

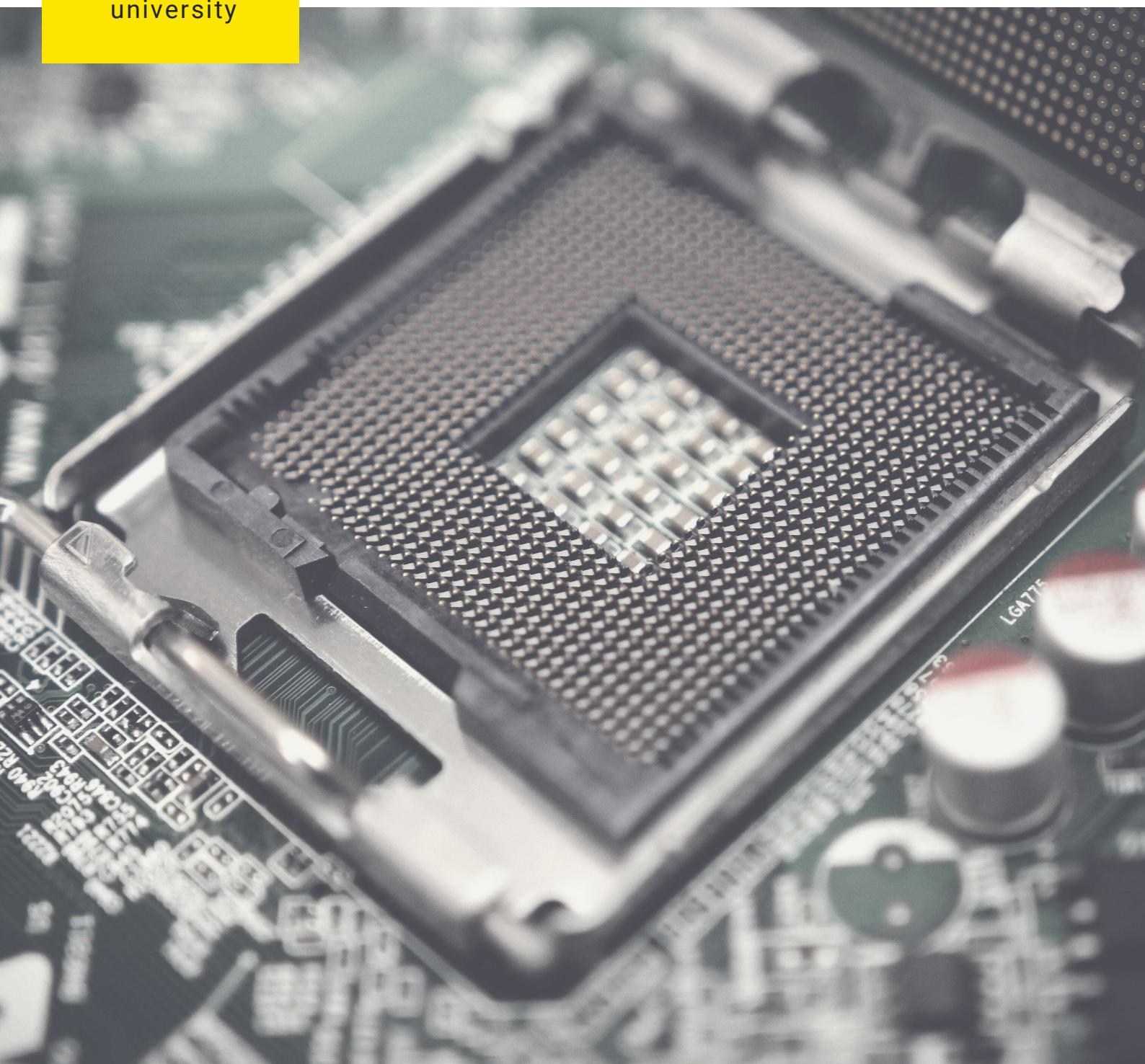


Microprocessors

PROJECT REPORT

Dr. Hossam Mostafa

ALEXANDRIA
university



TEAM 47

NAME

ID NO

AHMED HASSAN ABD-ELQADER ABDO	14
AHMED SAMIR MOHAMMED ARGAWY	19
IBRAHIM SOBHY IBRAHIM MOHAMMED	10
OMAR ADEL OMAR FITOUH	137
YAHYA HOSSAM ALI MOHAMMED	273

