

ΑΣΚΗΣΗ 1 ΜΗΧΑΤΡΟΝΙΚΗΣ

Να σχεδιάσετε ένα σύστημα το οποίο, όταν πατηθεί ένας διακόπτης επαναφοράς (push button), ελέγχει την κατάσταση τριών διακοπών on/off, και εξετάζει αν ο συνδυασμός των παραπάνω διακοπών συμφωνεί με αυτόν ο οποίος έχει αποθηκευτεί στο μικροελεγκτή του παραπάνω συστήματος.

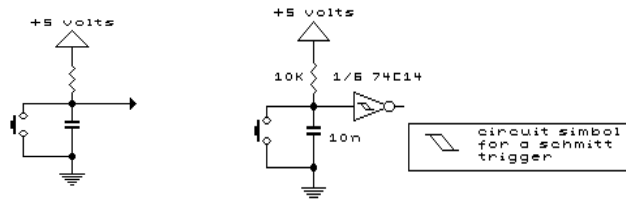
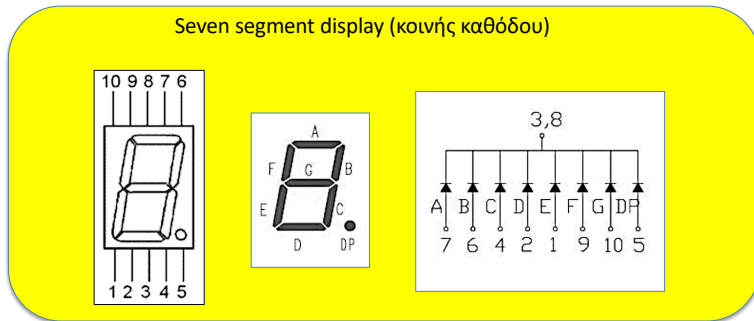
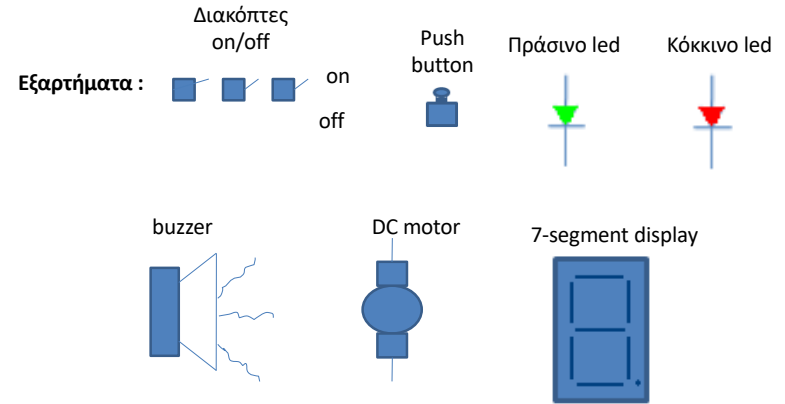
Αν ο συνδυασμός είναι σωστός, ένα πράσινο led ανάβει και ένας DC κινητήρας ξεκινάει να περιστρέφεται για όσο χρονικό διάστημα ο παραπάνω διακόπτης επαναφοράς (push button) είναι πατημένος.

Ενώ αν ο συνδυασμός είναι λάθος, ένα κόκκινο led ανάβει, μια ηχητική σειρήνα (buzzer) ηχεί για χρονικό διάστημα 3sec και ταυτόχρονα απεικονίζεται σε ένα display ο αύξων αριθμός των αποτυχημένων προσπαθειών.

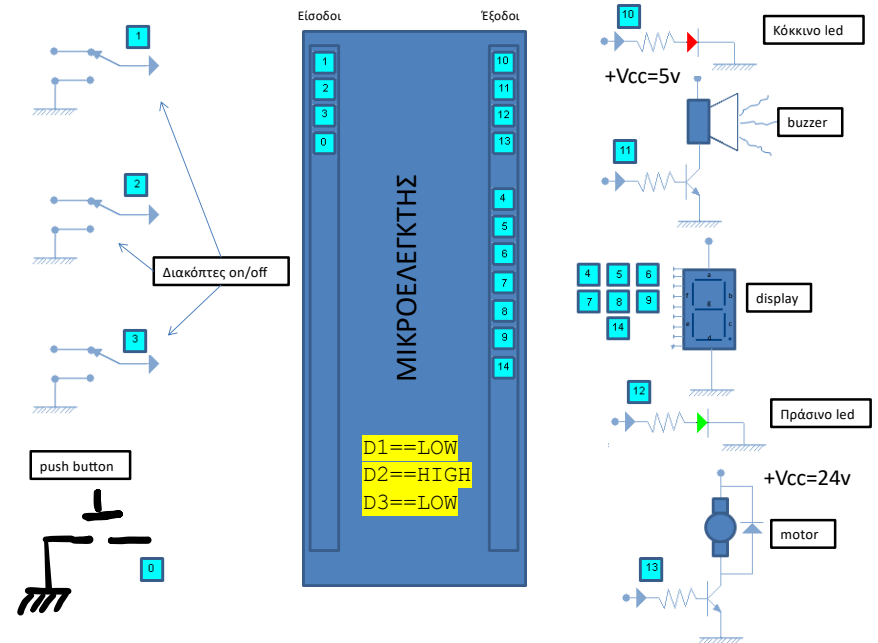
Τέλος όταν εισάγετε ένας σωστός συνδυασμός η ένδειξη του display να μηδενίζεται.

