Universitatea Tehnica din Cluj-Napoca Facultatea de Automatica si Calculatoare

Departamentul Calculatoare si tehnologia informatiei

-Seif-

Student: Jitaru Vasile Dumitru

Grupa: 30238

Profesor indrumator: Mircea Paul Muresan

Cuprins:

1. Specificare proiect
2. Proiectare si schema electrica
3. Manual de utilizare
4. Concluzii si dezvoltari ulterioare
5. Referinte
6. **Specificatie proiect**

Proiectul de fata consta in implementarea, folosind o placuta de dezvoltare Arduino a unui seif. Pentru deschiderea circuitului este folosita o metoda de dubla autentificare explicata pe larg in sectiunea 3.

**2.Proiectare si Schema electrica:**

Pentru implementarea acestui proiect s-au folosit urmatoarele componente:

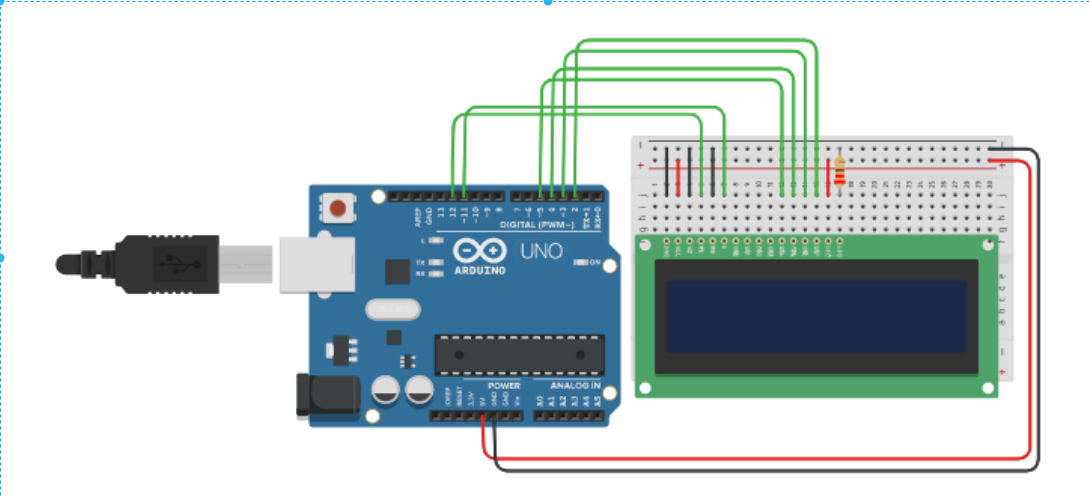
* Placa de dezvoltare Arduino Mega 2560
* Breadboard
* Display LCD
* KeyPad 4x4
* Modul Buzzer
* Motor servo sg90 9g
* Modul Wifi esp8266
* Led
* 2 rezistente
* Fire jumper (destul de multe in jur de 35-40)

2.1 Montaj

Pentru a nu se aglomera figura si pentru a usura citirea ei, in cele ce urmeaza o sa prezint o descompunere pe module a proiectului.

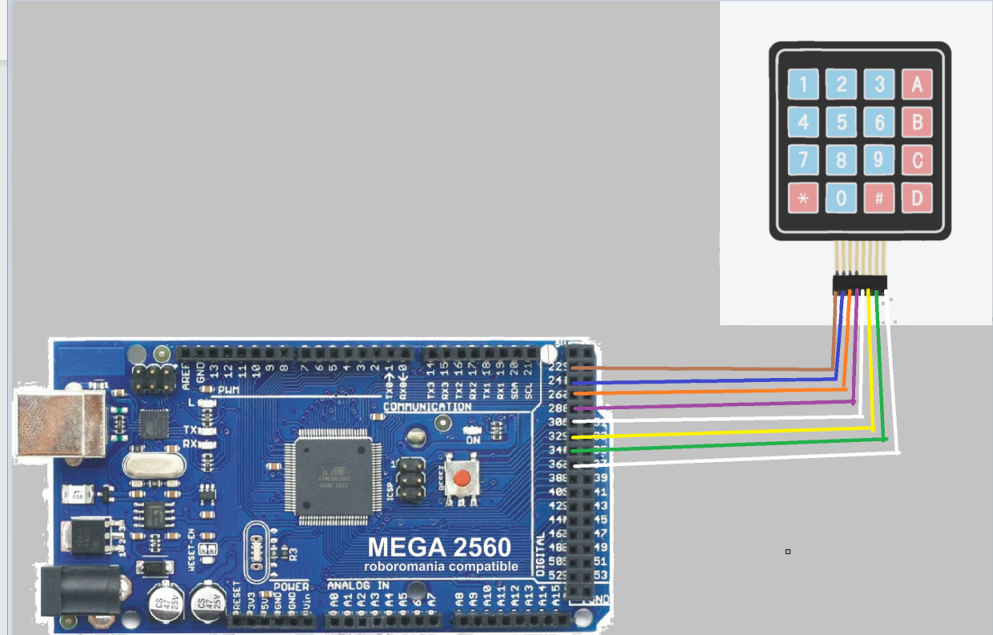
* + - * 1. Conectarea ecranului LCD:

