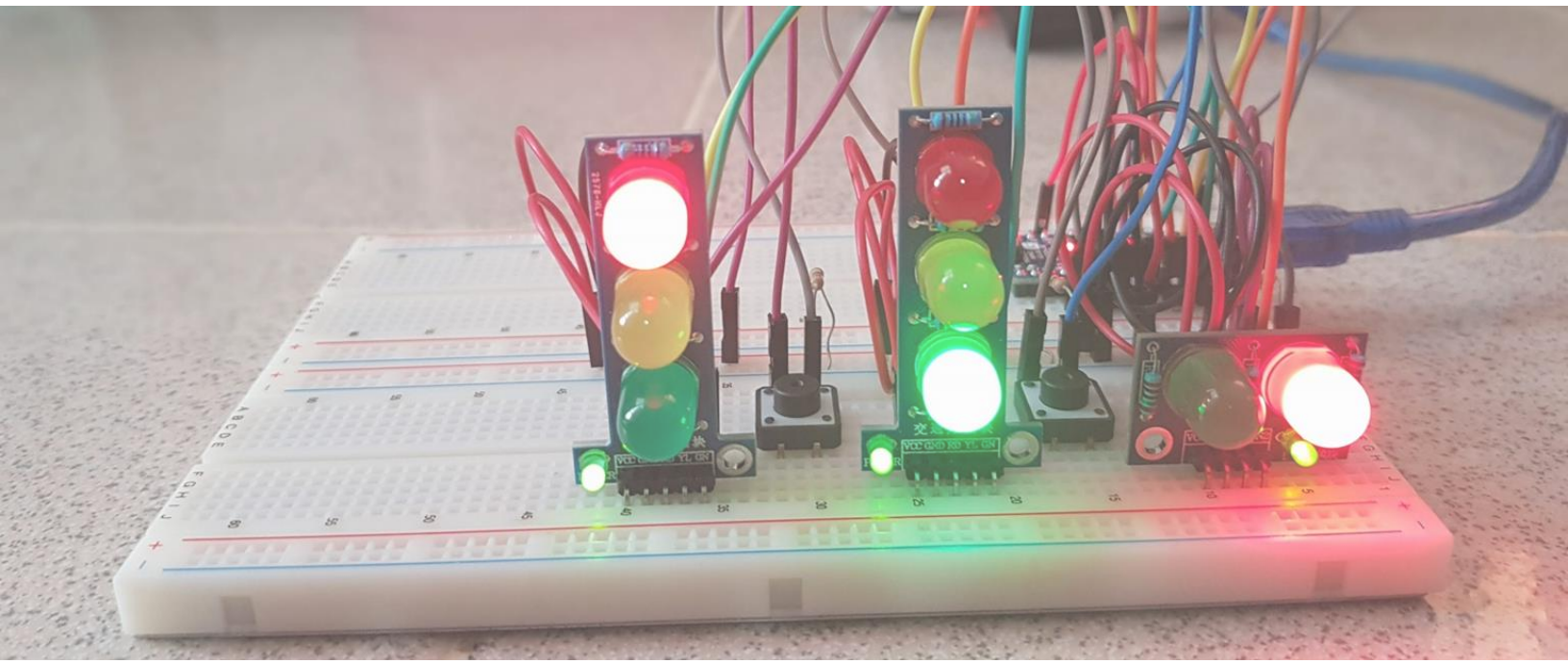
