

Μάθημα: Αισθητήρες-Ενεργοποιητές, Μηχατρονική: Έλεγχος DC motor με Arduino

Στόχοι:

- α) κατανόηση διαφορετικών τεχνικών ελέγχου dc motors μέσω προγραμματισμού του Arduino
- β) να γνωρίσουν τη χρησιμότητα του τρανζίστορ οδήγησης για την παροχή επαρκούς ρεύματος για την κίνηση του dc motor

Τα Υλικά που θα χρειαστούμε:

1. Πλακέτα Arduino Uno,
2. 1x ποτενσιόμετρο π.χ. 10KΩ, 22KΩ
3. 1 αντίσταση 220 Ω
4. 1 αντίσταση 10 KΩ
5. 1 διακόπτη – button (on/off) σε συνδεσμολογία pull down με την αντίσταση 10 KΩ
6. 1 δίοδο 1N4004
7. 1 NPN τρανζίστορ BC140 ή 2N2222A ή PN2222A (TO-92)
8. 1 DC motor 6Volts (ώστε να οδηγείται από το κύκλωμα 5Volts της πλακέτας Arduino) αλλιώς μπορούμε να χρησιμοποιήσουμε ανεξάρτητο κύκλωμα με ρελέ

Το Αντικείμενο της άσκησης:

Με την πρώτη εφαρμογή ελέγχουμε την περιστροφή του dc motor με ένα διακόπτη button οπότε το μοτέρ κινείται είτε με μηδενική ταχύτητα (button=off) είτε με μέγιστη ταχύτητα (button=ON) όπως για παράδειγμα σε έναν αναδευτήρα (χτυπητήρι) καφέ. Στην εφαρμογή αυτή χρησιμοποιούμε ακροδέκτες ψηφιακής εισόδου και ψηφιακής εξόδου με τις εντολές `digitalRead` και `digitalWrite`.

Με τη δεύτερη εφαρμογή η ταχύτητα περιστροφής του dc motor ελέγχεται με ένα ποτενσιόμετρο ακολουθώντας την τεχνική PWM παράγοντας έτσι ένα τετραγωνικό παλμό μεταβλητού κύκλου εργασίας (duty cycle) όπως είδαμε και σε προηγούμενα φύλλα έργου. Χρησιμοποιούμε αναλογικό ακροδέκτη εισόδου με την εντολή `analogRead` και παράγουμε σήμα PWM (που ισοδυναμεί σε αναλογικό σήμα εξόδου 0-5V) σε ψηφιακό ακροδέκτη με την ένδειξη ~ και την εντολή `analogWrite`.

Τι πρέπει να γνωρίζουμε:

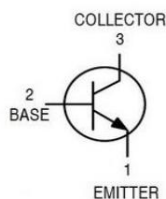
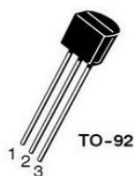
Ο ρόλος του τρανζίστορ: Και στις δύο εφαρμογές ο ακροδέκτης + του μοτέρ συνδέεται στα 5 Volts ενώ ο ακροδέκτης – συνδέεται στο συλλέκτη του τρανζίστορ NPN. Ο εκπομπός του τρανζίστορ είναι στη γείωση του κυκλώματος ενώ η βάση του τρανζίστορ οδηγείται από τον αντίστοιχο ακροδέκτη εξόδου της πλακέτας Arduino UNO μέσω αντίστασης 220Ω. Χρησιμοποιούμε το τρανζίστορ ως ένα ενδιάμεσο διακόπτη ο οποίος αφενός μεν λειτουργεί με το πολύ μικρό ρεύμα εξόδου που δίνει η πλακέτα Arduino αφετέρου δε παράγει στο κύκλωμα συλλέκτη-εκπομπού πολύ μεγαλύτερο ρεύμα ώστε να περιστραφεί ο κινητήρας. Οι ακροδέκτες των τρανζίστορ δίνονται στις παρακάτω φωτογραφίες