Conectarea ecranului LCD a fost efectuala dupa modelul de mai jos, utilizandu-se exact aceiasi pini doar corespunzatori pentru placa Mega. Schema de mai jos a fost creeata in ThinkerCad.



* + - * 1. Conectarea tastaturii

Schema de mai jos a fost facut in paint. Pinii care sunt legati la tastatura din figura corespund 1 la 1 cu pinii utilizati la proiect.



* + - * 1. Conectarea modulului wifi

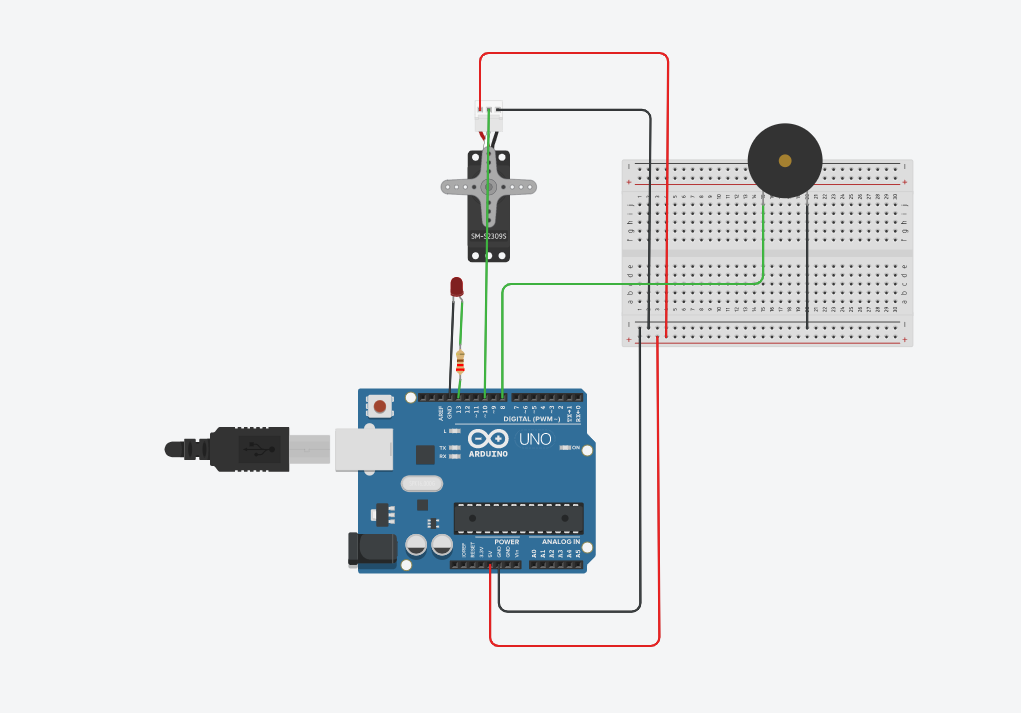
Schema de mai jos a fost preluata de la [1], singura diferenta prezenta intre schema de mai jos si proiectul in sine este faptul ca pinii TX si RX de la modul eps8266 sunt conectati la RX0 si TX0 nu la RX1 si TX1.

