

Τα μυστικά των βηματικών κινητήρων

Τ.Γ.Κωνσταντίνου

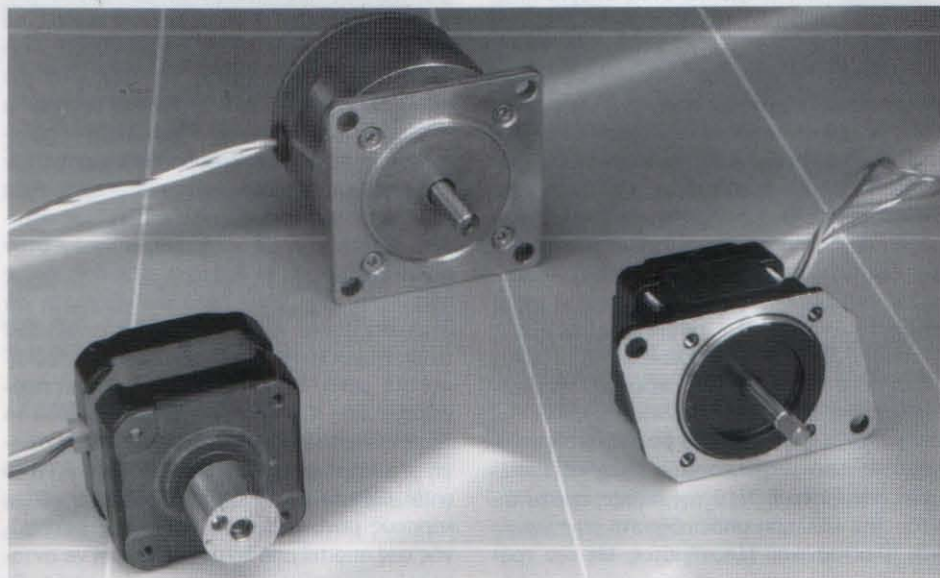
Μέρος 1ο: Αρχές λειτουργίας

Οι βηματικοί κινητήρες και ειδικότερα οι τρόποι οδήγησής τους αποτελούν ακόμα και σήμερα ένα δισεπίλυτο γρίφο για τους περισσότερους αναγνώστες μας. Το άρθρο αυτό, που θα ολοκληρωθεί σε δύο μέρη, δίνει απαντήσεις σε πολλά ερωτήματα καθώς καλύπτει τόσο τις αρχές λειτουργίας τους όσο και κάποιες ενδιαφέρουσες και αντιπροσωπευτικές εφαρμογές. Στο δεύτερο μέρος θα γίνει ιδιαίτερη αναφορά σε ένα κύκλωμα ισχύος τεσσάρων καναλιών ελεγχόμενο από τη θύρα ενός συνηθισμένου PC.

Δεν είναι λίγες οι φορές που ένας ηλεκτρονικός καλείται να συνδυάσει ένα τυπικό κύκλωμα ελέγχου με έναν κινητήρα, προκειμένου να επιτύχει την κίνηση ενός φορτίου ή, γενικότερα, την περιστροφή ενός άξονα. Ο απλούστερος τρόπος να το πετύχει αφορά στη χρήση ενός συνηθισμένου κινητήρα Συνεχούς (DC) ή Εναλλασσόμενου Ρεύματος (AC) που έχει χαμηλό κόστος και συνάμα οδηγείται από σχετικά απλά κυκλώματα. Μια περισσότερο όμως προσεκτική μελέτη αποδεικνύει πως αυτή η λύση δεν είναι τόσο καλή όσο φαίνεται με τη πρώτη ματιά. Και αυτό γιατί με ένα τέτοιο εξάρτημα είναι πολύ δύσκολο να προσδιοριστεί η ακριβής θέση και η ταχύτητα περιστροφής του περιστρεφόμενου άξονα και κατ' επέκταση το κατά πόσο έχει μετακινηθεί το φορτίο που είναι κομπλαρισμένο σε αυτόν. Αν κάποιος θελήσει να μάθει την ακριβή θέση του τελευταίου θα πρέπει να εξασφαλίσει την παρουσία κατάλληλων κυκλωμάτων με ανάδραση θέσης (κυκλώματα κλειστού βρόγχου) χωρίς να αποκλείσει τη χρήση φρένων και συμπλεκτών. Η καλύτερη λύση προβλέπει τη χρήση βηματικών κινητήρων, ενός είδους ηλεκτρομηχανικών εξαρτημάτων, ικανών να μετατρέπουν το πλήθος και τη συχνότητα των παλμών που τους διεγείρουν σε γωνία και ταχύτητα περιστροφής του άξονα τους αντίστοιχα.