ΛΥΣΗ

Αρχικά απεικονίζουμε τα εξαρτήματα που μας δίνει το πρόβλημα και στη συνέχεια τα χωρίζουμε σε εισόδους και εξόδους του μικροελεγκτή.



Η ενεργοποιώ τις αντιστάσεις PULL-UP.



```

int diakoptis1=1, diakoptis2=2, diakoptis3=3, push_button=0;
int apotiximenes_prospathies=0;
int red_led=10, green_led=12, motor =13, buzzer=11, a=4, b=5, c=6, d=7, e=8, f=9,
g=14;
int D1=0,D2=0,D3=0;
void setup()
{
  pinMode(diakoptis1, INPUT);
  pinMode(diakoptis2, INPUT);
  pinMode(diakoptis3, INPUT);
  pinMode(push_button, INPUT);
  digitalWrite(push_button, INPUT_PULLUP);
  digitalWrite(diakoptis1, INPUT_PULLUP);
  digitalWrite(diakoptis2, INPUT_PULLUP);
  digitalWrite(diakoptis3, INPUT_PULLUP);
  pinMode(a, OUTPUT);
  pinMode(b, OUTPUT);
  pinMode(c, OUTPUT);
  pinMode(d, OUTPUT);
  pinMode(e, OUTPUT);
  pinMode(f, OUTPUT);
  pinMode(red_led, OUTPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(green_led, OUTPUT);
  pinMode(motor, OUTPUT);
  pinMode(g, OUTPUT);
}

```

Εδώ δηλώνω ακέραιες μεταβλητές που αντιστοιχούν στο κάθε pin του μικροελεγκτή

Στη void setup() δηλώνω, ποια pin του μικροελεγκτή θα είναι είσοδοι (INPUT) και ποια θα είναι έξοδοι(OUTPUT)..

```

void loop()
{
  if (digitalRead(push_button) == LOW)
  {
    D1=digitalRead(diakoptis1);
    D2=digitalRead(diakoptis2);
    D3=digitalRead(diakoptis3);
    if (D1==LOW && D2==HIGH && D3==LOW)
    {
      digitalWrite(green_led,HIGH); //on πράσινου led
      digitalWrite(motor,HIGH); // on εκκίνηση του DC κινήτρου
      apotiximenes_prospathies=0;
      zero();
    }
    else {
      apotiximenes_prospathies++
      switch (apotiximenes_prospathies)
      {
        case 0 :
          zero(); break;
        case 1 :
          one(); break;
        case 2 :
          two(); break;
        .....
        case 9 :
          nine(); break;
      }
      digitalWrite(red_led,HIGH);
      digitalWrite(buzzer,HIGH); // led και buzzer
      delay(3000);
      digitalWrite(buzzer,LOW);
    }
  }
}

```

Με την if, μόλις πατηθεί το push button γίνεται έλεγχος αν ο συνδυασμός είναι αληθής, αν είναι σωστός τα pin 12 και 13 θα γίνουν HIGH με σκοπό τη φωτοβολία του πράσινου led και την εκκίνηση του DC κινήτρου. Αν ο συνδυασμός είναι λανθασμένος με την else αυξάνει ο μετρητής των αποτυχημένων προσαθειών κατά ένα.

Με την switch ανάλογα με το πόσες αποτυχημένες προσπάθειες έχουμε κάνει καλεί την αντίστοιχη συνάρτηση με την οποία θα αποτυπωθεί στο display ο αριθμός των αποτυχημένων προσαθειών

Στη συνέχεια τα pin 10 και 11 θα γίνουν HIGH με σκοπό να ανάψει το κόκκινο led και να ηχεί buzzer για 3sec, μετά τα 3sec το pin 11 LOW και το buzzer θα σταματήσει ..

```

void zero() { digitalWrite(a,HIGH); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,HIGH); digitalWrite(e,HIGH); digitalWrite(f,HIGH); digitalWrite(g,LOW); }
void one() { digitalWrite(a,LOW); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,LOW); digitalWrite(e,LOW); digitalWrite(f,LOW); digitalWrite(g,LOW); }
void two() { digitalWrite(a,HIGH); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,HIGH); digitalWrite(e,LOW); digitalWrite(f,LOW); digitalWrite(g,HIGH); }
void three() { digitalWrite(a,HIGH); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,HIGH); digitalWrite(e,LOW); digitalWrite(f,LOW); digitalWrite(g,HIGH); }
void four() { digitalWrite(a,LOW); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,LOW); digitalWrite(e,LOW); digitalWrite(f,HIGH); digitalWrite(g,HIGH); }
void five() { digitalWrite(a,HIGH); digitalWrite(b,LOW); digitalWrite(c,HIGH); digitalWrite(d,HIGH); digitalWrite(e,LOW); digitalWrite(f,HIGH); digitalWrite(g,HIGH); }
void six() { digitalWrite(a,HIGH); digitalWrite(b,LOW); digitalWrite(c,HIGH); digitalWrite(d,HIGH); digitalWrite(e,HIGH); digitalWrite(f,HIGH); digitalWrite(g,HIGH); }
void seven() { digitalWrite(a,HIGH); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,LOW); digitalWrite(e,LOW); digitalWrite(f,LOW); digitalWrite(g,LOW); }
void eight() { digitalWrite(a,HIGH); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,HIGH); digitalWrite(e,HIGH); digitalWrite(f,HIGH); digitalWrite(g,HIGH); }
void nine() { digitalWrite(a,HIGH); digitalWrite(b,HIGH); digitalWrite(c,HIGH); digitalWrite(d,LOW); digitalWrite(e,LOW); digitalWrite(f,HIGH); digitalWrite(g,HIGH); }

```