ABSTRACT OF THE PROJECT

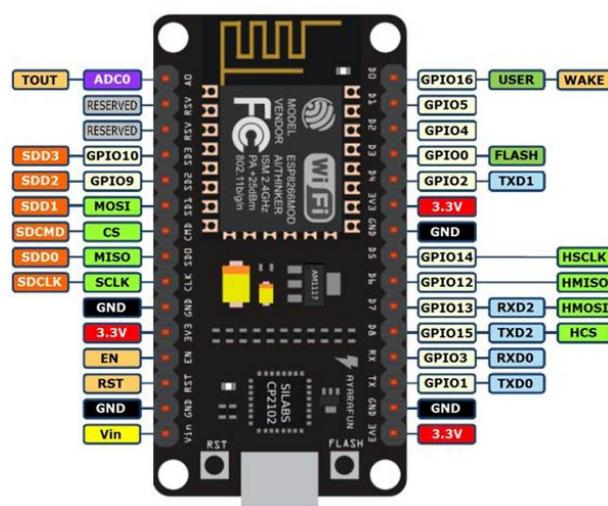
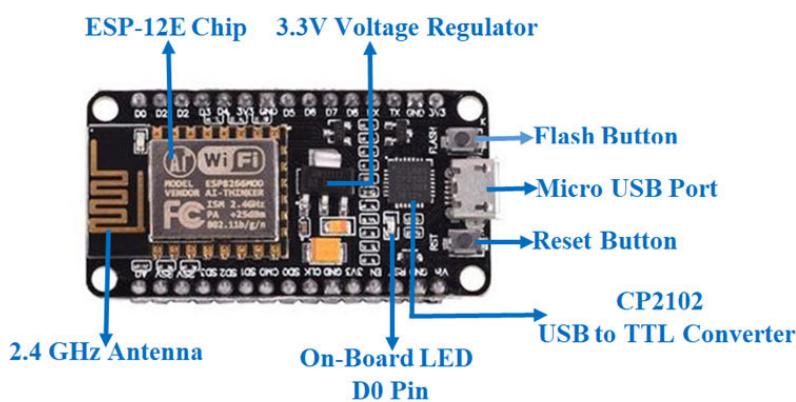
In this project we will use esp8266 with Micropython to display 1-digit number on 7-segment screen then increase , decrease and reset this number by 2 methods:

- Circuit in breadboard with 3 push buttons increase, decrease and reset connected to the module which we programmed using thonny program.
 - Using website by connecting our programmed module with Wi-Fi

SHORT DESCRIPTION

Brief About NodeMCU ESP8266:

The NodeMCU ESP8266 development board comes with the ESP-12E module containing ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjustable clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects. NodeMCU can be powered using Micro USB jack



Brief About MicroPython:

MicroPython is a tiny open source Python programming language interpreter that runs on small embedded development boards. With MicroPython you can write clean and simple Python code to control hardware instead of having to use complex low-level languages like C or C++.

HARDWARE DESCRIPTION

Materials :

- Breadboard
- Resistors
- Push buttons
- 7-segment
- Wires
- ESP8266

Connections:

- 7-segment

-Connections with module :

