Introduction

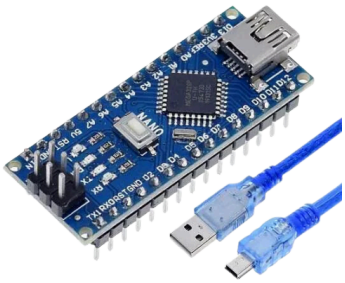
Creating secure and smart systems for everyday use is super important in our tech world. This project is all about making things safer by using Arduino to build a special lock system. This system has a keypad and a screen that shows information. Safety is a big deal, whether it's at home or in an office, so this project wants to give a good and changeable solution.

We're putting together different tech things, like the Arduino Nano, keypad, SG90 servo motor, and a screen. This shows that we can make strong security systems that go beyond the usual ways.

The process involves carefully combining hardware (the physical parts) and software (the computer programs) to make a smooth and good lock system. The Arduino Nano functions as the brain, and when you enter a secure code on the keypad, the motor moves to lock or unlock. The screen immediately informs you of what is happening, making it simple to use and keep an eye on things.

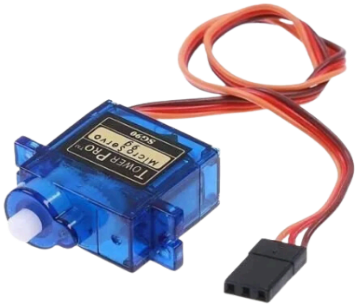
This project is not just about saying security is important. It's also about showing you how to accomplish it through each step. We want everyone, whether they are interested in technology or not, to be able to comprehend and test it out. It's a helpful project for making homes and offices safer using simple and practical solutions.

Equipment Used

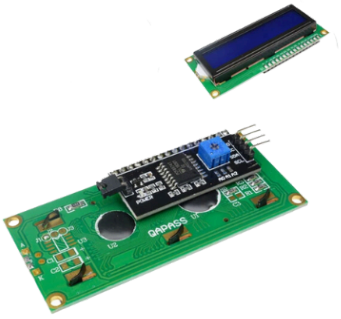


Arduino Nano board

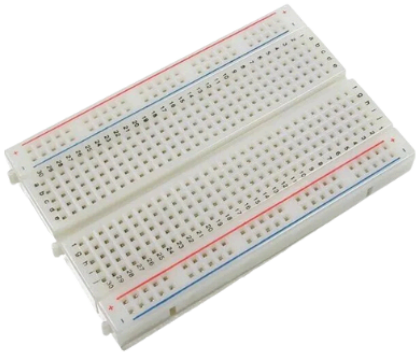
Keypad module



SG90 servo motor



LCD screen



Breadboard

Jumper wires

Door lock

Styrofoam

Description of each equipments

Arduino Nano:
The Arduino Nano is a compact microcontroller board, perfect for prototyping in limited spaces. Based on the ATmega328P, it offers versatile I/O capabilities for a variety of electronic projects.

Breadboard:
A fundamental tool for electronics prototyping, the breadboard enables easy and temporary circuit assembly without soldering. It provides a grid of interconnected holes for quick component connections.

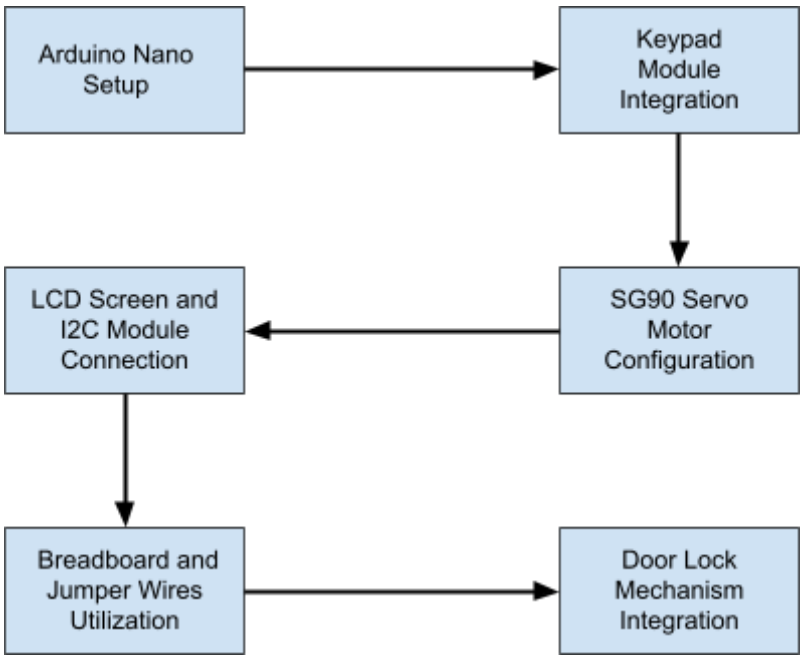
Servo Motor:
The servo motor is a precise rotary actuator commonly used in robotics and automation. It excels in maintaining specific angular positions, making it ideal for controlled movements in various applications.

