

Relatório Final - Sistema para CPD

Alunos: Anderson Gaspar de Medeiros

Daniel Cabral Correa

Natália Adriana Miranda

Professor: Adilson Jair Cardoso

Disciplina: Sistemas Embarcados

25 de junho de 2020

1 Implementação

Este projeto tem como objetivo elaborar um sistema de monitoramento para Centro de Processamento de Dados(CPD). Permitindo o controle de acesso identificado por senha, monitoramento da temperatura e luz do ambiente e verificação de presença na sala. Também será implementado um sistema de alerta. O projeto possui dois módulos, um para o teclado e outro principal. A comunicação entre eles é feita utilizando a interface I^2C , de modo mestre/escravo.

1.1 Descrição das Funcionalidades

- Controle de Acesso: Para adentrar na sala o usuário deve digitar sua senha no teclado acoplado a parede. Caso autorizado o usuário terá acesso a sala de controle;
- Controle Temperatura: Quando a temperatura da sala exceder um limite o usuário será informado, utilizando um sensor de temperatura;
- Controle de Presença: Verifica se existe a presença de alguém na sala, utilizando um sensor de presença;
- Controle de Luminosidade: Verifica a luminosidade do ambiente.

1.2 Etapas de Implementação

A implementação do projeto foi realizada nas seguintes etapas:

1. Confecção do hardware utilizando o programa Proteus para simulação;
2. Confecção do software;
3. Implementação do Teclado ;
4. Sensor de Presença;
5. Sensor de Luminosidade.
6. Testes dos componentes separadamente.
7. Implementação do Hardware completo com todos os componentes;
8. Implementação da Interface de Comunicação I^2C .

2 Imagens

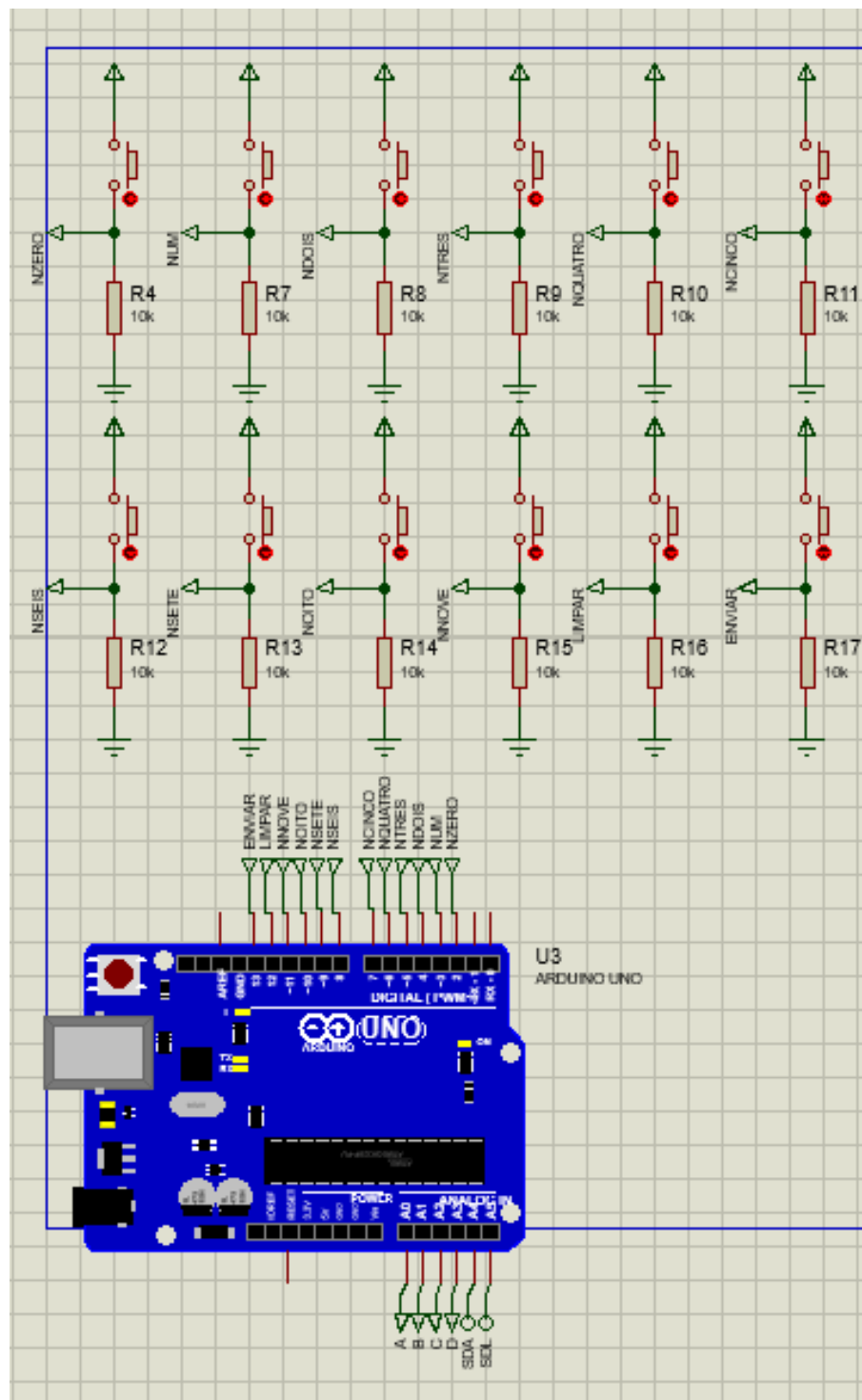


Figura 1 - Diagrama da implementação do teclado.

