Documentație

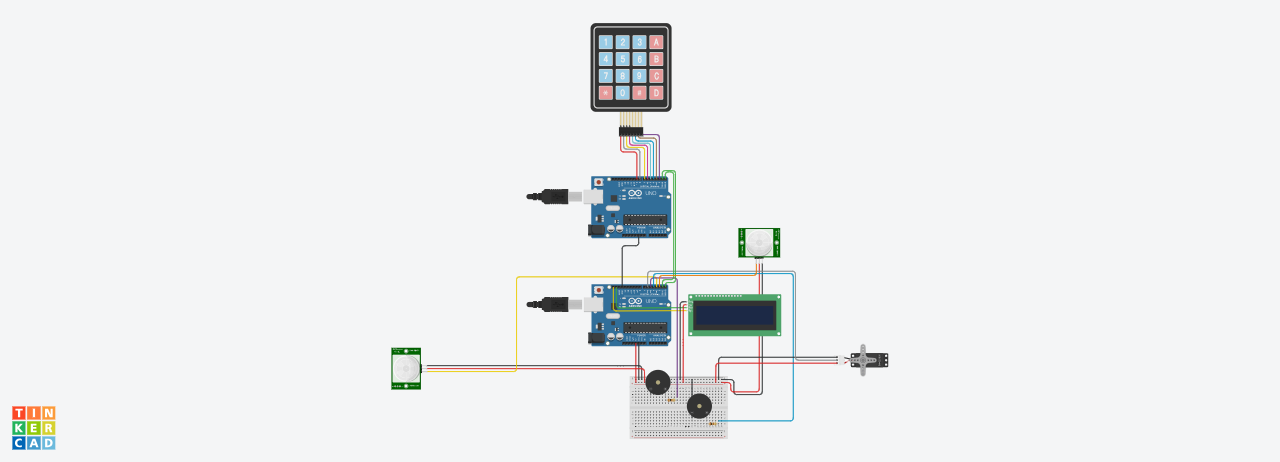
Implementarea unui sistem de alarmă bazat pe microcontroler, tastatură, ecran LCD, buzzer, senzori de prezență de tip PIR și servomotor – “passive infrared sensors”

Link : https://www.tinkercad.com/things/kStVZtnrkyA-sistem-de-alarma

Componente:

|  |  |  |  |
| --- | --- | --- | --- |
| NUME | PINI | INFO | CONECTAT LA PINII ARDUINO |
| PIR1 | GND, VCC, OUT | Detecteaza miscarile | D2 Arduino U1 |
| PIR2 | GND, VCC, OUT | Detecteaza miscarile | D3 Arduino U1 |
| PLACA ARDUINO U1 | GND, 5V | Arduino Uno R3 | GND, 5V |
| PLACA ARDUINO U3 | GND | Arduino Uno R3 | GND |
| REZISTENTA R1 | - | Rezistența de 100 Ω | - |
| REZISTENTA R2 | - | Rezistența de 100 Ω | - |
| KEYPAD1 | 8 pini | TASTATURA 4X4 | D2,D3,D4,D5,D6,D7,D8,D9 |
| Fire jumper | D, SDA, SCL, GND, VCC | Conecteaza componentele intre ele | D0-D9, VCC, GND, SCL, SDA |
| SERVO 1 | GND, VCC, OUT | Servomotorul armeaza si dezarmeaza sistemul | D6 Arduino U1 |
| PIEZO 1 | Pozitiv, Negativ | Emite semnalele sonore | D5 |
| PIEZO 2 | Pozitiv, Negativ | Emite semnalele sonore | D4 |
| U2 | GND, VCC, SDA, SCL | LCD 16 x 2 (I2C) | SDA, SCL |
| Breadboard | GND, VCC | Breadboard | GND, VCC |

Arhitectura Hardware:



A screenshot of a computer

Description automatically generated

Arhitectura Software:

Observăm că avem un ecran LCD pe care avem următoarele inscripții: „/”, „IGNITION-” , „FARURI-” și „.”. Printre componente avem un potențiometru. Acesta reprezintă motorul mașinii noastre, îl putem modifica și când trece de un anumit nivel va afișa „ACC” (la „IGNITION-”) iar atunci farurile vor fi pornite si se va putea acționa pedala de frână. Pentru a acționa pedala de frână vom apăsa butonul din dreptul LED-urilor roșii si pe ecran va apărea „#”.