BC140 NPN TRANSISTOR

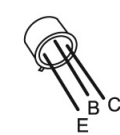


1 : EMITTER
2 : BASE
3 : COLLECTOR

2N2222



2N2222

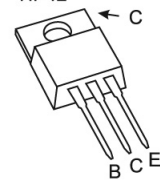


E = EMITTER
B = BASE
C = COLLECTOR

**2N3904
2N3906
2N4401
2N4403**



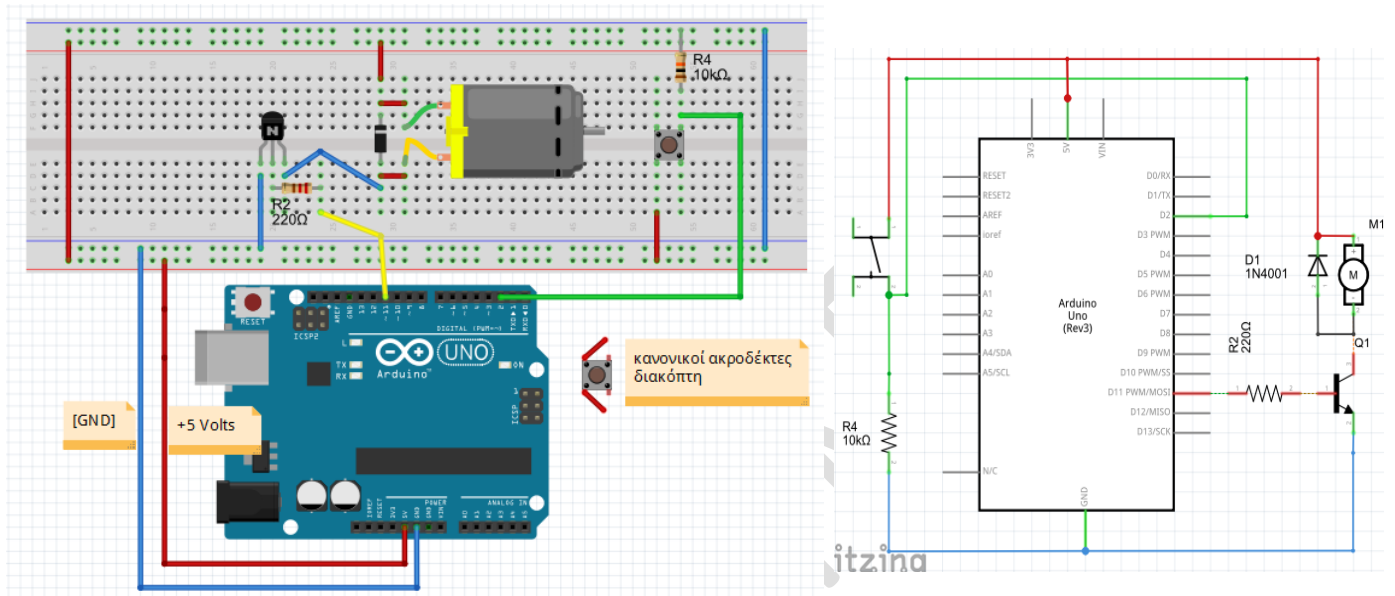
**TIP41
TIP42**



Copyright 2008 Michael A. Covington

Ο ρόλος της διόδου: συνδέουμε μια δίοδο παράλληλα στους ακροδέκτες του DC motor ανάστροφα πολωμένη. Η δίοδος επιτρέπει τη διέλευση ρεύματος μόνο προς μια κατεύθυνση και ο ρόλος της είναι να εμποδίζει το αντίστροφο ρεύμα που παράγεται από το DC motor, όταν κλείνουμε την τροφοδοσία του, να περάσει στην πλακέτα Arduino και να την καταστρέψει.

Εφαρμογή 1^η: Στο παρακάτω σχέδιο (έγινε με το Fritzing) βλέπουμε τη πρώτη εφαρμογή με τον κινητήρα σε λειτουργία ON/OFF και παρακάτω τον αντίστοιχο κώδικα σε Wiring C.



// εδώ το μοτέρ δουλεύει με τη μέγιστη ταχύτητα

```
const int switchpin = 2; // δήλωση του ψηφιακού ακροδέκτη εισόδου pin switch
const int motorpin = 11; // δήλωση του ψηφιακού ακροδέκτη εξόδου pin motor
int switchState = 0; // δήλωση μεταβλητής που ανιχνεύει την κατάσταση του button
```

