

Project Description: Building a Locker Door

Components Used:

1. Arduino Uno: The microcontroller board that acts as the brain of the project and controls all the components.
2. Battery: Provides power to the Arduino board and the connected components.
3. Breadboard: Used for connecting and organizing electrical components.
4. I2C LED: Communicates using the I2C backpack and provides visual information to the user.
5. Jumper wires: Used to establish connections between components on the breadboard.
6. Keypad: A matrix of buttons arranged in rows and columns, which is used for inputting passwords or commands.
7. Servo motor: A motor that can rotate to specific angles. It is used to control the opening and closing of the locker door.

Libraries Used:

1. Keypad library:
The library is used to interact with the keypad. It defines the layout of rows and columns and provides functions to read the keys pressed by the user.
(Keypad.h)
2. Wire library:
This library is used to communicate with the I2C LED. It initializes the I2C backpack communication and provides functions to send and receive data.
(Wire.h)
3. LiquidCrystal I2C library:
This library is used to simplify the control of the I2C LED display and helps to initialize the LED and provides functions to write text on the display.
(LiquidCrystal_I2C.h)
4. Servo library:
This library enables control of the servo motor. It provides functions to set the angle of the motor shaft, allowing the opening and closing of the door.
(Servo.h)
5. EEPROM library:
This library is used for reading from and writing to the EEPROM memory. I used it to store and retrieve the password data.
(EEPROM.h)

The product operates in multiple stages:

1. Initialization:

Adding necessary libraries

Declaring objects and variables (such as Servo and LiquidCrystal_I2C)

Initialising the LED display and servo motor.

2. Setup:

Initialise the LED display and turn on the backlight.

Display a message on the LED if it is the first run.

Attach the servo motor to a specific pin and set it to its closed position.

3. Main Loop:

The loop function continuously executes, handling the main functionality of the product.

If the door is closed (door = 0), the user input from the keypad is checked.

If the '#' key is pressed, the door locks, and the LED displays a "door closed" message.

If the door is open (door = 1), the 'Open' function handles the password entry and door opening.

The user's password input is stored in the Data array, and the keys are displayed on the LED.

When the maximum password length is reached, the entered password is compared with the master password.

If the passwords match, the door is opened by calling the 'ServoOpen' function, and the "door open" message is displayed.

If the passwords don't match, the "wrong password" message is displayed, and the door remains closed.

The data array is then cleared for the following password entry.

4. Helper Functions:

clearData: Clears the data array by setting each element to 0.

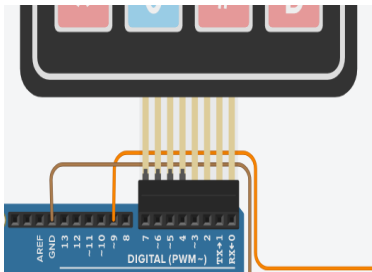
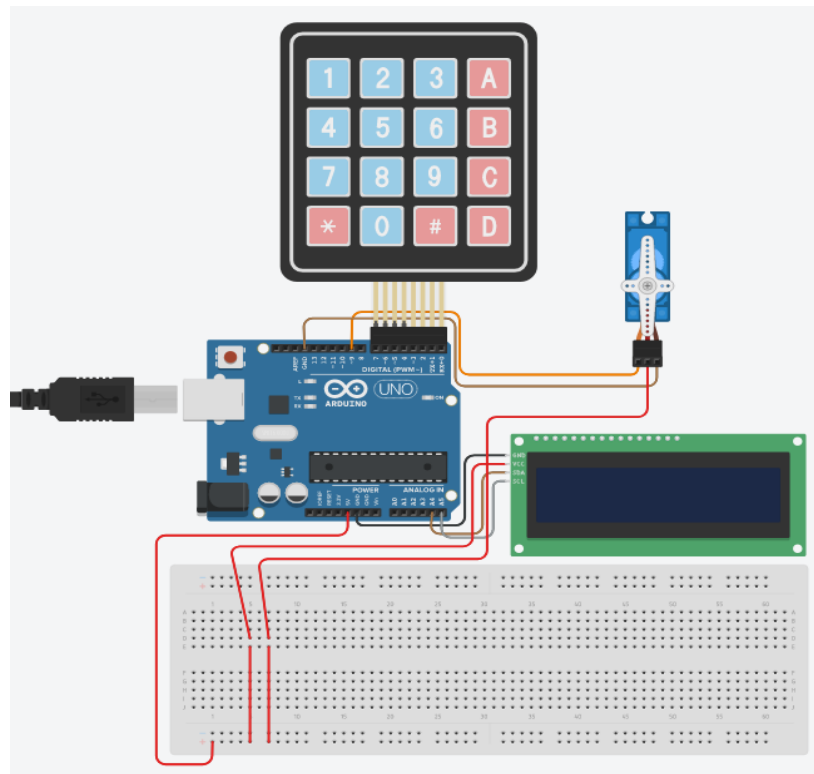
ServoOpen: Opens the door by changing the angle of the servo motor from 90 deg to 0 deg.

ServoClose: Closes the door by changing the angle of the servo motor from 0 deg to 90 deg.

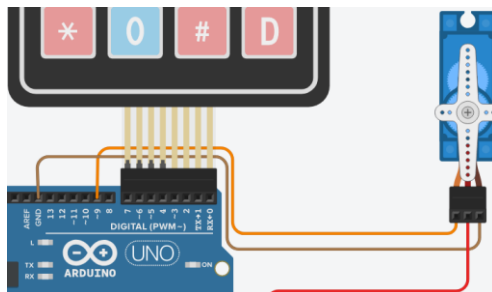
Open: Handles the process of password entry and door opening.

i2CAddrTest: Checks for a communication connection with the I2C LED using the I2C address.