În continuare putem modifica potențiometrul si se va afișa „ON” iar farurile vor fi in continuare pornite și frâna va putea fi apăsată. Daca potențiometrul este sub un anumit nivel , pe ecran se va afișa atât la starea motorului cât și la faruri „OFF” iar farurile de la frână nu vor funcționa chiar dacă va fi acționată pedala.

Dacă modificăm fotorezistențele de jos astfel încât să depășească un anumit ușile se vor deschide și se va schimba statusul acestora pe display.

Funcții folosite:

* void setup() - inițializează componentele hardware și portul serial la începutul programului
* void loop() - execută în mod continuu verificări ale stării senzorilor și a parolați primite, afișând mesaje corespunzătoare pe ecranul LCD și declanșând alarme în caz de nevoie
* void alert(String message) - afișează un mesaj pe ecranul LCD
* void print\_states() - afisează starea senzorilor PIR pe portul serial
* void zone\_libere() - afișează un mesaj pe ecranul LCD indicând că nu sunt mișcări detectate și oprește orice sunet de alarmă
* void alarm\_trigger() - gestionează declanșarea alarmei în funcție de starea senzorilor PIR
* void receive\_password() - primește și salvează o parolă de la portul serial

Librării folosite:

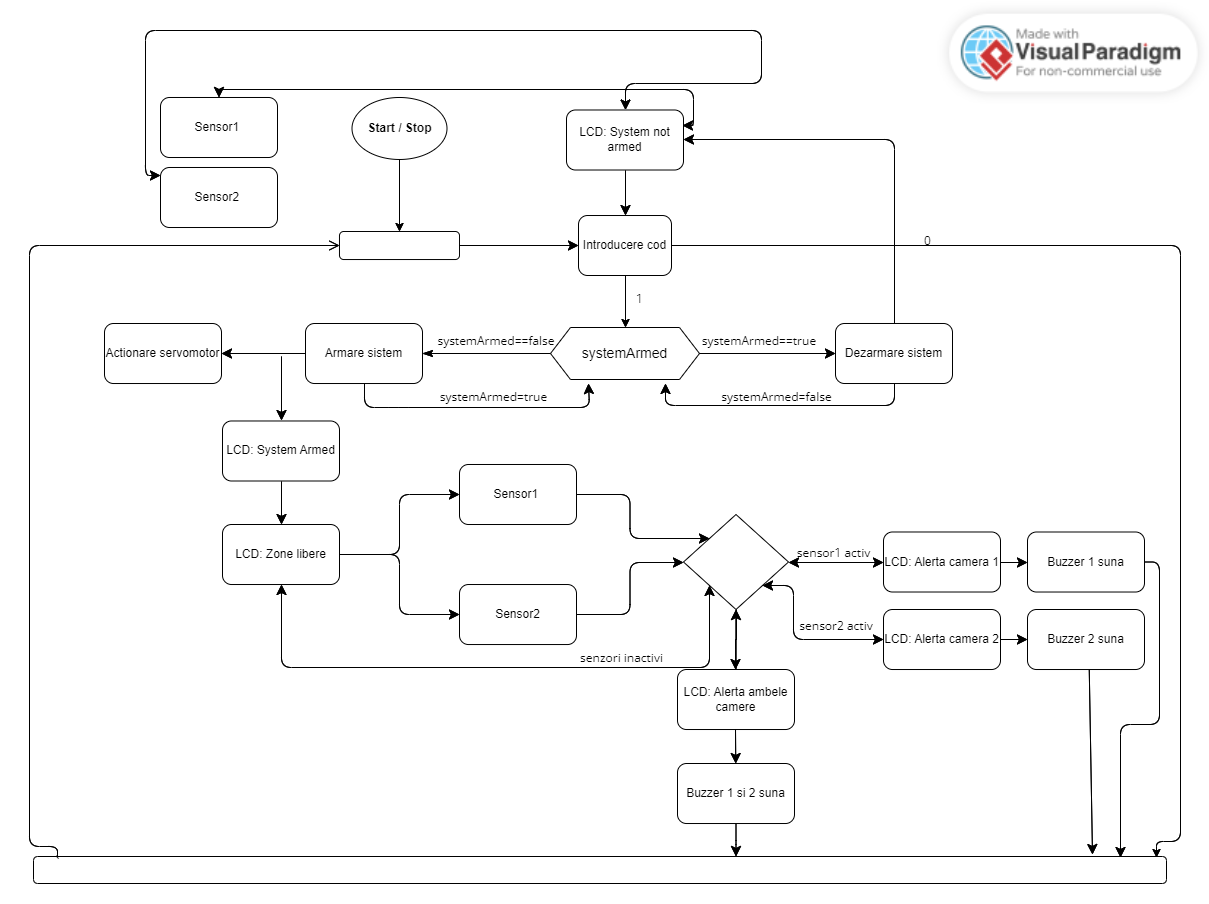
#include <Adafruit\_LiquidCrystal.h> - librărie folosită pentru a manipula cu ușurință un ecran de tip LCD