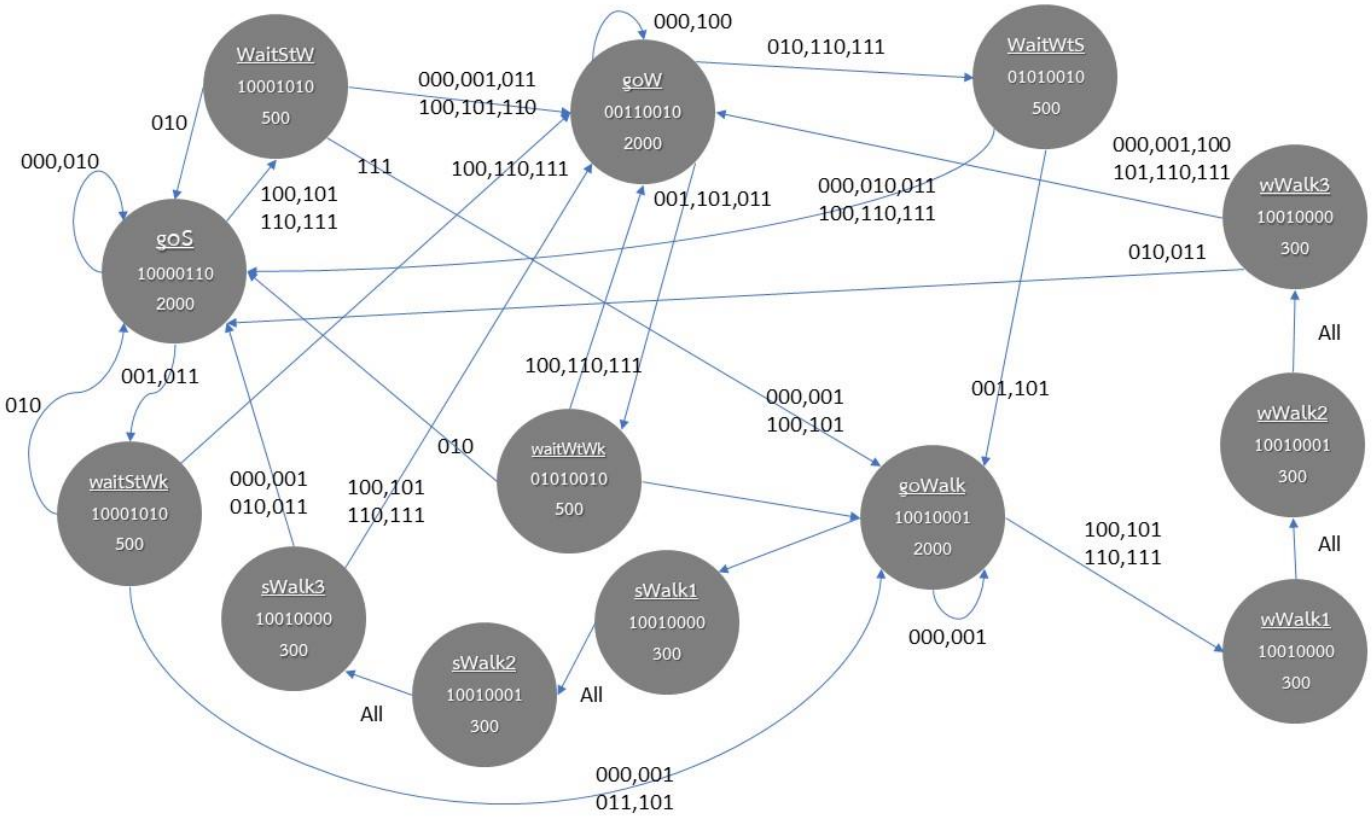
Nibbas : 62011308 & 62010986