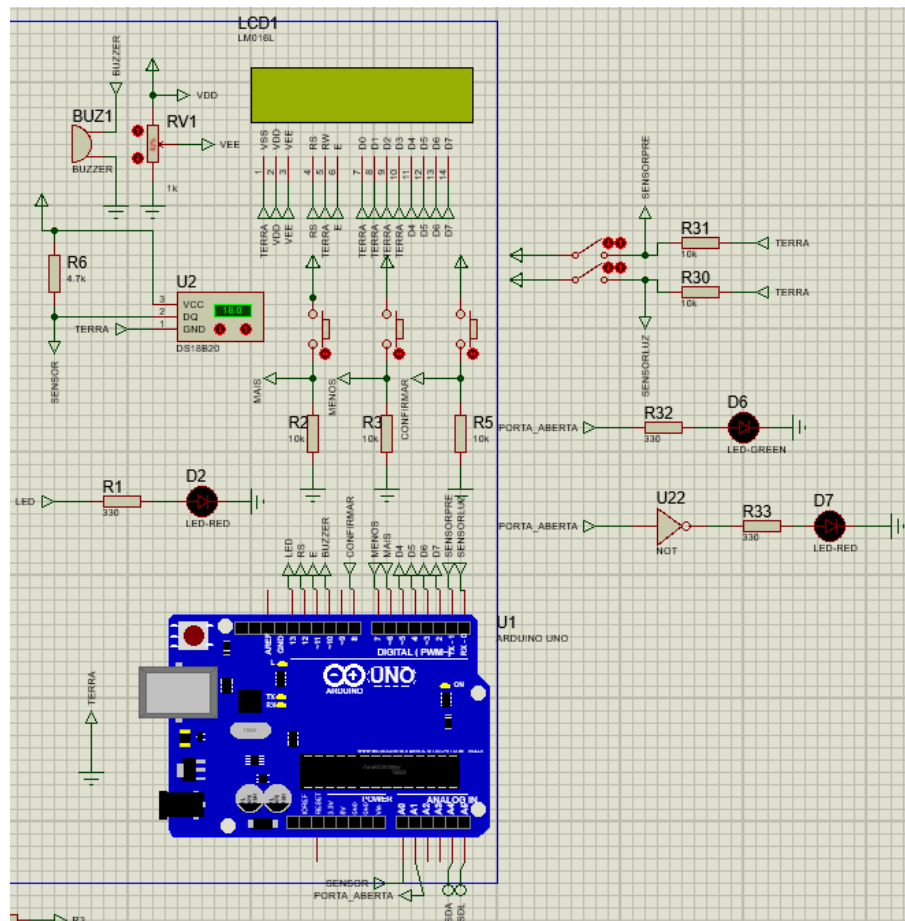


Figura 2 - Diagrama da implementação do LCD, Buzzer e do sensor de temperatura.

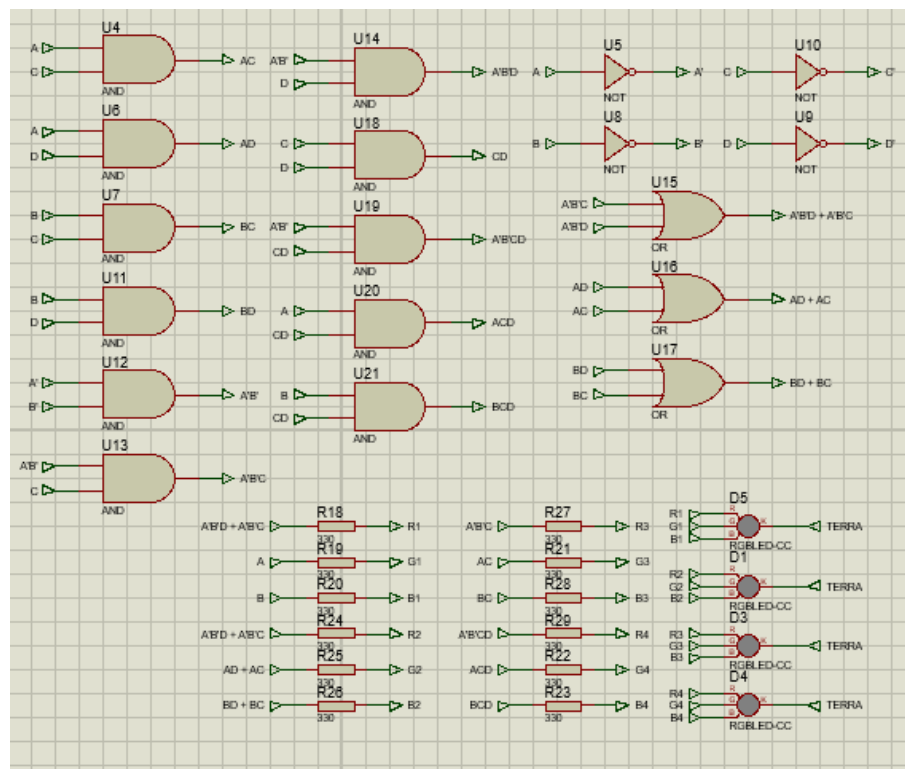


Figura 3 - Diagrama da implementação do mapa de *Karnaugh* em portas lógicas.

X	luz	R1	G1	B1	R2	G2	B2	R3	G3	B3	R4	G4	B4
0000	apagado	0	0	0	0	0	0	0	0	0	0	0	0
0001	red	1	0	0	1	0	0	0	0	0	0	0	0
0010	red	1	0	0	1	0	0	1	0	0	0	0	0
0011	red	1	0	0	1	0	0	1	0	0	1	0	0
0100	blue	0	0	1	0	0	0	0	0	0	0	0	0
0101	blue	0	0	1	0	0	1	0	0	0	0	0	0
0110	blue	0	0	1	0	0	1	0	0	1	0	0	0
0111	blue	0	0	1	0	0	1	0	0	1	0	0	1
1000	green	0	1	0	0	0	0	0	0	0	0	0	0
1001	green	0	1	0	0	1	0	0	0	0	0	0	0
1010	green	0	1	0	0	1	0	0	1	0	0	0	0
1011	green	0	1	0	0	1	0	0	1	0	0	1	0
1100	yellow	0	1	1	0	0	0	0	0	0	0	0	0
1101	yellow	0	1	1	0	1	1	0	0	0	0	0	0
1110	yellow	0	1	1	0	1	1	0	1	1	0	0	0
1111	yellow	0	1	1	0	1	1	0	1	1	0	1	1
		A'B'D + A		B	A'B'D + AD + AC		BD + BC	A'B'C	AC	BC	A'B'CD	ACD	BCD

Figura 4 - Planilha da implementação do mapa de *Karnaugh* em portas lógicas.