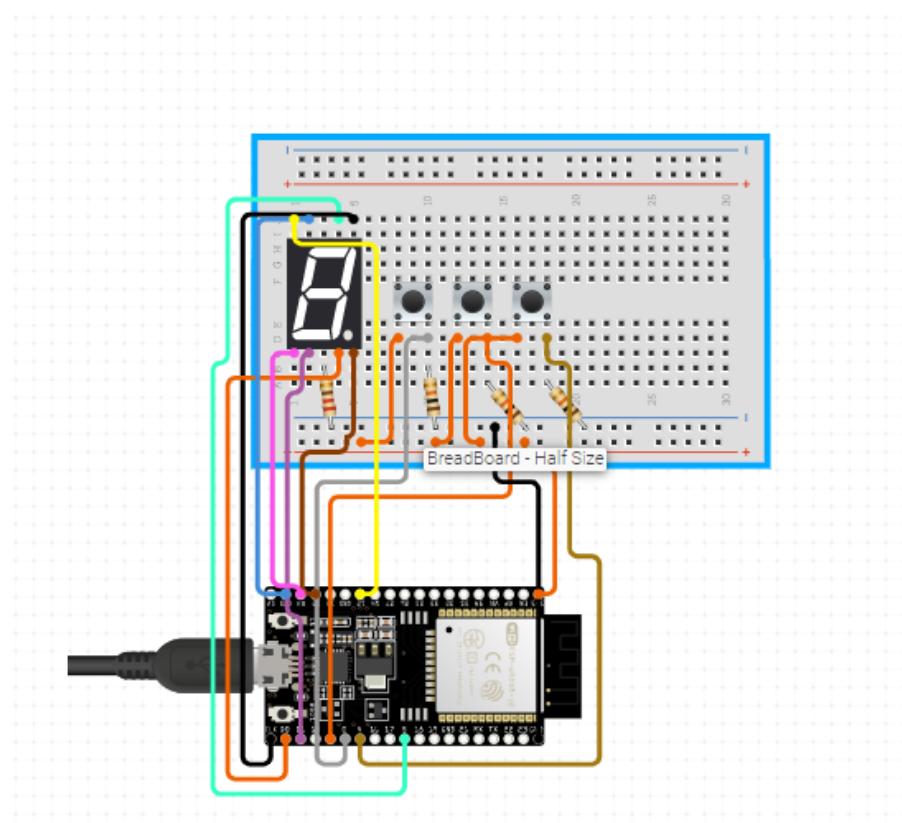
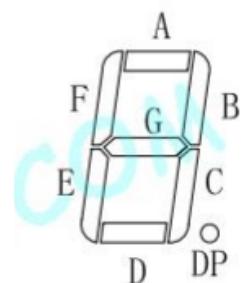
A connected to Pin 16 D connected to Pin 14
B connected to Pin 5 E connected to Pin 12
C connected to Pin 4 F connected to Pin 13
G connected to Pin 15

-The common pin connected to GND with resistor

- Buttons

-Inc button connected to pin 0 In the module and the other terminal connected to GND with resistor

-dec button connected to pin 2 In the module and the other terminal connected to GND with resistor



SOFTWARE DESCRIPTION

We first installed thonny then connect the module to pc

Then using thonny We flash the firmware in ESP2866 then Start Coding:

- with 2 push buttons increase, decrease

```
1 from machine import Pin  
2 import time  
3  
5 #####  
6 #####function  
7 #####  
8 def segment7(n):  
9     if n == 0:  
10         a.value(1)  
11         b.value(1)  
12         c.value(1)  
13         d.value(1)  
14         e.value(1)|  
15         f.value(1)  
16         g.value(0)  
17     elif n == 1:  
18         a.value(0)  
19         b.value(1)  
20         c.value(1)  
21         d.value(0)  
22         e.value(0)  
23         f.value(0)  
24         g.value(0)  
25  
26     elif n == 2:  
27         a.value(1)  
28         b.value(1)  
29         c.value(0)  
30         d.value(1)  
31         e.value(1)  
32         f.value(0)  
33         g.value(1)
```

At the first of the code we must call the library.

This “segment7” function controls which leds at the 7-Segment will turn on to make the number “n” appears at the 7- segment

if n=0 the leds A&B&C&D&E will be turned on and the others will be turned off and that will give us 0 at the 7-Segment

if n=1 the leds B&C will be turned on and the others will be turned off and that will give us 1 at the 7-Segment

if n=2 the leds A&B&D&E&G will be turned on and the others will be turned off and that will give us 2 at the 7-Segment

```

35 elif n == 3:
36     a.value(1)
37     b.value(1)
38     c.value(1)
39     d.value(1)
40     e.value(0)
41     f.value(0)
42     g.value(1)
43
44 elif n == 4:
45     a.value(0)
46     b.value(1)
47     c.value(1)
48     d.value(0)
49     e.value(0)
50     f.value(1)
51     g.value(1)
52
53 elif n == 5:
54     a.value(1)
55     b.value(0)
56     c.value(1)
57     d.value(1)
58     e.value(0)
59     f.value(1)
60     g.value(1)
61
62 elif n == 6:
63     a.value(1)
64     b.value(0)
65     c.value(1)
66     d.value(1)
67     e.value(1)
68     f.value(1)
69     g.value(1)
70