Τυπικές εφαρμογές βηματικών κινητήρων

Τους βηματικούς κινητήρες θα τους συνα-



ντήσουμε σε υπολογιστές και περιφερειακά υπολογιστών (σαρωτές, μηχανισμούς σκληρών ή μαλακών δίσκων, εκτυπωτές κλπ), φωτογραφικές μηχανές, τηλεσκόπια και συστήματα προσανατολισμού δορυφορικών 'πιάτων', ιατρικά μηχανήματα, ρομποτικούς βραχίονες, συστήματα ανάγνωσης γραμμωτού κώδικα και ηλεκτρομηχανικά συστήματα ή εργαλεία ακριβείας.

Τύποι βηματικών κινητήρων

Με τη πρώτη ματιά οι βηματικοί κινητήρες ξεχωρίζουν από τους συνηθισμένους στο μέγεθος, στο βάρος αλλά και στο πλήθος των

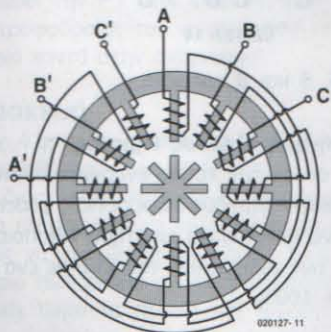
καλωδίων που καταλήγουν σε αυτούς. Στην πραγματικότητα οι ουσιαστικές διαφορές τους βρίσκονται στο εσωτερικό τους. Για να τις κατανοήσουμε και να τις αξιολογήσουμε θα πρέπει να κάνουμε μια σύντομη αναφορά στους τύπους των συγκεκριμένων κινητήρων.

Κινητήρας Μεταβλητής Αντίδρασης (σχ. 1)

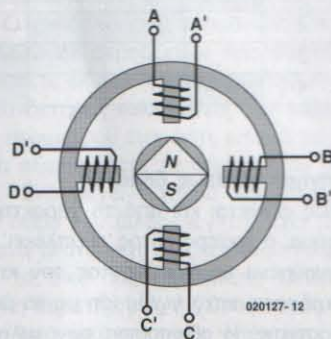
Αυτός ο τύπος κινητήρα στερείται μόνιμων μαγνητών. Για τον λόγο αυτό η περιστροφή του άξονά του πραγματοποιείται χωρίς της επίδραση κάποιας εξαναγκασμένης ή 'ανασταλτικής' ροπής. Χρησιμοποιείται σε εφαρμογές

Χαρακτηριστικά των θηματικών κινητήρων

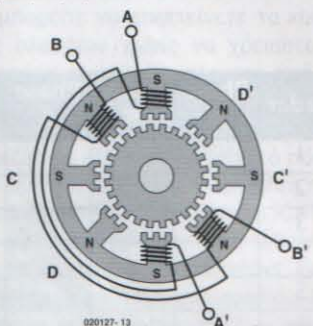
- * Η γωνία περιστροφής είναι ανάλογη του πλήθους των παλμών που δίνονται
- * Η ταχύτητα περιστροφής είναι ανάλογη της συχνότητας των παλμών
- * Εύκολος καθορισμός θέσης του άξονά τους μέσω συστημάτων ανοικτού βρόχου (δεν απαιτούνται κυκλώματα ανάδρασης θέσης)
- * Άριστη ανταπόκριση στις εντολές επιτάχυνσης, επιβράδυνσης και θηματοποίησης
- * Μη αθροιστικό σφάλμα κατά την περιστροφή του άξονα (το σφάλμα μετά από κάθε κίνηση είναι ίσο με +/-5% της γωνίας περιστροφής που αντιστοιχεί σε ένα θήμα)
- * Δυνατότητα χαμηλής ταχύτητας περιστροφής και υψηλής ροπής χωρίς τη χρήση μειωτήρων
- * Ανάπτυξη ροπής συγκράτησης (αν ζητηθεί)
- * Εσωτερική ανασταλτική ροπή
- * Δικατευθυντική λειτουργία
- * Άμεση ακινητοποίηση χωρίς καταστροφή του κινητήρα
- * Μακρόχρονη λειτουργία χωρίς προβλήματα λόγω της απουσίας ψηκτρών
- * Σφαιρικοί τριβείς ακριβείας (ανάλογα με τον τύπο και τον κατασκευαστή)



Σχ. 1. Βηματικός κινητήρας μεταβλητής αντίδρασης.