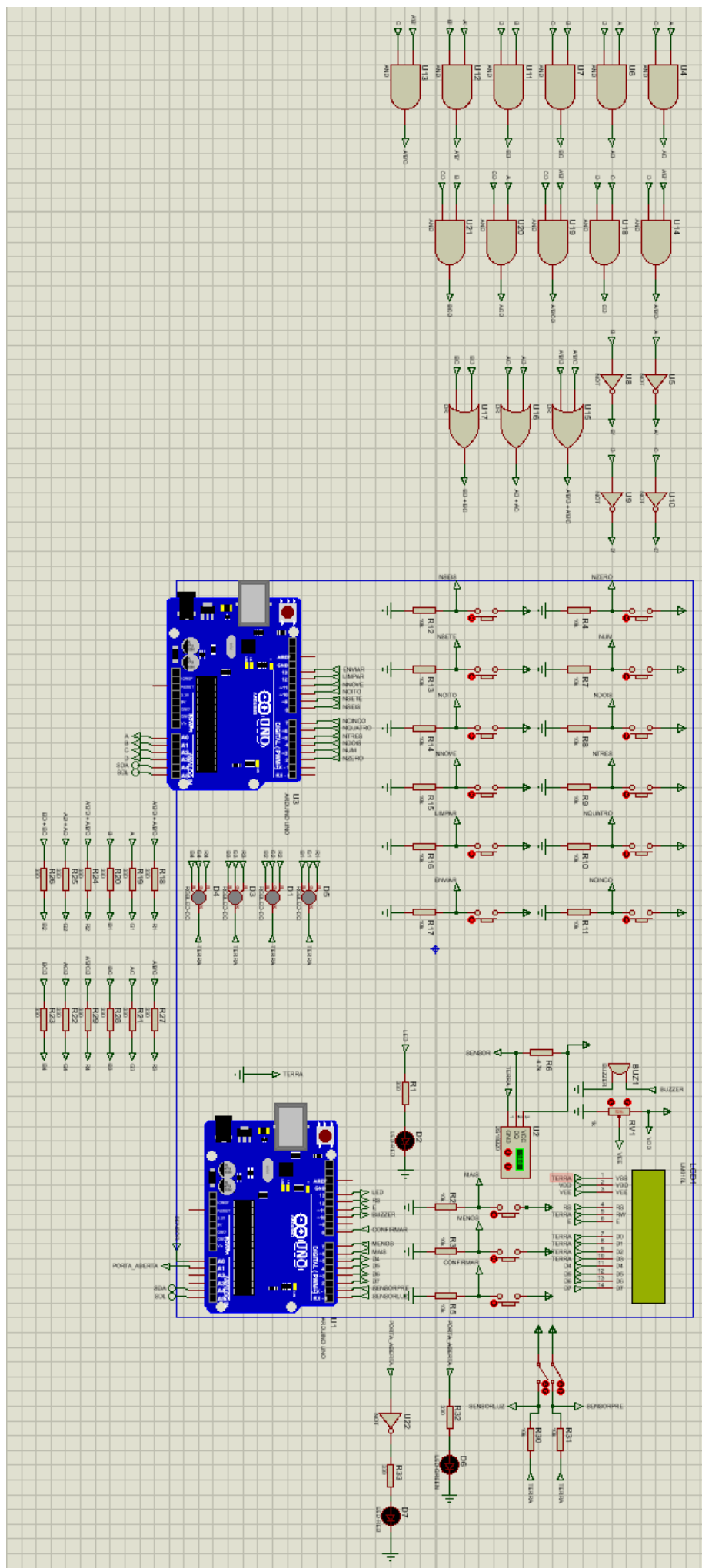


Figura 5 - Diagrama da implementação completa do projeto.

3 Códigos

3.1 Código do módulo principal

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <LiquidCrystal.h>
4
5 LiquidCrystal lcd(12,11,5,4,3,2);
6
7 const byte SLAVE_ADDRESS = 42; // end[0xFFFD] do escravo a ser enviado
   dados!
8 #define DS18B20_PIN A0
9
10 char msg1[16] = "[ ]SIM  [*]NAO";
11 char msg2[16] = "[*]SIM  [ ]NAO";
12 int mais;
13 int menos;
14 int confirma;
15 int raw_temp;
16 int delaytime=200;
17 float temperaturaAtual;
18 float temperaturaLimite = 23.00;
19 int menu = 0;
20
21 volatile byte recebido;
22 int funcionamento = 0;
23 int acesso = 0;
24 int temperatura = 0;
25 int presenca = 0;
26 int luminosidade = 0;
27 int alarme = 0;
28
29
30 int sensorPresenca = 0;
31 int sensorLuz = 0;
32 int senhaMax = 9530;
33 int autorizacaoAcesso = 0;
34 int porta_aberta =0;
35
36 bool ds18b20_start(){
37     bool ret = 0;
38     digitalWrite(DS18B20_PIN, LOW); // Send reset pulse to the
   DS18B20 sensor
39     pinMode(DS18B20_PIN, OUTPUT);
40     delayMicroseconds(500); // Wait 500 us
41     pinMode(DS18B20_PIN, INPUT);
42     delayMicroseconds(100); //wait to read the DS18B20
   sensor response
43     if (!digitalRead(DS18B20_PIN)) {
44         ret = 1; // DS18B20 sensor is present
45         delayMicroseconds(400); // Wait 400 us
46     }
47     return(ret);
48 }
49
50 void ds18b20_write_bit(bool value){
51     digitalWrite(DS18B20_PIN, LOW);
52     pinMode(DS18B20_PIN, OUTPUT);
53     delayMicroseconds(2);
54     digitalWrite(DS18B20_PIN, value);
55     delayMicroseconds(80);
56     pinMode(DS18B20_PIN, INPUT);
57     delayMicroseconds(2);
```