```

if n=3 the leds A&B&C&D&G will be turned on and the others will be turned off and that will give us 3 at the 7-Segment

if n=4 the leds B&C&F&G will be turned on and the others will be turned off and that will give us 4 at the 7-Segment

if n=5 the leds A&F&C&D&G will be turned on and the others will be turned off and that will give us 5 at the 7-Segment

if n=6 the leds A&C&D&E&F&G will be turned on and the others will be turned off and that will give us 6 at the 7-Segment

```

71 elif n == 7:
72     a.value(1)
73     b.value(1)
74     c.value(1)
75     d.value(0)
76     e.value(0)
77     f.value(0)
78     g.value(0)
79
80 elif n == 8:
81     a.value(1)
82     b.value(1)
83     c.value(1)
84     d.value(1)
85     e.value(1)
86     f.value(1)
87     g.value(1)
88
89 elif n == 9:
90     a.value(1)
91     b.value(1)
92     c.value(1)
93     d.value(1)
94     e.value(0)
95     f.value(1)
96     g.value(1)
97
98 #####define |
99 #define |
100 #define |
101 #define |
102 a = Pin(16 , Pin.OUT)
103 b = Pin(5 , Pin.OUT)
104 c = Pin(4 , Pin.OUT)
105 d = Pin(14 , Pin.OUT)
106 e = Pin(12 , Pin.OUT)
107 f = Pin(13 , Pin.OUT)
108 g = Pin(15 , Pin.OUT)
109
110 inc = Pin(0 , Pin.IN)
111 dec = Pin(2 , Pin.IN)
112

```

if n=7 the leds A&B&C will be turned on and the others will be turned off and that will give us 7 at the 7-Segment

if n=8 the leds A&B&C&D&E&F&G will be turned on and the others will be turned off and that will give us 8 at the 7-Segment

if n=9 the leds A&B&C&D&F&G will be turned on and the others will be turned off and that will give us 9 at the 7-Segment

in this part we define the input and output pins. we have output pins (16,5,4,14,12,13,15) responsible for display the 7-segment that has pins (a,b,c,d,e,f,g) And input pins (0,2) are for increment and decrement buttons.

```

113 #####
114 #####assign valu
115 #####
116 num = 0
117 segment7(num)
118 #####
119 #####
120
121
122 while True:
123     if inc.value() == 0:
124         num += 1
125         if num == 10:
126             num = 0
127             segment7(num)
128             time.sleep(1)
129
130     if dec.value() == 0:
131         num -= 1
132         if num == -1:
133             num = 9
134             segment7(num)
135             time.sleep(1)
136

```

When the circuit starts to work , the 7-segment displays 0 digit.

The " inc.value" default is 1 when we press the push button it becomes 0 and increases the value of num by 1 and the same in decrement .when the value of num became 10 we make it 0 immediately . when the value of num became -1 we make it 9 immediately.
 finally we make the delay time 1 second, so when we push the button every 1 second a value will be recorded , we made this to avoid recording many values when we press the button

SOFTWARE DESCRIPTION

- code with Wi-Fi  

```

1  from machine import Pin
2  import network
3  import socket
4
5  #####
6  #####function#####
7  #####
8  -def segment7(n):
9  -    if n == 0:
10         a.value(1)
11         b.value(1)
12         c.value(1)
13         d.value(1)
14         e.value(1)
15         f.value(1)
16         g.value(0)
17     - elif n == 1:
18         a.value(0)
19         b.value(1)
20         c.value(1)
21         d.value(0)
22         e.value(0)
23         f.value(0)
24         g.value(0)
26     - elif n == 2:
27         a.value(1)
28         b.value(1)
29         c.value(0)
30         d.value(1)
31         e.value(1)
32         f.value(0)
33         g.value(1)

```

at first we should import the libraries that we need .