#include <Servo.h> - librăria este utilizată pentru a controla servo-motoarele, ea permite atașarea, detasarea și controlul precis al poziției unui servo-motor

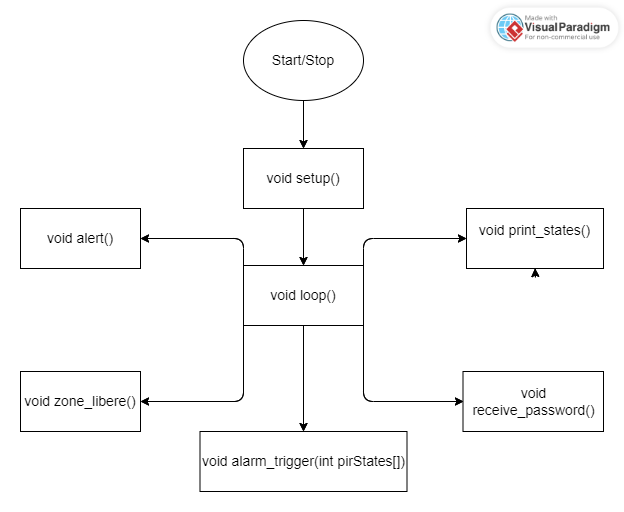
#include <string.h> - librărie folosită pentru a furniza funcții pentru manipularea șirurilor de caractere

#include <Keypad.h> - librăria este folosită pentru a lucra cu tastaturile matriceale, ea oferă funcții pentru detectarea apăsărilor de taste și pentru interpretarea caracterelor asociate fiecărei taste din matricea tastaturii

Schema Logică:



Schema apelării funcțiilor:



Concluzii:

Dificultăți:

* Comunicarea serială între cele două plăci

Bibliografie:

* SENSOR PIR:

https://www.instructables.com/How-to-Use-a-PIR-Motion-Sensor-With-Arduino/

<https://randomnerdtutorials.com/arduino-with-pir-motion-sensor/>

* SERVO:

<https://www.instructables.com/Arduino-Servo-Motors/>

* LCD:

<https://www.instructables.com/Arduino-Interfacing-With-LCD-Without->Potentiometer/

https://howtomechatronics.com/tutorials/arduino/lcd-tutorial/

https://docs.arduino.cc/learn/electronics/lcd-displays/

<https://www.youtube.com/watch?v=IM3d0dlAOJk>

* BUZZER: <https://www.ardumotive.com/how-to-use-a-buzzer-en.html>
* UART INTRE 2 PLACI:

https://www.instructables.com/Communication-Between-Two-Arduino-Using-UART/

<https://www.electroniclinic.com/serial-communication-between-two->arduino-boards/