```

58 }
59
60 void ds18b20_write_byte(byte value){
61     byte i;
62     for(i = 0; i < 8; i++){
63         ds18b20_write_bit(bitRead(value, i));
64     }
65
66 bool ds18b20_read_bit(void) {
67     bool value;
68     digitalWrite(DS18B20_PIN, LOW);
69     pinMode(DS18B20_PIN, OUTPUT);
70     delayMicroseconds(2);
71     pinMode(DS18B20_PIN, INPUT);
72     delayMicroseconds(5);
73     value = digitalRead(DS18B20_PIN);
74     delayMicroseconds(100);
75     return value;
76 }
77
78 byte ds18b20_read_byte(void) {
79     byte i, value;
80     for(i = 0; i < 8; i++){
81         bitWrite(value, i, ds18b20_read_bit());
82     }
83     return value;
84 }
85
86 bool ds18b20_read(int *raw_temp_value) {
87     if (!ds18b20_start()) // Send start pulse
88         return(0); // Return 0 if error
89     ds18b20_write_byte(0xCC); // Send skip ROM command
90     ds18b20_write_byte(0x44); // Send start conversion
91     // command
92     while(ds18b20_read_byte() == 0); // Wait for conversion
93     // complete
94     if (!ds18b20_start()) // Send start pulse
95         return(0); // Return 0 if error
96     ds18b20_write_byte(0xCC); // Send skip ROM command
97     ds18b20_write_byte(0xBE); // Send read command
98     *raw_temp_value = ds18b20_read_byte(); // Read temperature LSB
99     // byte and store it on raw_temp_value LSB byte
100     *raw_temp_value |= (unsigned int)(ds18b20_read_byte() << 8); //
101     // Read temperature MSB byte and store it on raw_temp_value MSB byte
102     return(1); // OK --> return 1
103 }
104
105 float medirTemp(int raw_temp){
106     // txt[0] = 223; // Put degree symbol[U{FF}D]
107     if(ds18b20_read(&raw_temp)) {
108         float resultado = (float)raw_temp / 16;
109         return(resultado);
110     }
111     return 0;
112 }
113
114 void inicializar(){
115     lcd.begin(16,2);
116     lcd.setCursor(0,0);
117     lcd.print("CONTROLE CPD");
118     lcd.setCursor(0,1);
119     lcd.print("NDA3000");
120     delay(300);
121     lcd.clear();
122 }

```

```

118
119 void alerta(int tempo){
120     int count=0;
121     int cont=2000 /( tempo * 2);
122     while(count<=cont){
123         digitalWrite(13, HIGH);
124         digitalWrite(10, HIGH);
125         delayMicroseconds(tempo);
126         digitalWrite(10, LOW);
127         delayMicroseconds(tempo);
128         count++;
129     }
130     digitalWrite(13, LOW);
131 }
132
133 void receiveEvent (int howMany){
134     if (Wire.available () > 0){
135         recebido = Wire.read();
136         if(recebido=='1'){
137             digitalWrite(A1, HIGH);
138         }
139         else{
140             digitalWrite(A1, LOW);
141         }
142     }
143 }
144 }
145
146 void setup() {
147     pinMode(A0, INPUT);
148     pinMode(A1, OUTPUT);
149     pinMode(0, INPUT);
150     pinMode(1, INPUT);
151     pinMode(6, INPUT);
152     pinMode(7, INPUT);
153     pinMode(8, INPUT);
154     pinMode(10, OUTPUT);
155     pinMode(13, OUTPUT);
156     Wire.begin (SLAVE_ADDRESS);
157     Wire.onReceive(receiveEvent); // register event
158     Serial.begin(9600);          // start serial for output
159     inicializar();
160 }
161
162 void loop() {
163     sensorLuz = digitalRead(0);
164     sensorPresenca = digitalRead(1);
165     mais = digitalRead(6);
166     menos = digitalRead(7);
167     confirma = digitalRead(8);
168     temperaturaAtual = medirTemp(raw_temp);
169
170     if ((funcionamento==1)&&(alarme==1)&&(sensorPresenca==1)) {
171         menu = 8;
172         lcd.clear();
173     }
174     else if ((funcionamento==1)&&(temperaturaAtual > temperaturaLimite)) {
175         menu = 9;
176         lcd.clear();
177     }
178     else if ((funcionamento==1)&&(sensorPresenca==0)&&(sensorLuz==1)) {
179         menu = 10;
180         lcd.clear();
181     }

```



```

182 switch (menu){
183     case 0:
184         lcd.setCursor(0,0);
185         lcd.print("SERVICO ATIVADO");
186         if (funcionamento==0){
187             lcd.setCursor(0,1);
188             lcd.print(msg1);
189         }
190         else{
191             lcd.setCursor(0,1);
192             lcd.print(msg2);
193         }
194         if (mais){
195             menu = 1;
196             delay(delaytime);
197             lcd.clear();
198         }
199         else if (menos){
200             menu = 7;
201             delay(delaytime);
202             lcd.clear();
203         }
204         else if (confirma){
205             funcionamento = funcionamento xor 1;
206             delay(delaytime);
207             lcd.clear();
208         }
209         break;
210     case 1:
211         lcd.setCursor(0,0);
212         lcd.print("CONTROLE ACESSO");
213         if (acesso==0){
214             lcd.setCursor(0,1);
215             lcd.print(msg1);
216         }
217         else{
218             lcd.setCursor(0,1);
219             lcd.print(msg2);
220         }
221         if (mais){
222             menu = 2;
223             delay(delaytime);
224             lcd.clear();
225         }
226         else if (menos){
227             menu = 0;
228             delay(delaytime);
229             lcd.clear();
230         }
231         else if (confirma){
232             acesso = acesso xor 1;
233             delay(delaytime);
234             lcd.clear();
235         }
236         break;
237     case 2:
238         lcd.setCursor(0,0);
239         lcd.print("CONTROLE TEMPE");
240         if (temperatura==0){
241             lcd.setCursor(0,1);
242             lcd.print(msg1);
243         }
244         else{
245             lcd.setCursor(0,1);

```

```

246         lcd.print(msg2);
247     }
248     if (mais){
249         menu = 3;
250         delay(delaytime);
251         lcd.clear();
252     }
253     else if (menos){
254         menu = 1;
255         delay(delaytime);
256         lcd.clear();
257     }
258     else if (confirma){
259         temperatura = temperatura xor 1;
260         delay(delaytime);
261         lcd.clear();
262     }
263     break;
264     case 3:
265         lcd.setCursor(0,0);
266         lcd.print("TEMPERATURA MAX");
267         lcd.setCursor(0,1);
268         lcd.print(temperaturaLimite);
269         if (mais){
270             menu = 5;
271             delay(delaytime);
272             lcd.clear();
273         }
274         else if (menos){
275             menu = 2;
276             delay(delaytime);
277             lcd.clear();
278         }
279         else if (confirma){
280             menu = 4;
281             delay(delaytime);
282             lcd.clear();
283         }
284         break;
285     case 4:
286         lcd.setCursor(0,0);
287         lcd.print("ALTERAR TEMP MAX");
288         lcd.setCursor(0,1);
289         lcd.print(temperaturaLimite);
290         if (mais&&(temperaturaLimite<=100)){
291             temperaturaLimite = temperaturaLimite + 1;
292             delay(delaytime);
293             lcd.clear();
294         }
295         else if (menos&&(temperaturaLimite>=15)){
296             temperaturaLimite = temperaturaLimite - 1;
297             delay(delaytime);
298             lcd.clear();
299         }
300         else if (confirma){
301             menu = 3;
302             delay(delaytime);
303             lcd.clear();
304         }
305         break;
306     case 5:
307         lcd.setCursor(0,0);
308         lcd.print("MONITOR PRESENCA");
309         if (presenca==0){

```