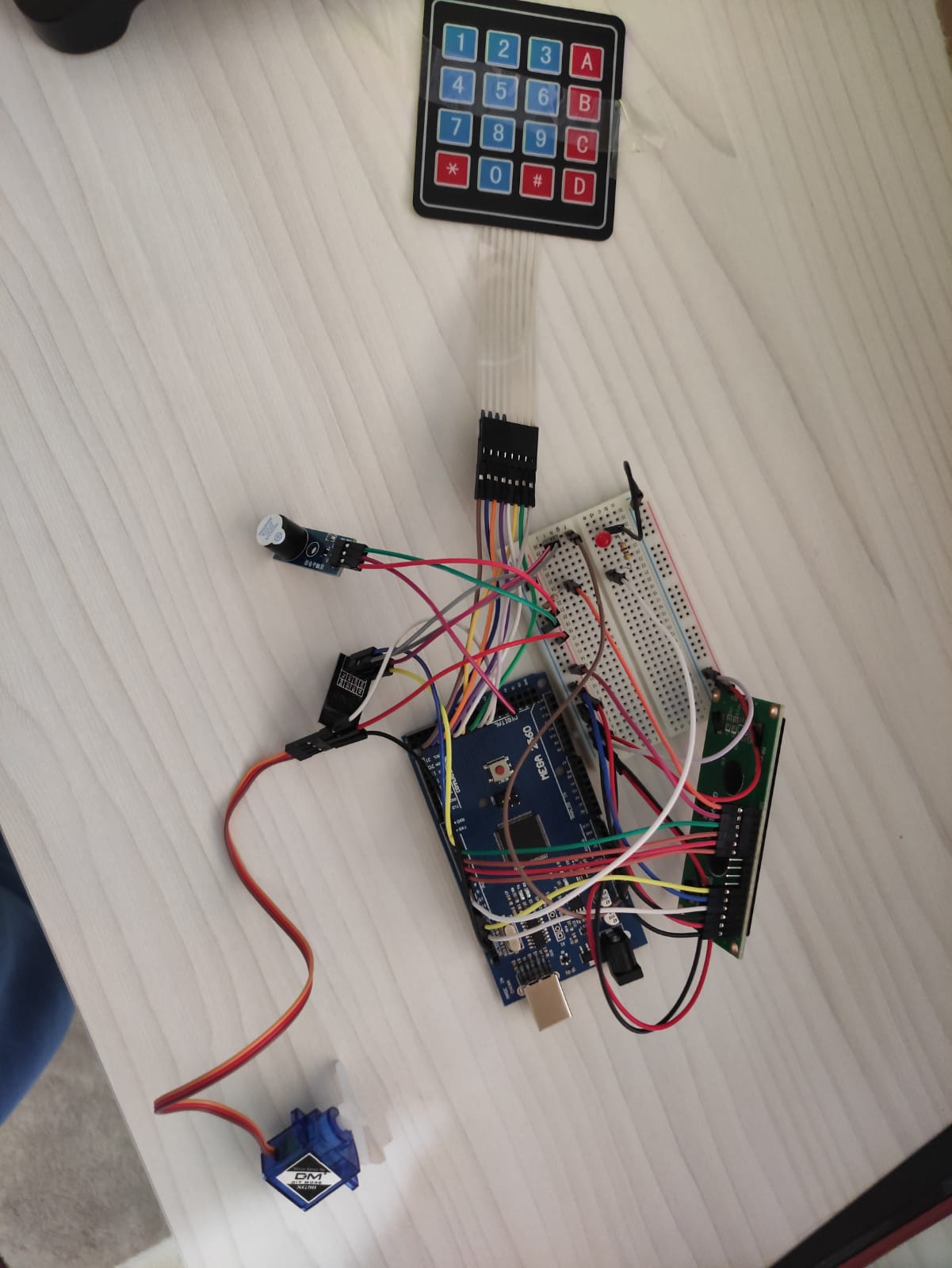
A picture containing text, electronics, screenshot

Description automatically generated

* + - * 1. Conectarea motorasului servo al buzzerului, si al ledului.

In figura de mai jos este prezentat cum au fost conectate motorul servo, buzzerul si ledul. Figura a fost creeara in ThinkerCad si dupa adaptata pentru proiectul de fata. Pinul de intrare pentru motorul servo este pinul 38, pentru led este pinul 7, iar pentru buzzer este pinul 46.



Montarea fizica a circuitului este urmatoarea: 

**2.2 Comentarea codului.**

In cele ce urmeaza o sa se prezinte funtionalitatea fiecarei funcii din codul sursa al proiectului.

1. void setup()
2. {
3. RemoteXY\_Init ();
4. pinMode(PIEZO, OUTPUT);
5. pinMode (PIN\_LED, OUTPUT);
6. col = 0;
7. lcd.begin(16, 2);
8. myservo.attach(38);
9. ServoClose();
10. digitalWrite(PIEZO, LOW);
12. }

RemoteXY\_Init() este o functie care se gaseste in libraria RemoteXY.h, librarie care se ocupa cu conexiunea cu modului wifi. La linia 4 si 5 se seteaza pinii de iesire ai ledului, care o sa indice daca dispozitivul mobil este conectat cu succes la modului wifi si este gata sa transmita date, si buzzerul care va scoate sunete specifice pentru parola corecta si pentru parola gresita.