This “segment7” function controls which leds at the 7-Segment will turn on to make the number “n” appears at the 7-segment

if n=0 the leds A&B&C&D&E will be turned on and the others will be turned off and that will give us 0 at the 7-Segment

if n=1 the leds B&C will be turned on and the others will be turned off and that will give us 1 at the 7-Segment

if n=2 the leds A&B&D&E&G will be turned on and the others will be turned off and that will give us 2 at the 7-Segment

```

35 = elif n == 3:
36     a.value(1)
37     b.value(1)
38     c.value(1)
39     d.value(1)
40     e.value(0)
41     f.value(0)
42     g.value(1)

44 = elif n == 4:
45     a.value(0)
46     b.value(1)
47     c.value(1)
48     d.value(0)
49     e.value(0)
50     f.value(1)
51     g.value(1)

52

53 = elif n == 5:
54     a.value(1)
55     b.value(0)
56     c.value(1)
57     d.value(1)
58     e.value(0)
59     f.value(1)
60     g.value(1)

62 = elif n == 6:
63     a.value(1)
64     b.value(0)
65     c.value(1)
66     d.value(1)
67     e.value(1)
68     f.value(1)
69     g.value(1)

```

if n=3 the leds A&B&C&D&G will be turned on and the others will be turned off and that will give us 3 at the 7-Segment

if n=4 the leds B&C&F&G will be turned on and the others will be turned off and that will give us 4 at the 7-Segment

if n=5 the leds A&F&C&D&G will be turned on and the others will be turned off and that will give us 5 at the 7-Segment

if n=6 the leds A&C&D&E&F&G will be turned on and the others will be turned off and that will give us 6 at the 7-Segment

```

71 = elif n == 7:
72     a.value(1)
73     b.value(1)
74     c.value(1)
75     d.value(0)
76     e.value(0)
77     f.value(0)
78     g.value(0)
79
80 = elif n == 8:
81     a.value(1)
82     b.value(1)
83     c.value(1)
84     d.value(1)
85     e.value(1)
86     f.value(1)
87     g.value(1)
88
89 = elif n == 9:
90     a.value(1)
91     b.value(1)
92     c.value(1)
93     d.value(1)
94     e.value(0)
95     f.value(1)
96     g.value(1)
97
98 a = Pin(16, Pin.OUT)
99 b = Pin(5, Pin.OUT)
100 c = Pin(4, Pin.OUT)
101 d = Pin(14, Pin.OUT)
102 e = Pin(12, Pin.OUT)
103 f = Pin(13, Pin.OUT)
104 g = Pin(15, Pin.OUT)
105
106 num = 0
107 segment7(num)

```

if n=7 the leds A&B&C will be turned on and the others will be turned off and that will give us 7 at the 7-Segment

if n=8 the leds A&B&C&D&E&F&G will be turned on and the others will be turned off and that will give us 8 at the 7-Segment

if n=9 the leds A&B&C&D&F&G will be turned on and the others will be turned off and that will give us 9 at the 7-Segment

in this part we define the input pins. we have input pins (16,5,4,14,12,13,15) responsible for display the 7-segment that has pins (a,b,c,d,e,f,g) . When the circuit starts to work , the 7-segment displays 0 digit.

```

116 ssid = 'ESP8266-AP-WebServer'
117 password = '123456789'
118
119 ap = network.WLAN(network.AP_IF)
120 ap.active(True)
121 ap.config(essid=ssid, password=password)
122 while not ap.active():
123     pass
124 print('network config:', ap.ifconfig())
125
126 s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
127 s.bind(('0.0.0.0', 80))
128 s.listen(5)