```

310         lcd.setCursor(0,1);
311         lcd.print(msg1);
312     }
313     else{
314         lcd.setCursor(0,1);
315         lcd.print(msg2);
316     }
317     if (mais){
318         menu = 6;
319         delay(delaytime);
320         lcd.clear();
321     }
322     else if (menos){
323         menu = 3;
324         delay(delaytime);
325         lcd.clear();
326     }
327     else if (confirma){
328         presenca = presenca xor 1;
329         delay(delaytime);
330         lcd.clear();
331     }
332     break;
333     case 6:
334         lcd.setCursor(0,0);
335         lcd.print("CONTROLE LUMUS");
336         if (luminosidade==0){
337             lcd.setCursor(0,1);
338             lcd.print(msg1);
339         }
340         else{
341             lcd.setCursor(0,1);
342             lcd.print(msg2);
343         }
344         if (mais){
345             menu = 7;
346             delay(delaytime);
347             lcd.clear();
348         }
349         else if (menos){
350             menu = 5;
351             delay(delaytime);
352             lcd.clear();
353         }
354         else if (confirma){
355             luminosidade = luminosidade xor 1;
356             delay(delaytime);
357             lcd.clear();
358         }
359         break;
360     case 7:
361         lcd.setCursor(0,0);
362         lcd.print("ALARME");
363         if (alarme==0){
364             lcd.setCursor(0,1);
365             lcd.print(msg1);
366         }
367         else{
368             lcd.setCursor(0,1);
369             lcd.print(msg2);
370         }
371         if (mais){
372             menu = 0;
373             delay(delaytime);

```

```

374     lcd.clear();
375 }
376 else if(menos){
377     menu = 6;
378     delay(delaytime);
379     lcd.clear();
380 }
381 else if(confirma){
382     alarme = alarme xor 1;
383     delay(delaytime);
384     lcd.clear();
385 }
386 break;
387 case 8:
388     lcd.setCursor(0,0);
389     lcd.print("ALARME ACIONADO");
390     lcd.setCursor(0,1);
391     lcd.print("F");
392     lcd.setCursor(1,1);
393     lcd.print(funcionamento);
394     lcd.setCursor(7,1);
395     lcd.print("A");
396     lcd.setCursor(8,1);
397     lcd.print(alarme);
398     lcd.setCursor(14,1);
399     lcd.print("P");
400     lcd.setCursor(15,1);
401     lcd.print(sensorPresenca);
402     alerta(200);
403     menu = 0;
404     delay(delaytime);
405     lcd.clear();
406     break;
407 case 9:
408     lcd.setCursor(0,0);
409     lcd.print("PROBLEMA TEMP");
410     lcd.print("F");
411     lcd.setCursor(1,1);
412     lcd.print(funcionamento);
413     lcd.setCursor(10,1);
414     lcd.print("T");
415     lcd.setCursor(11,1);
416     lcd.print(temperaturaAtual);
417     alerta(400);
418     menu = 0;
419     delay(delaytime);
420     lcd.clear();
421     break;
422 case 10:
423     lcd.setCursor(0,0);
424     lcd.print("LUZ ACESA");
425     lcd.print("F");
426     lcd.setCursor(1,1);
427     lcd.print(funcionamento);
428     lcd.setCursor(7,1);
429     lcd.print("L");
430     lcd.setCursor(8,1);
431     lcd.print(sensorLuz);
432     lcd.setCursor(14,1);
433     lcd.print("P");
434     lcd.setCursor(15,1);
435     lcd.print(sensorPresenca);
436     lcd.print(presenca);
437     alerta(500);

```

```

438     menu = 0;
439     delay(delaytime);
440     lcd.clear();
441     break;
442 }
443 }

```

3.2 Código do módulo teclado

```

1  #include <Arduino.h>
2  #include <Wire.h>
3
4  const byte SLAVE_ADDRESS = 42;
5  char senha_digitada[4];
6  char senha0[4];
7  char senha1[4];
8  char senha2[4];
9  char senha3[4];
10 char senha4[4];
11 char senha5[4];
12 char senha6[4];
13 char senha7[4];
14 int button_zero;
15 int button_um;
16 int button_dois;
17 int button_tres;
18 int button_quatro;
19 int button_cinco;
20 int button_seis;
21 int button_sete;
22 int button_oito;
23 int button_nove;
24 int button_enviar;
25 int button_limpar;
26 int cont = 0;
27 int cont2 = 0;
28 int contSenha = 1;
29 int salvar = 0;
30 int delaytime=150;
31 int blink = 0;
32 int valor = 12;
33
34 void luzTeclado(int num){
35     if(num == 0){
36         digitalWrite(A0, LOW);
37         digitalWrite(A1, LOW);
38         digitalWrite(A2, LOW);
39         digitalWrite(A3, LOW);
40     }
41     else if(num == 1){
42         digitalWrite(A0, LOW);
43         digitalWrite(A1, LOW);
44         digitalWrite(A2, LOW);
45         digitalWrite(A3, HIGH);
46     }
47     else if(num == 2){
48         digitalWrite(A0, LOW);
49         digitalWrite(A1, LOW);
50         digitalWrite(A2, HIGH);
51         digitalWrite(A3, LOW);
52     }
53     else if(num == 3){
54         digitalWrite(A0, LOW);

```