1. void playMelody(int \*melody, int \*noteDurations, int notesLength)
2. {
4. for (int thisNote = 0; thisNote < notesLength; thisNote++) {
5. int noteDuration = 1000 / noteDurations[thisNote];
6. tone(PIEZO, melody[thisNote], noteDuration);
7. int pauseBetweenNotes = noteDuration \* 1.30;
8. delay(pauseBetweenNotes);
9. noTone(PIEZO);
10. }
11. }

Functie de mai sus primeste ca si parametri o melodie, durata notelor si lungimea lor.Cu ajutorul acesteia se va programa buzzerul sa cante unde din cele doua tonuri.

1. void checkWifiPass(){
2. permission = 1;
3. for(int i = 0; i < 5; i++){
4. if(RemoteXY.passKey[i] != passW[i])
5. permission = 0;
6. break;
7. }
8. }

Functie de mai sus va verifica daca parola transmisa prin modulul wifi este corecta. Daca da atunci va deschide sistemul, daca nu acesta va ramane inchis.

1. void ServoClose()
2. {
3. for (int pos = 90; pos >= 0; pos -= 5) {
4. myservo.write(pos); // tell servo to go to position in variable 'pos'
5. delay(15); // waits 15ms for the servo to reach the position
6. }
7. }
9. void ServoOpen()
10. {
11. for (int pos = 0; pos <= 90; pos += 5) {
12. myservo.write(pos); // tell servo to go to position in variable 'pos'
13. delay(15); // waits 15ms for the servo to reach the position
14. }
15. }

Functiile de mai sus se ocupa cu controlul motorasului Servo. ServoOpen() va scrie unghiul motorasului astfel incar acesta sa indice faptul ca seiful este inchis, ServoClose() va face fix opusul va inchide seiful.

1. void backSpace(){
2. col--;
3. lcd.setCursor(col,0);
4. lcd.print(" ");
5. }
7. void backSpace1(int i){
8. i--;
9. lcd.setCursor(i,0);
10. lcd.print(" ");
11. }

Functiile de mai sus sunt functii care au efectul de a sterge ultimul caracter scris pe lcd.

1. void typePassword(){
2. col = 0;
3. clearLCD();
4. lcd.setCursor(col,0);
5. lcd.print("Intruduceti");
6. lcd.setCursor(col,1);
7. lcd.print("parola");
8. for(int i = 0; i<5 ;i ++){
9. col = i;
10. lcd.setCursor(i,0);
12. char key = customKeypad.getKey();
13. if (key){
14. pass[i] = key;
15. if(pass[i] == 'A'){
16. backSpace1(i);
17. i=i-2;
18. }
19. else{
20. lcd.print(pass[i]);
21. if(i == 0){
22. clearLCD1();
23. }
24. delay(400);
25. lcd.setCursor(i,0);
26. lcd.print('\*');
27. }
28. }else
29. i--;
30. }
31. col++;
32. verificaParola();
33. }

Cu ajutorul functiei de mai sus se va introduce parola seifului. Prima data se scrie pe ecran mesajul „Introduceti parola” dupa care se citeste de la tastatura numerica 5 caractere care se vor scrie pe display, dupa care aceste caractere vor fii inlocuite pe display de caracterul „\*” pentru a da impresia introducerii unei parole. Daca se introduce caracterul „A” acesta are rolul de backspace. La linie 32 se apeleaza functia care va verifica daca parola este corecta, functie a carei cod este mai jos.

1. void verificaParola()
2. {
3. int ok = 1;
4. for(int i=0;i<=5;i++)
5. if(pass[i] != mainPass[i] ){
6. lcd.setCursor(0,0);
7. lcd.print("PAROLA GRESITA");
8. lcd.setCursor(0,1);
9. lcd.print("PRESS C TO RETRY");
10. ok = 0;
11. playCloseMelody();
12. delay(400);
13. clearLCD()
14. break;
15. }
16. if (ok == 1){
17. while(permission==0){
18. lcd.setCursor(0,0);
19. lcd.print("SE ASTEAPTA");
20. delay(400);
21. }
22. clearLCD();
23. lcd.setCursor(0,0);
24. lcd.print("PAROLA CORECTA");
25. ServoOpen();
26. playOpenMelody();
27. delay(400);
28. clearLCD();
29. }
30. }

Aceasta functie are rolul de a verifica daca parola introduse este corecta. Daca da atunci se verifica daca sistemul este deschis si daca este atunci se scrie „Parola corecta” pe lcd si se va lansa un ton care sa indice ca parola este corecta. Daca nu este corecta atunci se va afisa „Parola gresita ” si „Press c to retry.

