

```
1  #include "array_store.h"
2
3  uint32_t password_uid[32][2];
4  int length_arr;
5
6  void store_rfid(void)
7  {
8      uint32_t uid;
9      int check;
10     eleventh_2_LCD();
11     int z = get_password();
12     if (z == 0x00987654)
13     {
14         uid = rfid_get_data();
15         check = check_id(uid);
16     }
17     else
18     {
19         eighteenth_2_LCD();
20         PWM_statel();
21         delay_ms(1000);
22         PWM_state0();
23         return;
24     }
25     while (1)
26     {
27         if (uid == 0x00000000)
28         {
29             fourth_2_LCD();
30             delay_ms(1000);
31             return;
32         }
33         else if (check)
34         {
35             third_2_LCD();
36             delay_ms(1000);
37             seventh_2_LCD();
38             int x;
39             while (1)
40             {
41                 x = read_key();
42                 if (x == 0x31 || x == 0x32)
43                     break;
44                 delay_ms(300);
45             }
46             if (x == 0x31)
47             {
48                 int i = 0;
49                 for (i = 0; i < 32; i++)
50                 {
51                     if (password_uid[i][1] == uid)
52                         break;
53                 }
54                 password_uid[i][0] = 0x00000000;
55                 password_uid[i][1] = 0x00000000;
56                 for (int d = i; d < 31; d++)
57                 {
58                     password_uid[d][0] = password_uid[d+1][0];
59                     password_uid[d][1] = password_uid[d+1][1];
60                 }
61                 length_arr--;
62                 return;
63             }
64             else if (x == 0x32)
65             {
66                 return;
67             }
68         }
69         else if (!check)
70             break;
71     }
72     fifth_2_LCD();
73     send_full(uid);
```

```
74     delay_ms(1000);
75     password_uid[length_arr][1] = uid;
76     int x, y;
77     while (1)
78     {
79         CMD_2_LCD(LCD_CLR);
80         sixth_2_LCD();
81         y = get_password();
82         CMD_2_LCD(LCD_CLR);
83         eighth_2_LCD();
84         x = get_password();
85         CMD_2_LCD(LCD_CLR);
86         if (x == y)
87             break;
88         ninth_2_LCD();
89         delay_ms(1000);
90     }
91     tenth_2_LCD();
92     password_uid[length_arr][0] = x;
93     length_arr++;
94 }
95 bool check_id(uint32_t uid)
96 {
97     for (int i = 0; i < 32; i++)
98     {
99         if (password_uid[i][1] == uid)
100             return true;
101     }
102     return false;
103 }
104 void read_user(void)
105 {
106     uint32_t uid = rfid_get_data();
107     int i = 0;
108     bool flag = false;
109     for (i = 0; i < 32; i++)
110     {
111         if (password_uid[i][1] == uid)
112         {
113             flag = false;
114             break;
115         }
116         else
117             flag = true;
118     }
119     if (uid == 0x00000000)
120     {
121         fourth_2_LCD();
122         delay_ms(1000);
123         return;
124     }
125     if (flag)
126         twelvth_2_LCD();
127     else
128     {
129         fifth_2_LCD();
130         send_full(uid);
131         delay_ms(500);
132         CMD_2_LCD(LCD_CLR);
133         sixth_2_LCD();
134
135         int y = get_password();
136         CMD_2_LCD(LCD_CLR);
137         if (y == password_uid[i][0])
138         {
139             thirteenth_2_LCD();
140             set_time_out(1000000);
141             bool open;
142             while(!timed_out())
143             {
144                 open = ADC1_read(0x1);
145                 if (open)
```

```
147         break;
148     }
149     if (open)
150     {
151         sixteenth_2_LCD();
152         delay_ms(500);
153         set_time_out(400000);
154         PWM_state2();
155         while(!timed_out())
156         {
157         }
158         PWM_state0();
159     }
160     else if (!open)
161     {
162         fifteenth_2_LCD();
163     }
164 }
165 else
166 {
167     fourteenth_2_LCD();
168     PWM_state1();
169     delay_ms(1000);
170     PWM_state0();
171 }
172 }
173 delay_ms(1000);
174 }
175 void storing(void)
176 {
177     page_program(*password_uid, length_arr);
178 }
179 void read_flash(void)
180 {
181     uint32_t array[64];
182     page_read(array);
183     for (int i = 0; i < 32; i++)
184     {
185         for (int j = 0; j < 2; j++)
186         {
187             password_uid[i][j] = array[(2*i) + j];
188         }
189     }
190     int x;
191     for (x = 0; x < 32; x++)
192     {
193         if (password_uid[x][1] == 0xFFFFFFFF)
194             break;
195     }
196     length_arr = x;
197 }
198 void user(void)
199 {
200     seventeenth_2_LCD();
201     Data_2_LCD(length_arr + 0x30);
202 }
```