic devices, such as calculators, digital clocks and small appliances. The display is easy to read and provides a simple, yet effective way to convey information to users.

Common Instructions

- Your project should be written in C or C++
- You can use the Arduino libraries
- Your code should have the expected output.
- Submit your work to a git repository by the specified deadline

The electronics components needed to test the output of your code will be given to your team and they should be returned at the end of the program in the same condition you received them.

Mandatory Part

The LCD display should display the potentiometer value which corresponds to the brightness of the LED.

The LCD display should be updated in real-time as the potentiometer value changes.
You will be given the following components:

- A microcontroller (Arduino UNO)
- LCD Display
- Resistors 180 ohm, 10k ohm LED
- Potentiometer 10k Ohms
- A breadboard
- Jumper wires

Bonus Part

Now that you can display the current potentiometer value and the brightness of the LED on the LCD display, you should be able to control the brightness of the LED from your computer via the Arduino terminal instead of the potentiometer.

.Submission and Peer-Evaluation

Prepare a demonstration of the projects features, design, and performance.

Turn in your assignment in an open git repository of your choice or to the git repository that was assigned to you. Only the code inside your repository will be evaluated during the defense. Be prepared to explain your code concisely and the methods you used.

All members on the group must be present at the evaluation.

You will receive a badge on intra after successful evaluation.

Ensure the project is reliable and safe to use.

Please use the electronics components safely:

- Power supply: Always use the correct voltage and current rated power supply for your Arduino board to avoid damaging the board or causing a fire.
- ESD protection: Electrostatic discharge (ESD) can damage electronic components, including the Arduino board. Avoid touching the pins and connectors of the board with your bare hands. Use an anti-static wrist strap or touch a grounded metal object before handling the board.
- Overheating: Do not overload the board with too much current, as this can cause the board to overheat and possibly catch fire. Use external power sources and avoid powering high current devices directly from the board.
- Short circuits: Avoid creating short circuits on the board by ensuring that there are no loose wires or connections touching each other. Use proper insulation and avoid exposing the board to moisture.
- Software safety: Always use reliable and tested code when programming the Arduino board. Do not run code that could potentially cause harm to people or property. Always use the appropriate safety precautions when testing and using your projects.