```
55     digitalWrite(A1, LOW);
56     digitalWrite(A2, HIGH);
57     digitalWrite(A3, HIGH);
58 }
59 else if (num == 4) {
60     digitalWrite(A0, LOW);
61     digitalWrite(A1, HIGH);
62     digitalWrite(A2, LOW);
63     digitalWrite(A3, LOW);
64 }
65 else if (num == 5) {
66     digitalWrite(A0, LOW);
67     digitalWrite(A1, HIGH);
68     digitalWrite(A2, LOW);
69     digitalWrite(A3, HIGH);
70 }
71 else if (num == 6) {
72     digitalWrite(A0, LOW);
73     digitalWrite(A1, HIGH);
74     digitalWrite(A2, HIGH);
75     digitalWrite(A3, LOW);
76 }
77 else if (num == 7) {
78     digitalWrite(A0, LOW);
79     digitalWrite(A1, HIGH);
80     digitalWrite(A2, HIGH);
81     digitalWrite(A3, HIGH);
82 }
83 else if (num == 8) {
84     digitalWrite(A0, HIGH);
85     digitalWrite(A1, LOW);
86     digitalWrite(A2, LOW);
87     digitalWrite(A3, LOW);
88 }
89 else if (num == 9) {
90     digitalWrite(A0, HIGH);
91     digitalWrite(A1, LOW);
92     digitalWrite(A2, LOW);
93     digitalWrite(A3, HIGH);
94 }
95 else if (num == 10) {
96     digitalWrite(A0, HIGH);
97     digitalWrite(A1, LOW);
98     digitalWrite(A2, HIGH);
99     digitalWrite(A3, LOW);
100 }
101 else if (num == 11) {
102     digitalWrite(A0, HIGH);
103     digitalWrite(A1, LOW);
104     digitalWrite(A2, HIGH);
105     digitalWrite(A3, HIGH);
106 }
107 else if (num == 12) {
108     digitalWrite(A0, HIGH);
109     digitalWrite(A1, HIGH);
110     digitalWrite(A2, LOW);
111     digitalWrite(A3, LOW);
112 }
113 else if (num == 13) {
114     digitalWrite(A0, HIGH);
115     digitalWrite(A1, HIGH);
116     digitalWrite(A2, LOW);
117     digitalWrite(A3, HIGH);
118 }
```

```
119     else if (num == 14){
120         digitalWrite(A0, HIGH);
121         digitalWrite(A1, HIGH);
122         digitalWrite(A2, HIGH);
123         digitalWrite(A3, LOW);
124     }
125     else if (num == 15){
126         digitalWrite(A0, HIGH);
127         digitalWrite(A1, HIGH);
128         digitalWrite(A2, HIGH);
129         digitalWrite(A3, HIGH);
130     }
131 }
132
133 void setup() {
134     pinMode(2, INPUT);
135     pinMode(3, INPUT);
136     pinMode(4, INPUT);
137     pinMode(5, INPUT);
138     pinMode(6, INPUT);
139     pinMode(7, INPUT);
140     pinMode(8, INPUT);
141     pinMode(9, INPUT);
142     pinMode(10, INPUT);
143     pinMode(11, INPUT);
144     pinMode(12, INPUT);
145     pinMode(13, INPUT);
146     pinMode(A0, OUTPUT);
147     pinMode(A1, OUTPUT);
148     pinMode(A2, OUTPUT);
149     pinMode(A3, OUTPUT);
150     senha0[0] = '1';
151     senha0[0] = '1';
152     senha0[0] = '1';
153     senha0[0] = '1';
154     Wire.begin();
155 }
156
157 void loop() {
158     button_zero = digitalRead(2);
159     button_um = digitalRead(3);
160     button_dois = digitalRead(4);
161     button_tres = digitalRead(5);
162     button_quatro = digitalRead(6);
163     button_cinco = digitalRead(7);
164     button_seis = digitalRead(8);
165     button_sete = digitalRead(9);
166     button_oito = digitalRead(10);
167     button_nove = digitalRead(11);
168     button_enviar = digitalRead(12);
169     button_limpar = digitalRead(13);
170
171     if (button_zero){
172         valor = 0;
173     }
174     else if (button_um){
175         valor = 1;
176     }
177     else if (button_dois){
178         valor = 2;
179     }
180     else if (button_tres){
181         valor = 3;
182     }
```

```
183     else if (button_quatro){
184         valor = 4;
185     }
186     else if (button_cinco){
187         valor = 5;
188     }
189     else if (button_seis){
190         valor = 6;
191     }
192     else if (button_sete){
193         valor = 7;
194     }
195     else if (button_oito){
196         valor = 8;
197     }
198     else if (button_nove){
199         valor = 9;
200     }
201     else if (button_limpar){
202         valor = 10;
203     }
204     else if (button_enviar){
205         valor = 11;
206     }
207     else{
208         valor = 12;
209     }
210
211     switch (cont){
212         case 0:
213             if (salvar==0){
214                 luzTeclado(15);
215             }
216             else if (salvar == 1){
217                 luzTeclado(7);
218             }
219             switch (valor){
220                 case 0:
221                     senha_digitada[cont] = '0';
222                     cont = cont + 1;
223                     delay(delaytime);
224                     break;
225                 case 1:
226                     senha_digitada[cont] = '1';
227                     cont = cont + 1;
228                     delay(delaytime);
229                     break;
230                 case 2:
231                     senha_digitada[cont] = '2';
232                     cont = cont + 1;
233                     delay(delaytime);
234                     break;
235                 case 3:
236                     senha_digitada[cont] = '3';
237                     cont = cont + 1;
238                     delay(delaytime);
239                     break;
240                 case 4:
241                     senha_digitada[cont] = '4';
242                     cont = cont + 1;
243                     delay(delaytime);
244                     break;
245                 case 5:
246                     senha_digitada[cont] = '5';
```