1. void clearLCD(){
2. lcd.setCursor(col,0);
3. lcd.print(" ");
4. lcd.setCursor(col,1);
5. lcd.print(" ");
6. }
8. void clearLCD1(){
9. lcd.setCursor(1,0);
10. lcd.print(" ");
11. lcd.setCursor(0,1);
12. lcd.print(" ");
13. }

Aceste functii le-am folosit pentru a putea sterge continutul de pe Lcd. Daca foloseam lcd.clear(), functia implementata pentru lcd din biblioteca aferenta imi afisa ceva caractere ciudate si dupa ce m-am interesat pe pe diferite forumuri am decis sa imi implementez singur aceste funtii.

1. void loop()
2. {
3. digitalWrite(PIEZO, LOW);
4. RemoteXY\_Handler ();
5. if(RemoteXY.connect\_flag == 1){
7. checkWifiPass();
8. }
9. digitalWrite(PIN\_LED, (RemoteXY.connect\_flag==0)?LOW:HIGH);
10. if(permission == 0){
11. lcd.setCursor(0,0);
12. lcd.print("SE ASTEAPTA");
13. lcd.setCursor(0,1);
14. lcd.print("PAROLA WIfI");
15. delay(400);
16. }
17. else{
18. contor++;
19. if(contor == 1){
20. clearLCD();
21. lcd.setCursor(0,0);
22. lcd.print("PAROLA WIFI");
23. lcd.setCursor(0,1);
24. lcd.print("ACCEPTATA ");
25. delay(400);
26. }
27. char customKey = customKeypad.getKey();
28. if(customKey == 'C'){
29. typePassword();
30. }
31. if(customKey == 'B'){
32. ServoClose();
33. }
35. }
36. }

In functia loop se apeleaza functiile principale. La linia 4, functia RemoteXY\_Handler() este o alta functie care se ocupa de conexiunea cu telefonul prin wifi. La linia 5 se verifica daca exista vre-un dispozitiv conectat la modul, si daca da se aprinde si ledul care indica acest lucru(linia 9). Daca sistemul este deschis (s-a primit parola corecta) atunci se va continua cu functionalitatea aplicatiei. Se citeste un caracter si daca caracterul este „C” atunci se va putea introduce parola. Daca caracterul este „B” se va putea inchide din nou seiful.

Pasul 1: Se conecteaza placuta de dezvoltare Arduino la calculator unde urmeaza sa fie programata. Prima data inainte sa fie programata trebuie scoase firele care intra la pinii rx si tx. Dupa programare se baga la loc.

Pasul2: Pe telefonul Android se conecteaza la reteaua WiFi RemoteXY, dupa care se va deschide aplicatia si se selecteaza butonul cu inconita Wifi care ne va duce la panoul urmator, unde se va introduce parola care va deschide sistemul. Daca telefonul a fost connectat cu succes la placuta se va porni ledul conectat la pinul 7.

Pasul3. Pe ecranul LCD ar trebuii acum sa apara un text prin care reiese faptul ca sistemul a fost deschis. Pentru deschiderea seiful se va apasa tasta C dupa care se va introduce parola locala a acestuia formata din 5 caractere. Daca s-a sesizat gresirea parolei prin tasta A este disponibila o operatie de BackSpace. Daca parola este corecta atunci atunci va aparea mesajul care sa ateste acest lucru(„Parola corecta”) pentru cateva secunde doar dupa se va scrie unghiul motorasului servo, si va suna un anumit ton. Daca mesajul este gresit atunci va aparea un text care ne va indica ce putem sa facem in caz ca parola este greista si va suna un alt ton diferit de cel precedent. Daca se doreste reintroducerea parolei se apasa tasta C.

Pasul 7: Daca se doreste inchiderea seifului se va apasa tasta B ceea ce va rescrie unghiul motorasului servo la valoarea initiala.

**4.Concluzii si dezvoltari ulterioare**

Proiectul de fata poate avea diferite dezvoltari ulterioare, cum ar fii: implementarea une metode mai sigura de conectare cu un dispozitiv mobile, adaugarea de diferiti senzori, poate chiar autentificarea cu amprenta folosind un senzor ca atare sau telefonul mobil.

**5. Referinte**

[1] https://forum.arduino.cc/index.php?topic=629684.0