Jumper Wires:
Jumper wires are flexible, insulated connectors used for temporary electrical connections on a breadboard or between components. They simplify circuit testing and modifications without the need for soldering.

I2C Display:
The I2C display utilizes the I2C protocol for easy interfacing with microcontrollers like Arduino. Ideal for presenting information in projects, these displays come in various sizes and types.

4x4 Keypad:
The 4x4 keypad is a matrix of 16 push buttons arranged in rows and columns. Commonly used for user input, it finds applications in security systems, digital locks, and menu navigation interfaces in electronic projects.

Working Flowchart



The methodology involves a series of carefully orchestrated steps and the utilization of specific equipment to create a secure Arduino-based lock system

Step 1: Arduino Nano Setup - Begin by setting up the Arduino Nano board as the central processing unit, ensuring proper connections and power supply.

Step 2: Keypad Module Integration - Connect the keypad module to the Arduino Nano board, establishing a secure communication link for user input.

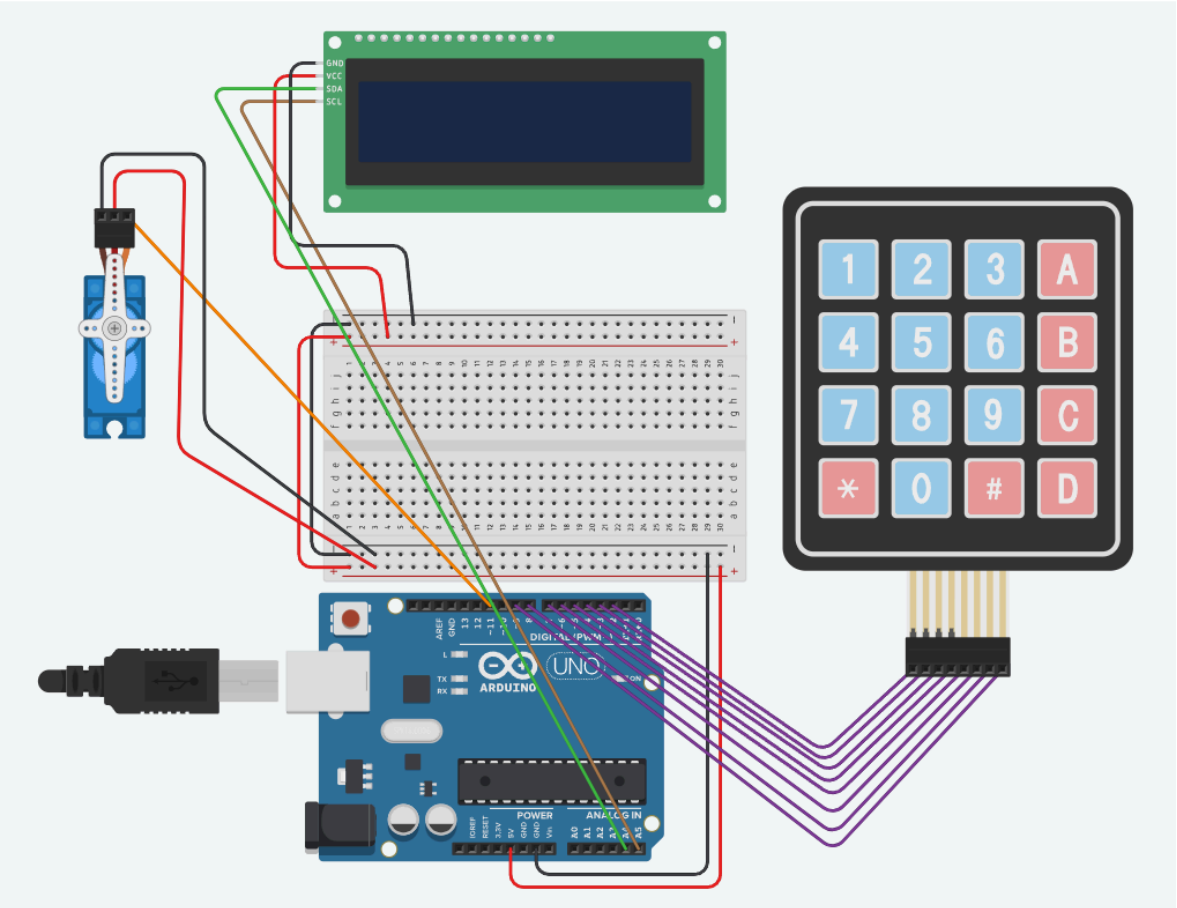
Step 3: SG90 Servo Motor Configuration - Integrate the SG90 servo motor to control the mechanical aspect of the lock. Program the Arduino to respond to specific keypad inputs by controlling the servo motor.