```

247         cont = cont + 1;
248         delay(delaytime);
249     break;
250     case 6:
251         senha_digitada[cont] = '6';
252         cont = cont + 1;
253         delay(delaytime);
254     break;
255     case 7:
256         senha_digitada[cont] = '7';
257         cont = cont + 1;
258         delay(delaytime);
259     break;
260     case 8:
261         senha_digitada[cont] = '8';
262         cont = cont + 1;
263         delay(delaytime);
264     break;
265     case 9:
266         senha_digitada[cont] = '9';
267         cont = cont + 1;
268         delay(delaytime);
269     break;
270     case 11:
271         cont2 = cont2 + 1;
272         if (cont2 == 5){
273             salvar = 1;
274         }
275         else if (cont2 == 10){
276             salvar = 0;
277             cont2 = 0;
278         }
279         delay(delaytime);
280     break;
281 }
282 break;
283 case 1:
284     if (blink == 1){
285         blink = 0;
286         luzTeclado(0);
287     }
288     else if (blink == 0 ){
289         blink = 1;
290         if (salvar == 0){
291             luzTeclado(12);
292         }
293         else if (salvar == 1){
294             luzTeclado(4);
295         }
296     }
297     switch (valor){
298     case 0:
299         senha_digitada[cont] = '0';
300         cont = cont + 1;
301         delay(delaytime);
302     break;
303     case 1:
304         senha_digitada[cont] = '1';
305         cont = cont + 1;
306         delay(delaytime);
307     break;
308     case 2:
309         senha_digitada[cont] = '2';
310         cont = cont + 1;

```

```

311         delay(delaytime);
312     break;
313     case 3:
314         senha_digitada[cont] = '3';
315         cont = cont + 1;
316         delay(delaytime);
317     break;
318     case 4:
319         senha_digitada[cont] = '4';
320         cont = cont + 1;
321         delay(delaytime);
322     break;
323     case 5:
324         senha_digitada[cont] = '5';
325         cont = cont + 1;
326         delay(delaytime);
327     break;
328     case 6:
329         senha_digitada[cont] = '6';
330         cont = cont + 1;
331         delay(delaytime);
332     break;
333     case 7:
334         senha_digitada[cont] = '7';
335         cont = cont + 1;
336         delay(delaytime);
337     break;
338     case 8:
339         senha_digitada[cont] = '8';
340         cont = cont + 1;
341         delay(delaytime);
342     break;
343     case 9:
344         senha_digitada[cont] = '9';
345         cont = cont + 1;
346         delay(delaytime);
347     break;
348     case 10:
349         cont = cont - 1;
350         delay(delaytime);
351     break;
352 }
353 break;
354 case 2:
355     if(blink == 1){
356         blink = 0;
357         luzTeclado(0);
358     }
359     else if(blink == 0 ){
360         blink = 1;
361         if(salvar == 0){
362             luzTeclado(13);
363         }
364         else if(salvar == 1){
365             luzTeclado(5);
366         }
367     }
368     switch (valor){
369         case 0:
370             senha_digitada[cont] = '0';
371             cont = cont + 1;
372             delay(delaytime);
373         break;
374         case 1:

```

```

375         senha_digitada[cont] = '1';
376         cont = cont + 1;
377         delay(delaytime);
378     break;
379     case 2:
380         senha_digitada[cont] = '2';
381         cont = cont + 1;
382         delay(delaytime);
383     break;
384     case 3:
385         senha_digitada[cont] = '3';
386         cont = cont + 1;
387         delay(delaytime);
388     break;
389     case 4:
390         senha_digitada[cont] = '4';
391         cont = cont + 1;
392         delay(delaytime);
393     break;
394     case 5:
395         senha_digitada[cont] = '5';
396         cont = cont + 1;
397         delay(delaytime);
398     break;
399     case 6:
400         senha_digitada[cont] = '6';
401         cont = cont + 1;
402         delay(delaytime);
403     break;
404     case 7:
405         senha_digitada[cont] = '7';
406         cont = cont + 1;
407         delay(delaytime);
408     break;
409     case 8:
410         senha_digitada[cont] = '8';
411         cont = cont + 1;
412         delay(delaytime);
413     break;
414     case 9:
415         senha_digitada[cont] = '9';
416         cont = cont + 1;
417         delay(delaytime);
418     break;
419     case 10:
420         cont = cont - 1;
421         delay(delaytime);
422     break;
423 }
424 break;
425 case 3:
426     if(blink == 1){
427         blink = 0;
428         luzTeclado(0);
429     }
430     else if(blink == 0 ){
431         blink = 1;
432         if(salvar == 0){
433             luzTeclado(14);
434         }
435         else if(salvar == 1){
436             luzTeclado(6);
437         }
438     }

```

```

439     switch (valor){
440         case 0:
441             senha_digitada[cont] = '0';
442             cont = cont + 1;
443             delay(delaytime);
444         break;
445         case 1:
446             senha_digitada[cont] = '1';
447             cont = cont + 1;
448             delay(delaytime);
449         break;
450         case 2:
451             senha_digitada[cont] = '2';
452             cont = cont + 1;
453             delay(delaytime);
454         break;
455         case 3:
456             senha_digitada[cont] = '3';
457             cont = cont + 1;
458             delay(delaytime);
459         break;
460         case 4:
461             senha_digitada[cont] = '4';
462             cont = cont + 1;
463             delay(delaytime);
464         break;
465         case 5:
466             senha_digitada[cont] = '5';
467             cont = cont + 1;
468             delay(delaytime);
469         break;
470         case 6:
471             senha_digitada[cont] = '6';
472             cont = cont + 1;
473             delay(delaytime);
474         break;
475         case 7:
476             senha_digitada[cont] = '7';
477             cont = cont + 1;
478             delay(delaytime);
479         break;
480         case 8:
481             senha_digitada[cont] = '8';
482             cont = cont + 1;
483             delay(delaytime);
484         break;
485         case 9:
486             senha_digitada[cont] = '9';
487             cont = cont + 1;
488             delay(delaytime);
489         break;
490         case 10:
491             cont = cont - 1;
492             delay(delaytime);
493         break;
494     }
495     break;
496     case 4:
497         if(blink == 1){
498             blink = 0;
499             luzTeclado(0);
500         }
501         else if(blink == 0 ){
502             blink = 1;

```