Σχ. 2. Βηματικός κινητήρας με μόνιμο μαγνήτη.



Σχ. 3. Ο υβριδικός βηματικός κινητήρας αποτελεί έναν έξυπνο συνδυασμό των δύο παραπάνω τύπων.

μοίρες.

Μέγεθος

Γενικά, οι βηματικοί κινητήρες ομαδοποιούνται με κριτήριο τις διαστάσεις τους (διάμετρο του σώματός τους). Αν π.χ. ένας κινητήρας έχει μέγεθος 23, το 'σώμα' του έχει διάμετρο 2,3 ίντσες. Τα πιο συνηθισμένα μεγέθη είναι τα 11, 17, 23, 34 και 43.

Ισχύς

Τα επίπεδα ισχύος που δουλεύουν οι βηματικοί κινητήρες κυμαίνονται από μερικές εκατοντάδες mWatt (μικροί κινητήρες) έως αρκετά Watt (μεγαλύτεροι κινητήρες). Η μέγιστη ισχύς απωλειών που μπορούν να 'σπκώσουν' καθορίζεται από τα θερμικά χαρακτηριστικά των τυλιγμάτων τους. Αυτά όμως ση-

Μειονεκτήματα των βηματικών κινητήρων

- * Ανάδειξη παρασιτικών συντονισμών αν δεν ληφθούν τα κατάλληλα μέτρα
- * Αδυναμία λειτουργίας σε πολύ μεγάλες ταχύτητες
- * Αν υπερφορτωθούν χάνουν θήματα, με αποτέλεσμα την ανάγκη επαναρύθμισής τους
- * Συγκρινόμενοι με ένα συνηθισμένο κινητήρα AC ή DC ίδιου μεγέθους παρέχουν αισθητά μικρότερη ροπή

όπου η ανάγκη μεγάλων ροπών είναι μικρή (π.χ. για τη μετακίνηση δρομέων ηλεκτρονικών εξαρτημάτων).

Κινητήρας Μόνιμου μαγνήτη (σχ. 2)

Αντίθετα με τον προηγούμενο τύπο, ο συγκεκριμένος βασίζεται σε μόνιμους μαγνήτες τοποθετημένους στο κινούμενο στέλεχος του (δρομέα). Χαρακτηρίζεται από τη χαμηλή ταχύτητα περιστροφής, την εξ ίσου χαμηλή ροπή του και τους μεγάλους θηματοισμούς (κάθε θήμα αντιστοιχεί σε 45 ή 90 μοίρες). Το χαμηλό όμως κόστος του αντισταθμίζει τις οποιοσδήποτε αδυναμίες κάνοντας τον ιδανικό σε εφαρμογές χαμηλής ισχύος όπως π.χ. στους εκτυπωτές ψεκασμού.

Υβριδικός κινητήρας (σχ. 3)

Ο υβριδικός κινητήρας συνδυάζει τα θετικά χαρακτηριστικά των δύο προηγούμενων ελαχιστοποιώντας τα μειονεκτήματά τους. Αναγνωρίζεται εύκολα από το σταθερό τμήμα του (στάτη) με τις πολλαπλές οδοντώσεις και το μόνιμο μαγνήτη που έχει στο δρομέα. Σε κάθε θήμα του στρέφεται κατά 0,9, 1,8 ή 3,6 μοίρες και χαρακτηρίζεται από υψηλή στατική και δυναμική ροπή. Τα δύο τελευταία γνωρίσματα του επιτρέπουν να στρέφεται αξιόπιστα σε μεγάλες ταχύτητες, γεγονός που τον κάνει ιδανικό στις βιομηχανικές εφαρμογές. Οι πιο συνηθισμένοι υβριδικοί κινητήρες έχουν θήμα που αντιστοιχεί σε γωνία 1,8

μειώνονται σπάνια στα συνοδευτικά τεχνικά εγχειρίδια, αναγκάζοντας τους σχεδιαστές να θεωρούν σαν όριο λειτουργίας τους εκείνο που μετράται όταν η θερμοκρασία που αναπτύσσει το σώμα του κινητήρα ξεπεράσει κατά 65° C τη θερμοκρασία του περιβάλλοντος. Είναι προφανές πως αν ο κινητήρας ψύχεται οι απώλειες του μειώνονται σημαντικά.