```

we defined 2 variables to be the network name & it's password then make our esp to act as access point and activate it . we defined the network configuration like ssid & password . while function here to make sure that ap is active before executing orders then print network configurations . now we can connect to the AP but we can't open the web site so we import library socket to reach our web site . we defined a variable s has INET & STREAM that represent IPV4 & TCP which manage the operation between source and client . we bind the socket to define the IP address & port number 80 (http). if there is more than one device connected to the AP we use listen to manage executing orders from every device with maximum 5 devices .

```

134 def web_page(num):
135
136     print('Current Number is ' + str(num))
137
138     html_page = """<!DOCTYPE HTML>
139 <html>
140     <head> <meta name="viewport" content="width=device-width, initial-scale=1"> <
141     <body> <body style="background-image: linear-gradient(#373B44, #4286f4);>
142
143         <font face="Agency FB" color="white" size="10">
144             <center> <p><strong>7-SEGMENT CONTROLLER</strong></p></center> </font>
145
146         <center>
147             <form>
148                 <br><button type='submit' name='INC' value='1' style='background-color: transparent; border: none; font-size: inherit; color: inherit;'>INC</button>
149                 <br><button type='submit' name='DEC' value='1' style='background-color: transparent; border: none; font-size: inherit; color: inherit;'>DEC</button>
150                 <br><button type='submit' name='RES' value='1' style='background-color: transparent; border: none; font-size: inherit; color: inherit;'>RES</button>
151
152             <font face="Agency FB" size="4" color="white">
153                 <p>NOW DISPLAYING: <strong>"""+str(num)+"""\</strong></p> </font>
154
155             <font face="Arial" color="white" size="1">
156                 <p style="position: absolute; bottom: 5px">
157                     Powered by: Ahmed Hassan, Ahmed Hassanen, Ibrahim Mekkawy, Om
158             </p>
159         </center>
160     </body>
161 </html>"""
162     return html_page
163

```

at first we defined the web page function that interface with the variable num and display the current value of num. then we defined the page's design : number of buttons , buttons name and value , button's location , font size , background colour ,..