Finite State Machine

#Assignment 6



State Transition Graph

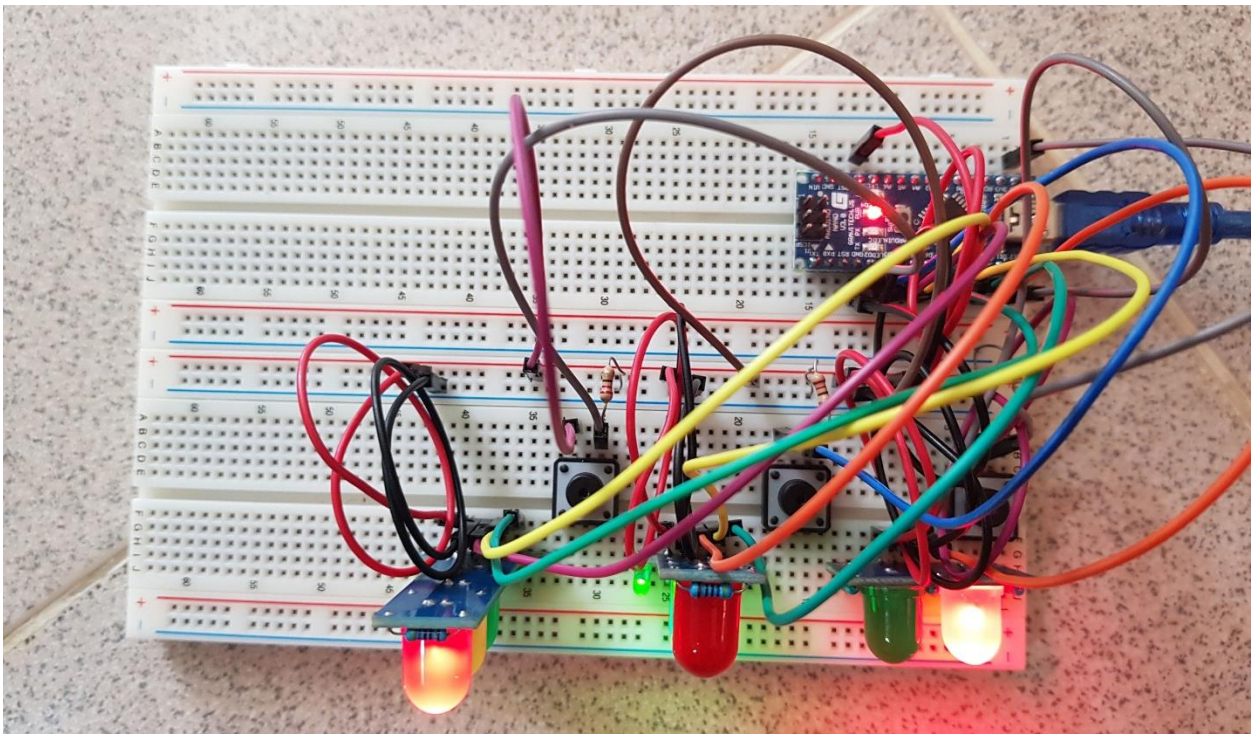


Nibbas : 62011308 & 62010986

State Transition Table

num	Name	Light(wRYG.sRYG.RG)	Time	0(000)	1(001)	2(010)	3(011)	4(100)	5(101)	6(110)	7(111)
0	goS	1000.0110 = 0x86	2000	goS	waitStWk	goS	waitStWk	waitStW	waitStW	waitStW	waitStW
1	waitStW	1000.1010 = 0x8A	500	goW	goW	goS	goW	goW	goW	goW	goWalk
2	goW	0011.0010 = 0x32	2000	goW	waitWtWk	waitWtS	waitWtWk	goW	waitWtWk	waitWtS	waitWtS
3	waitWtS	0101.0010 = 0x52	500	goS	goWalk	goS	goS	goS	goWalk	goS	goS
4	goWalk	1001.0001 = 0x91	2000	goWalk	goWalk	sWalk1	sWalk1	wWalk1	wWalk1	wWalk1	wWalk1
5	wWalk1	1001.0000 = 0x90	300	wWalk2	wWalk2	sWalk2	sWalk2	wWalk2	wWalk2	wWalk2	wWalk2
6	wWalk2	1001.0001 = 0x91	300	wWalk3	wWalk3	sWalk3	sWalk3	wWalk3	wWalk3	wWalk3	wWalk3
7	wWalk3	1001.0000 = 0x90	300	goW	goW	goS	goS	goW	goW	goW	goW
8	waitStWk	1000.1010 = 0x8A	500	goWalk	goWalk	goS	goWalk	goW	goWalk	goW	goW
9	sWalk1	1001.0000 = 0x90	300	sWalk2	sWalk2	sWalk2	sWalk2	wWalk2	wWalk2	wWalk2	wWalk2
10	sWalk2	1001.0001 = 0x91	300	sWalk3	sWalk3	sWalk3	sWalk3	wWalk3	wWalk3	wWalk3	wWalk3
11	sWalk3	1001.0000 = 0x90	300	goS	goS	goS	goS	goW	goW	goW	goW
12	waitWtWk	0101.0010 = 0x52	500	goWalk	goWalk	goS	goWalk	goW	goWalk	goW	goW