Step 4: LCD Screen and I2C Module Connection - Connect the LCD screen to the I2C module and link it to the Arduino for real-time display of system status and user feedback.

Step 5: Breadboard and Jumper Wires Utilization - Utilize the breadboard and jumper wires to ensure proper connections among the components, facilitating a neat and organized setup.

Step 6: Door Lock Mechanism Integration - Connect the system to a door lock, ensuring compatibility and secure engagement.

Circuit



Discussion

Here's a step-by-step guide on how to build it:

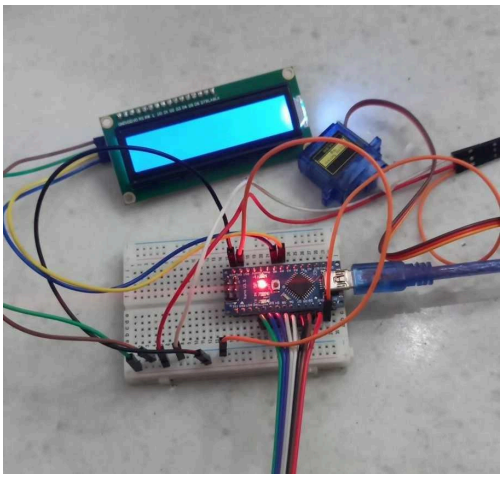
In the first step, establish the connection for the servo motor by attaching its power (red), ground (brown), and signal (orange) wires to the Arduino Nano. Channel the power wire to the 5V slot, the ground wire to GND, and the signal wire to a PWM capable pin on the Arduino, specifically D11.

Moving on to the second step, integrate the number pad with the Arduino Nano. The keypad features eight pins, divided into four rows and four columns. Connect the four row pins to D2, D3, D4, and D5 on the Arduino, and the four column pins to D6, D7, D8, and D9.

Proceed to the third step, where the focus is on connecting the I2C 16x2 LCD display module to the Arduino Nano. Ensure a seamless connection by attaching the SDA on the I2C module to A4, SCL to A5, VCC to 5V, and GND to GND on the Arduino.

For the fourth step, initiate the upload of the code. Connect the Arduino Nano to the computer, launch the Arduino IDE, select the appropriate board, designate the correct COM port, and execute the code upload.

Finally, in the fifth step, craft the model locker using Styrofoam cut into pieces based on specific measurements. Install the entire project within the locker door to ensure proper functionality and integration of the components.



Now on how it will work:

Upon powering up the Arduino Nano, the servo motor, LCD display, and buzzer undergo initialization. The LCD screen promptly showcases the welcoming message, "WELCOME TO DOOR LOCK SYSTEM," setting the stage for the system's operation.

Engaging with the system involves user input facilitated by the connected number pad. Pressing the corresponding keys on the pad allows users to input numerical values. The LCD display, in response, prompts the user with the message "Enter Password," creating an interactive interface. Real-time feedback is provided on the LCD, displaying the digits entered by the user.

The servo motor's control is seamlessly integrated into the process. As the user inputs numbers, the Arduino interprets the sequence. If the entered password aligns with the correct combination, the servo motor activates, resulting in the opening of the door lock. Conversely, if an incorrect password is entered, the servo motor remains static, leaving the door lock securely engaged. This systematic approach ensures a responsive and secure door lock system, enhancing user experience and overall functionality.

Conclusion:

In wrapping up, making the Arduino-based lock system was like putting together puzzle pieces to create a secure and easy-to-use solution. We carefully set up the Arduino Nano, made sure the SG90 servo motor fit perfectly, and added the LCD display for real-time information. Each step shows how much care we put into making a system that works well. We added extra protection to keep things safe. Looking ahead, we're excited about exploring more cool features to make our system even better for everyone who uses it.