Schematic:

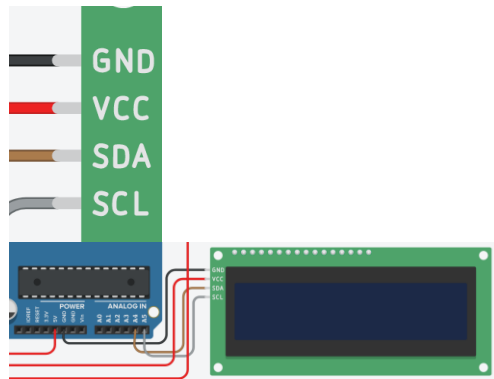


The keypad is linked to the Arduino board through a matrix layout. It is configured in such a way that each key represents a distinct combination of rows and columns. The keypad is wired to the Arduino digital pins.



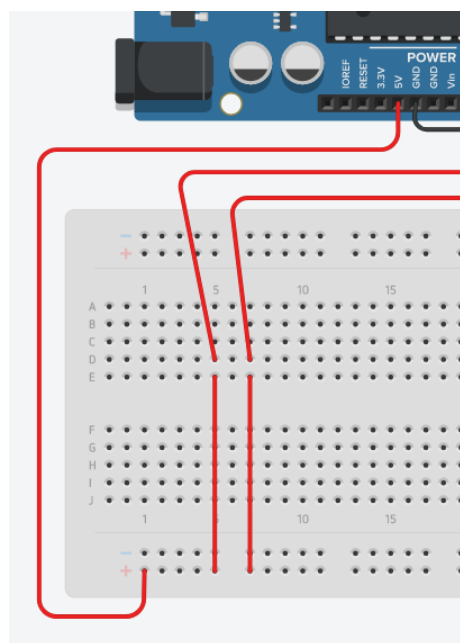
The servo motor is connected to the Arduino board using three wires: power, ground, and control.

1. Power: The power wire (red) of the servo motor should be connected to a 5V pin on the Arduino board. This wire supplies power to the servo motor.
2. Ground: The ground wire (usually black or brown) of the servo motor should be connected to a GND pin on the Arduino board.
3. Control: The control wire (usually yellow or orange) of the servo motor should be connected to a digital pin on the Arduino board. This wire carries the control signal that determines the position of the servo motor.



The I2C LED display is connected to the Arduino board using the I2C backpack.

1. GND (Ground): Connect the GND pin of the I2C backpack to the GND pin of the Arduino board. This establishes a common ground reference.
2. VCC (Power): Connect the VCC pin of the I2C backpack to the 5V pin of the Arduino board. This provides power to the LED display.
3. SDA (Serial Data): Connect the SDA pin of the I2C backpack to the A4 analog pin on the Arduino board. This is the data line used for communication between the Arduino and the LED display.
4. SCL (Serial Clock): Connect the SCL pin of the I2C backpack to the A5 analog pin on the Arduino board. This is the connection used for synchronizing data transfer between the Arduino and the LED display.



To power the I2C LED display and the servo motor, the breadboard is used as a central point for distributing power. The breadboard acts as a platform to connect and organize the wiring connections.

1. The power wires of the I2C LCD display and the servo motor are connected to the breadboard.
2. Jumper wires are used to connect the power rails on the breadboard to the main power supply.
3. A jumper wire from the 5V pin on the Arduino board is connected to the power rail on the breadboard.

The power wires of the I2C LCD display and the servo motor are connected to the power rail on the breadboard using jumper wires. This ensures that they receive power from the main power supply.

Sources:

Product

Password Based Door Lock System Using Arduino and Keypad:

[Password Based Door Lock System Using Arduino and Keypad - YouTube](#)

Password-Based Door Lock System Using Arduino and Keypad:

[Password-Based Door Lock System Using Arduino and Keypad - YouTube](#)

Complications

LCD1602 I2C Address for Arduino explained:

[LCD1602 I2C Address for Arduino explained - YouTube](#)

Arduino IDE Libraries

Keypad library: [Arduino Playground - Keypad Library](#)

Servo motor library: [Servo - Arduino Reference](#)

I2C LED library: [GitHub - blackhack/LCD_I2C: Arduino library to control 16x2 LCD via an I2C adapter based on PCF8574T](#)

LiquidCrystal library: [GitHub - johnrickman/LiquidCrystal_I2C: LiquidCrystal Arduino library for the DFRobot I2C LCD displays](#)

EEPROM library: [EEPROM Library | Arduino Documentation](#)

Books and Websites

Last Minute ENGINEERS:

[In-Depth: Interfacing an I2C LCD with Arduino \(lastminuteengineers.com\)](#)

[How Servo Motor Works & Interface It With Arduino - Last Minute Engineers](#)

Quick-Start-Guide Books:

<https://www.az-delivery.de/pages/search-results-page?q=arduino%20book>

Courses

Arduino MASTERCLASS | Full Programming Workshop in 90 Minutes!:

[Arduino MASTERCLASS | Full Programming Workshop in 90 Minutes! - YouTube](#)

FreeCodeCamp (multiple):

[Arduino Course for Beginners - Open-Source Electronics Platform - YouTube](#)