```

503     if(salvar == 0){
504         luzTeclado(15);
505     }
506     else if(salvar == 1){
507         luzTeclado(7);
508     }
509 }
510 switch (valor){
511     case 10:
512         cont = cont - 1;
513         delay(delaytime);
514         break;
515     case 11:
516         if (salvar==1){
517             switch (contSenha){
518                 case 0:
519                     senha0[0] = senha_digitada[0];
520                     senha0[1] = senha_digitada[1];
521                     senha0[2] = senha_digitada[2];
522                     senha0[3] = senha_digitada[3];
523                     contSenha = contSenha + 1;
524                     break;
525                 case 1:
526                     senha1[0] = senha_digitada[0];
527                     senha1[1] = senha_digitada[1];
528                     senha1[2] = senha_digitada[2];
529                     senha1[3] = senha_digitada[3];
530                     contSenha = contSenha + 1;
531                     break;
532                 case 2:
533                     senha2[0] = senha_digitada[0];
534                     senha2[1] = senha_digitada[1];
535                     senha2[2] = senha_digitada[2];
536                     senha2[3] = senha_digitada[3];
537                     contSenha = contSenha + 1;
538                     break;
539                 case 3:
540                     senha3[0] = senha_digitada[0];
541                     senha3[1] = senha_digitada[1];
542                     senha3[2] = senha_digitada[2];
543                     senha3[3] = senha_digitada[3];
544                     contSenha = contSenha + 1;
545                     break;
546                 case 4:
547                     senha4[0] = senha_digitada[0];
548                     senha4[1] = senha_digitada[1];
549                     senha4[2] = senha_digitada[2];
550                     senha4[3] = senha_digitada[3];
551                     contSenha = contSenha + 1;
552                     break;
553                 case 5:
554                     senha5[0] = senha_digitada[0];
555                     senha5[1] = senha_digitada[1];
556                     senha5[2] = senha_digitada[2];
557                     senha5[3] = senha_digitada[3];
558                     contSenha = contSenha + 1;
559                     break;
560                 case 6:
561                     senha6[0] = senha_digitada[0];
562                     senha6[1] = senha_digitada[1];
563                     senha6[2] = senha_digitada[2];
564                     senha6[3] = senha_digitada[3];
565                     contSenha = contSenha + 1;
566                     break;

```

```

567         case 7:
568             senha7[0] = senha_digitada[0];
569             senha7[1] = senha_digitada[1];
570             senha7[2] = senha_digitada[2];
571             senha7[3] = senha_digitada[3];
572             contSenha = 0;
573         break;
574     }
575 }
576 else{
577     if ((senha0[0]==senha_digitada[0])&&(senha0[1]==
senha_digitada[1])&&(senha0[2]==senha_digitada[2])&&(senha0[3]==
senha_digitada[3])){
578         Wire.beginTransaction (SLAVE_ADDRESS);
579         Wire.write ('1');
580         Wire.endTransmission ();
581         delay(500);
582         Wire.beginTransaction (SLAVE_ADDRESS);
583         Wire.write ('0');
584         Wire.endTransmission ();
585     }
586     else if ((senha1[0]==senha_digitada[0])&&(senha1[1]==
senha_digitada[1])&&(senha1[2]==senha_digitada[2])&&(senha1[3]==
senha_digitada[3])){
587         Wire.beginTransaction (SLAVE_ADDRESS);
588         Wire.write ('1');
589         Wire.endTransmission ();
590         delay(500);
591         Wire.beginTransaction (SLAVE_ADDRESS);
592         Wire.write ('0');
593         Wire.endTransmission ();
594     }
595     else if ((senha2[0]==senha_digitada[0])&&(senha2[1]==
senha_digitada[1])&&(senha2[2]==senha_digitada[2])&&(senha2[3]==
senha_digitada[3])){
596         Wire.beginTransaction (SLAVE_ADDRESS);
597         Wire.write ('1');
598         Wire.endTransmission ();
599         delay(500);
600         Wire.beginTransaction (SLAVE_ADDRESS);
601         Wire.write ('0');
602         Wire.endTransmission ();
603     }
604     else if ((senha3[0]==senha_digitada[0])&&(senha3[1]==
senha_digitada[1])&&(senha3[2]==senha_digitada[2])&&(senha3[3]==
senha_digitada[3])){
605         Wire.beginTransaction (SLAVE_ADDRESS);
606         Wire.write ('1');
607         Wire.endTransmission ();
608         delay(500);
609         Wire.beginTransaction (SLAVE_ADDRESS);
610         Wire.write ('0');
611         Wire.endTransmission ();
612     }
613     else if ((senha4[0]==senha_digitada[0])&&(senha4[1]==
senha_digitada[1])&&(senha4[2]==senha_digitada[2])&&(senha4[3]==
senha_digitada[3])){
614         Wire.beginTransaction (SLAVE_ADDRESS);
615         Wire.write ('1');
616         Wire.endTransmission ();
617         delay(500);
618         Wire.beginTransaction (SLAVE_ADDRESS);
619         Wire.write ('0');
620         Wire.endTransmission ();

```

```

621     }
622     else if ((senha5[0]==senha_digitada[0]) &&(senha5[1]==
senha_digitada[1]) &&(senha5[2]==senha_digitada[2]) &&(senha5[3]==
senha_digitada[3])){
623         Wire.beginTransaction (SLAVE_ADDRESS);
624         Wire.write ('1');
625         Wire.endTransmission ();
626         delay(500);
627         Wire.beginTransaction (SLAVE_ADDRESS);
628         Wire.write ('0');
629         Wire.endTransmission ();
630     }
631     else if ((senha6[0]==senha_digitada[0]) &&(senha6[1]==
senha_digitada[1]) &&(senha6[2]==senha_digitada[2]) &&(senha6[3]==
senha_digitada[3])){
632         Wire.beginTransaction (SLAVE_ADDRESS);
633         Wire.write ('1');
634         Wire.endTransmission ();
635         delay(500);
636         Wire.beginTransaction (SLAVE_ADDRESS);
637         Wire.write ('0');
638         Wire.endTransmission ();
639     }
640     else if ((senha7[0]==senha_digitada[0]) &&(senha7[1]==
senha_digitada[1]) &&(senha7[2]==senha_digitada[2]) &&(senha7[3]==
senha_digitada[3])){
641         Wire.beginTransaction (SLAVE_ADDRESS);
642         Wire.write ('1');
643         Wire.endTransmission ();
644         delay(500);
645         Wire.beginTransaction (SLAVE_ADDRESS);
646         Wire.write ('0');
647         Wire.endTransmission ();
648     }
649     else{
650         Wire.beginTransaction (SLAVE_ADDRESS);
651         Wire.write ('0');
652         Wire.endTransmission ();
653     }
654 }
655 delay(delaytime);
656 break;
657 }
658 break;
659 }
660 }

```