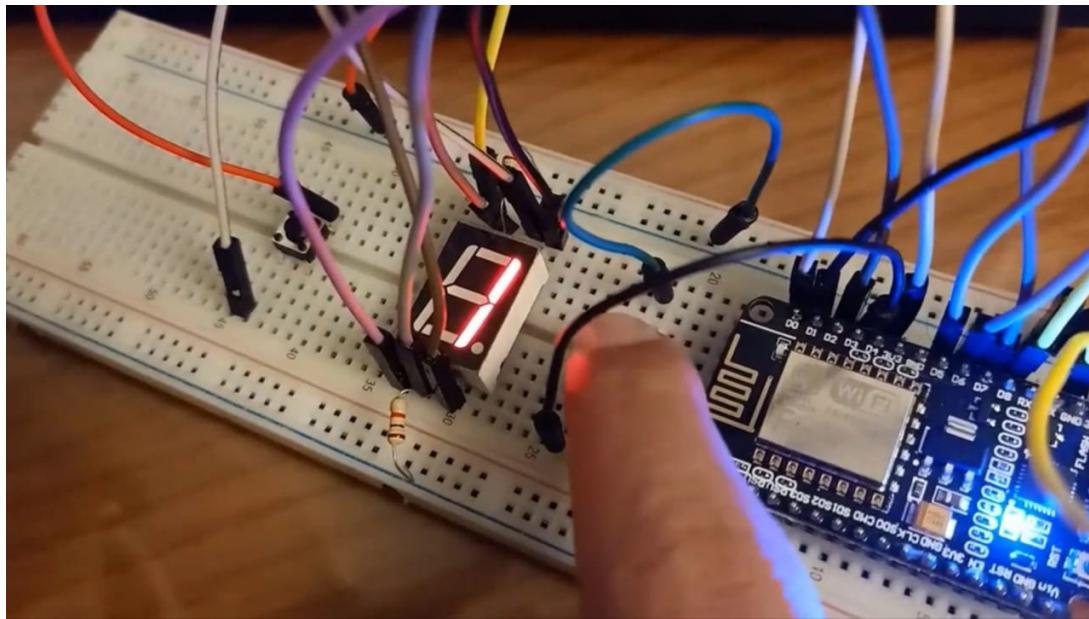
```

164 #####Run-Time code#####
165 #####
166 #####
167 #####
168 =while True:
169     conn, addr = s.accept()
170     print("Got connection from %s" % str(addr))
171
172     request=conn.recv(1024)
173     print("")
174     print("")
175     print("Content %s" % str(request))
176
177     request = str(request)
178     inc_on = request.find('/?INC=1')
179     dec_on = request.find('/?DEC=1')
180     res_on = request.find('/?RES=1')
181     if inc_on == 6:
182         print('Increament ON')
183         print(str(inc_on))
184         num += 1
185         if num == 10:
186             num = 0
187             segment7(num)
188     elif dec_on == 6:
189         print('Decrement ON')
190         print(str(dec_on))
191         num -= 1
192         if num == -1:
193             num = 9
194             segment7(num)
195     elif res_on == 6:
196         print('Reset ON')
197         print(str(res_on))
198         num = 0
199         segment7(num)
200     response = web_page(num)
201     conn.send('HTTP/1.1 200 OK\n')
202     conn.send('Content-Type: text/html\n')
203     conn.send('Connection: close\n\n')
204     conn.sendall(response)
205     conn.close()

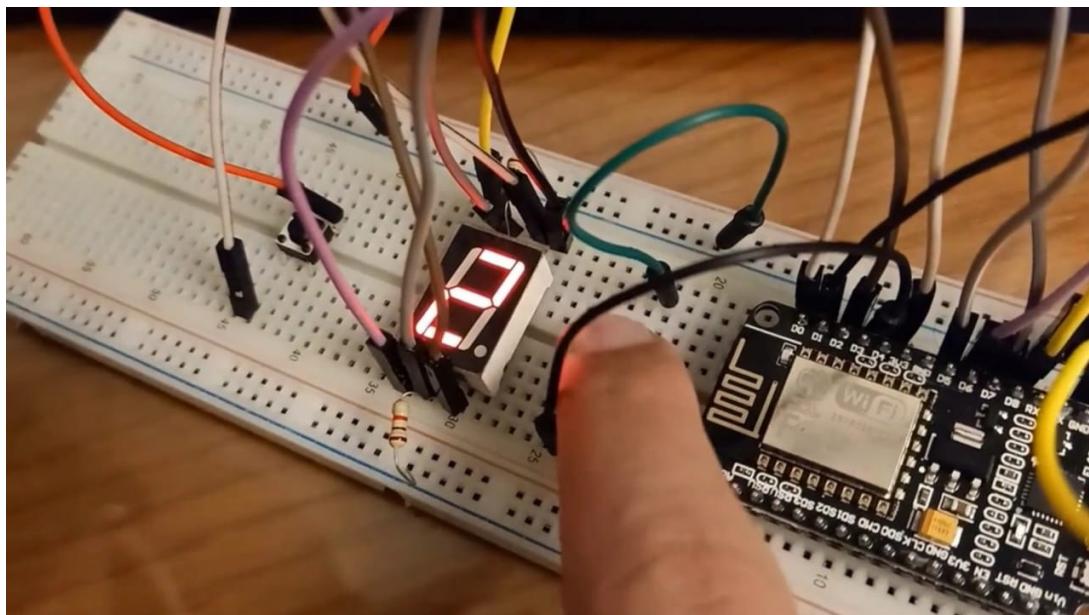
```