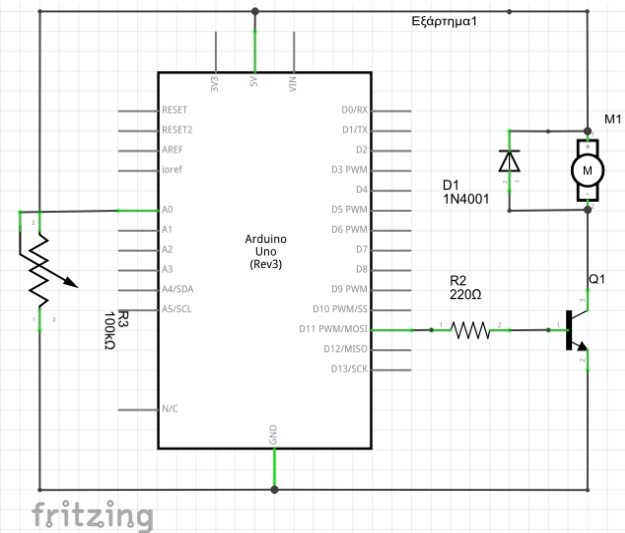
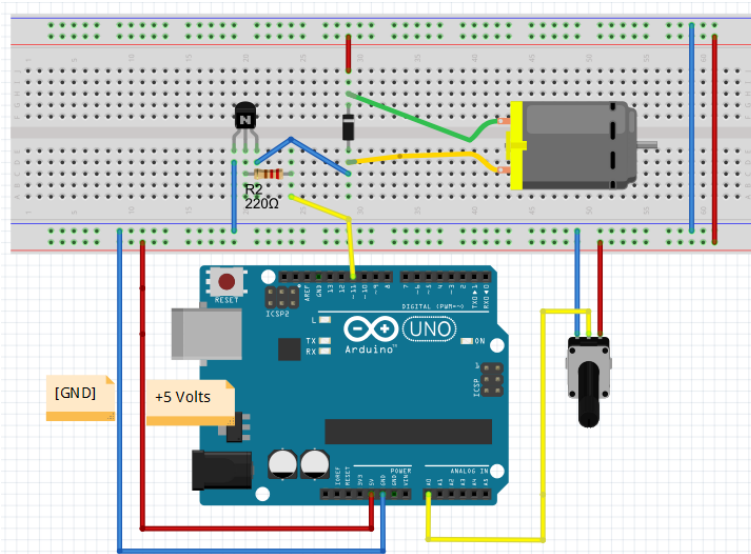
// κώδικας που εκτελείται μόνο μια φορά

```
void setup() {
  pinMode(switchpin, INPUT); // ρύθμιση ώστε ο ακροδέκτης switchpin να λειτουργεί ως είσοδος
  pinMode(motorpin, OUTPUT); // ρύθμιση ώστε ο ακροδέκτης motorpin να λειτουργεί ως έξοδος
}
```

// κώδικας που επαναλαμβάνεται συνεχώς

```
void loop() {
  switchState = digitalRead(switchpin); // διαβάζουμε την κατάσταση του button
  if (switchState == HIGH) // αν το button έχει πατηθεί
  {
    digitalWrite(motorpin, HIGH); // στέλνουμε 5V στη βάση του τρανζίστορ οπότε άγει
  }
  else
  {
    digitalWrite(motorpin, LOW); // αλλιώς 0V στη βάση του τρανζίστορ οπότε δεν άγει
  }
}
```

Εφαρμογή 2^η: Στο παρακάτω σχέδιο (έγινε με το *Fritzing*) βλέπουμε τη δεύτερη εφαρμογή με τον ακροδέκτη εξόδου του ποτενσιόμετρου να είναι συνδεδεμένος στον αναλογικό ακροδέκτη A0 της πλακέτας. Περιστρέφοντας το ποτενσιόμετρο μεταβάλλουμε και την ταχύτητα περιστροφής του μοτέρ.



// εδώ το μοτέρ δουλεύει με μεταβλητή ταχύτητα

```
const int potpin = A0; // δήλωση του αναλογικού ακροδέκτη εισόδου potpin για ποτενσιόμε.
const int motorpin = 11; // δήλωση του ψηφιακού ακροδέκτη εξόδου motorpin που θα
// λειτουργεί με την τεχνική PWM και τροφοδοτεί τη βάση
// του τρανζίστορ
int val = 0; // δήλωση της μεταβλητής που θα διαβάζει την τάση από το
// ποτενσιόμετρο
```

// κώδικας που εκτελείται μόνο μια φορά

```
void setup() {
  pinMode(motorpin, OUTPUT); // δεν είναι υποχρεωτική η δήλωση για ακροδέκτες εξόδου PWM
}
```

// κώδικας που επαναλαμβάνεται συνεχώς

```
void loop() {
  val = analogRead(potpin); //εκχώρηση στη μεταβλητή val της αναλογικής τάσης από ποτενσιόμετρο
  analogWrite(motorpin, val / 4); // η αναλογική τάση εισόδου αντιστοιχεί σε επίπεδα 0-1023 (10 bit
  // A/D) ενώ η αναλογική τάση εξόδου αντιστοιχεί σε επίπεδα 0-255.
  // Έτσι διαιρούμε τα 0-1023 επίπεδα με το 4 ώστε να αντιστοιχούν
  // σε 0-255 επίπεδα
}
```

** Μπορούμε να κολλήσουμε ένα χαρτάκι στον άξονα του μοτέρ για να βλέπουμε καλύτερα την περιστροφή.

ΚΑΛΗ ΕΠΙΤΥΧΙΑ