Circuit



Source Code

1. สร้าง structer ของ State โดยเก็บค่า 8-bit ที่จะให้ไฟสว่าง เวลา และค่าของ State ที่จะให้ทำงานต่อไป

```
27 struct State {
28   unsigned long ST_Out; // 8-bit pattern to street output
29   unsigned long Time; // delay in ms units
30   unsigned long Next[12];
31 }; // next state for inputs 0,1,2,3,4,5,6,7,8,9,10,11,12
```

2. กำหนดค่าให้แต่ละ State จาก State Transition Table

```
33 typedef const struct State SType;
34 SType FSM[13] = {
35   {0x86, 2000, {goS, waitStWk, goS, waitStWk, waitStW, waitStW, waitStW, waitStW}}, //goS 0
36   {0x8A, 500, {goW, goW, goS, goW, goW, goW, goW, goWalk}}, //waitS 1
37   {0x32, 2000, {goW, waitWtWk, waitWtS, waitWtWk, goW, waitWtWk, waitWtS, waitWtS}}, //goW 2
38   {0x52, 500, {goS, goWalk, goS, goS, goS, goWalk, goS, goS}}, //waitW 3
39   {0x91, 2000, {goWalk, goWalk, sWalk1, sWalk1, wWalk1, wWalk1, wWalk1, wWalk1}}, //goWalk 4
40   {0x90, 300, {wWalk2, wWalk2, sWalk2, sWalk2, wWalk2, wWalk2, wWalk2, wWalk2}}, //wWalk1 5
41   {0x91, 300, {wWalk3, wWalk3, sWalk3, sWalk3, wWalk3, wWalk3, wWalk3, wWalk3}}, //wWalk2 6
42   {0x90, 300, {goW, goW, goS, goS, goW, goW, goW, goW}}, //wWalk3 7
43   {0x8A, 500, {goWalk, goWalk, goS, goWalk, goW, goWalk, goW, goW}}, //waitStWk 8
44   {0x90, 300, {sWalk2, sWalk2, sWalk2, sWalk2, wWalk2, wWalk2, wWalk2, wWalk2}}, //sWalk1 5
45   {0x91, 300, {sWalk3, sWalk3, sWalk3, sWalk3, wWalk3, wWalk3, wWalk3, wWalk3}}, //sWalk2 6
46   {0x90, 300, {goS, goS, goS, goS, goW, goW, goW, goW}}, //sWalk3 7
47   {0x52, 500, {goWalk, goWalk, goS, goWalk, goW, goWalk, goW, goW}} //waitWtWk 12
48 };
```

3. นำค่า bit มากำหนดสถานะไฟแต่ละดวงและ delay ตามค่าของ State ปัจจุบัน

```
61   digitalWrite(LED_W_R, !(FSM[S].ST_Out & B10000000));
62   digitalWrite(LED_W_Y, !(FSM[S].ST_Out & B01000000));
63   digitalWrite(LED_W_G, !(FSM[S].ST_Out & B00100000));
64   digitalWrite(LED_S_R, !(FSM[S].ST_Out & B00010000));
65   digitalWrite(LED_S_Y, !(FSM[S].ST_Out & B00001000));
66   digitalWrite(LED_S_G, !(FSM[S].ST_Out & B00000100));
67   digitalWrite(LED_Walk_R, !(FSM[S].ST_Out & B00000010));
68   digitalWrite(LED_Walk_G, !(FSM[S].ST_Out & B00000001));
69   delay(FSM[S].Time);
```

4. อ่านค่าจาก Switch แล้วนำมาคำนวณหา State ต่อไปที่จะทำงาน

```
71   input1 = digitalRead(WEST_BUTTON_PIN);
72   input2 = digitalRead(SOUTH_BUTTON_PIN);
73   input3 = digitalRead(WALK_BUTTON_PIN);
74   input = input1 * 4 + input2 * 2 + input3;
75   S = FSM[S].Next[input];
```