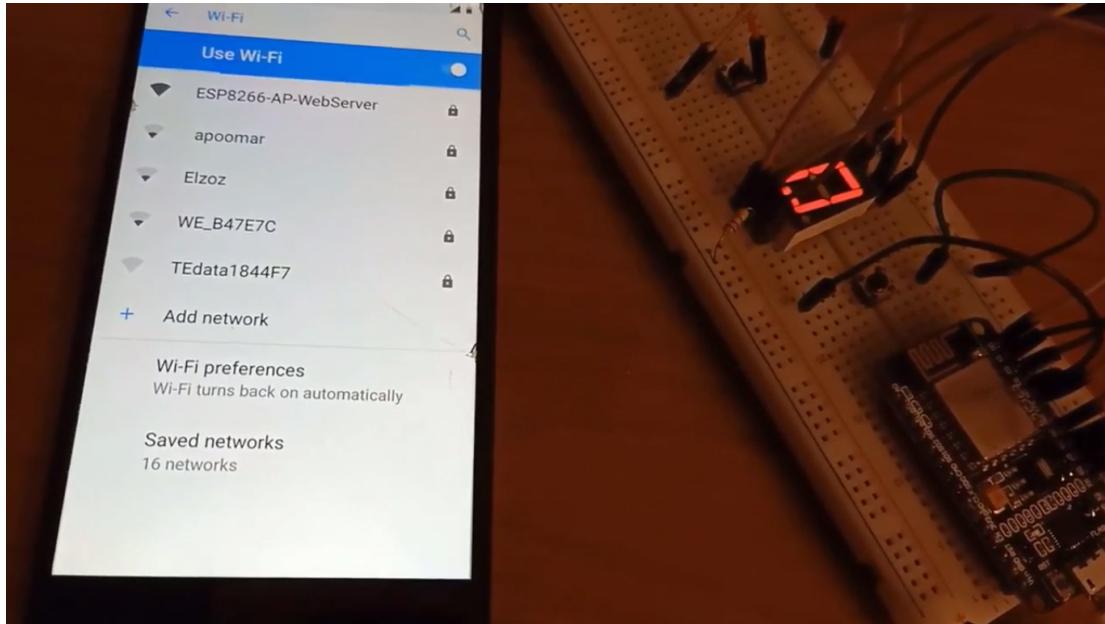
Runtime code
at the start it requests the order that the user want to execute, and they are three orders 1-inc_on 2-dec_on 3- res_on , and prints "request"
if we choose inc_on it prints 'Increment ON' and inc the num with 1 and if the num reaches 10 it becomes =0
if we choose dec_on it prints 'Decrement ON' and inc the num with -1 and if the num reaches -1 it becomes =9
if we choose res_on it prints 'Reset ON' and the num becomes 0
and at the end it write the response and close the connection

THE CIRCUIT AT RUN TIME

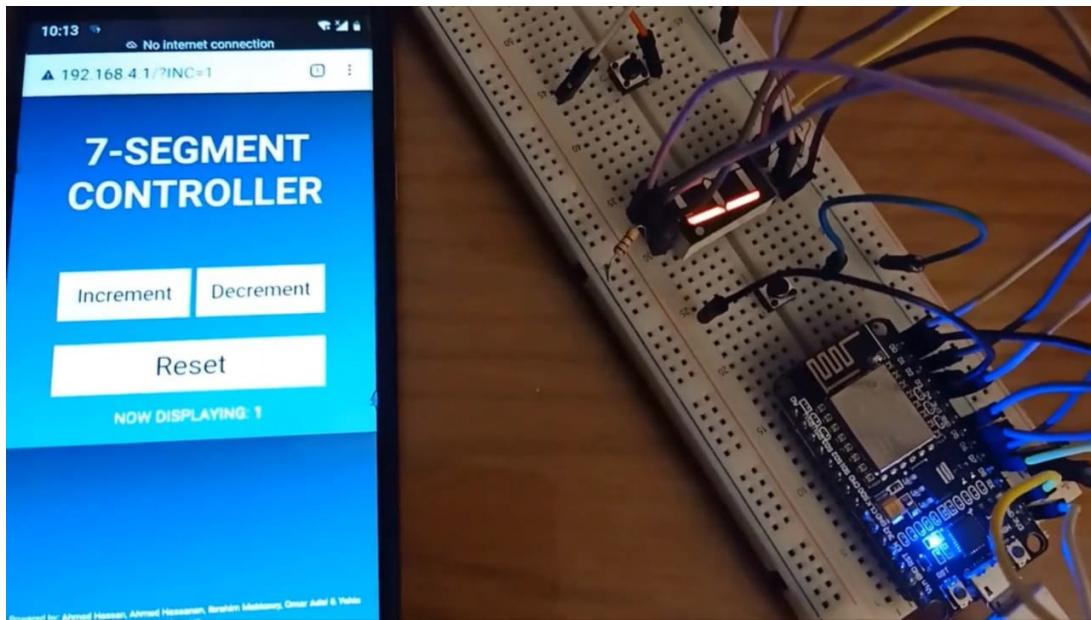


The segment displays 1 then we'll press inc button.





while connecting the esp access point with other device



after press inc button .