Λίγα λόγια ακόμα

Επειδή είμαστε βέβαιοι πως οι αναγνώστες μας θα ήθελαν να μάθουν μερικά ακόμα πράγματα για τους βηματικούς κινητήρες κρίναμε σκόπιμο να συνεχίσουμε την σύντομη αυτή αναφορά επισημαίνοντας ορισμένα επιπλέον χαρακτηριστικά τους.

Το περιστρεφόμενο μαγνητικό πεδίο

Όταν ένα φασικό τυλιγμά του στάτη ενός κινητήρα διαρρέεται από ρεύμα, αναπτύσσει γύρω του μαγνητική ροή που επιδρά στον δρομέα. Ο τελευταίος, προκειμένου να εξισορροπήσει τις δυνάμεις που ασκούνται σε αυτόν στρέφεται κατά μια συγκεκριμένη γωνία και ηρεμεί. Αν οι φάσεις του κινητήρα ενεργοποιούνται η μια μετά την άλλη, τότε ο δρομέας θα αλλάζει συνεχώς γωνία διδοώντας μας την αίσθηση της περιστροφής.

Δημιουργία ροπής

Το μέγεθος της ροπής που αναπτύσσεται από ένα βηματικό κινητήρα εξαρτάται από

πολλούς παράγοντας ανάμεσα στους οποίους ξεχωρίζουν η ταχύτητα θηματισμού, η ένταση του ρεύματος που διαρρέει τα τυλίγματα όπως επίσης και το είδος της βαθμίδας οδήγησης τους. Αιτία δημιουργίας της ροπής αποτελεί η αλληλεπίδραση των μαγνητικών ροών του στάτη και του δρομέα. Πιο συγκεκριμένα, η ροπή αναπτύσσεται τη στιγμή που οι δύο ροές μετατοπίζονται η μια σε σχέση με την άλλη. Ο στάτης είναι κατασκευασμένος από υλικό υψηλής μαγνητικής διαπερατότητας που έχει σαν συνέπεια τον περιορισμό της μαγνητικής ροής στο σώμα του. Αυτό έχει σαν συνέπεια την συγκέντρωση της ροής στους πόλους που περιστοιχίζουν το δρομέα. Η ροπή που αναπτύσσεται είναι ανάλογη της μαγνητικής ροής που με τη σειρά της είναι ανάλογη με την ένταση του ρεύματος που διαρρέει το εκάστοτε διεγερμένο τύλιγμα.

Φάσεις

Σαν φάσεις ορίζουμε το σύνολο των διαφορετικών 'διαδρομών' που μπορεί να ακολουθήσει η μαγνητική ροή στα τυλίγματα του στάτη. Γενικά, οι θηματικοί κινητήρες εργάζονται με δύο φάσεις, αν και είναι πολύ πιθανό να συναντήσουμε κάποιους με τρεις ή με πέντε. Ένας διπολικός κινητήρας βασίζει τη λειτουργία του σε ένα τύλιγμα για κάθε φάση. Αντίθετα ένας μονοπολικός διαθέτει ένα τύλιγμα με μεσαία λήψη ή και δύο τυλίγματα για κάθε μια φάση.

Συνδεσμολογίες των τυλιγμάτων

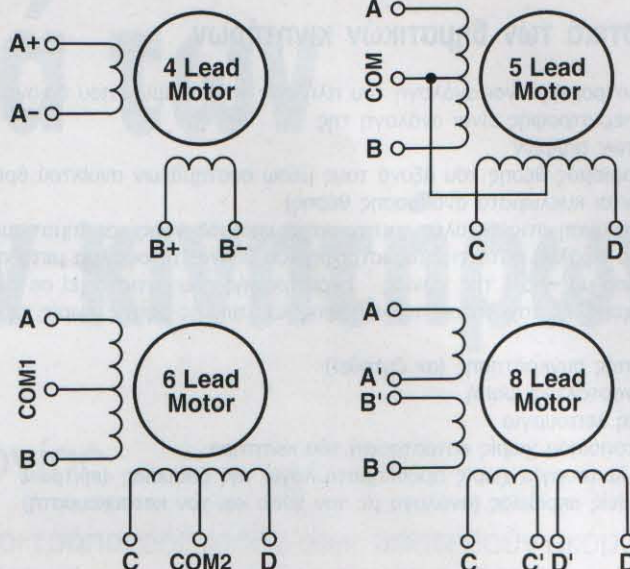
Σήμερα οι θηματικοί κινητήρες διατίθενται στην αγορά σε πολλές παραλλαγές που, μεταξύ άλλων, ξεχωρίζουν από το πλήθος των καλωδίων που βγαίνουν από αυτούς. Πιο αναλυτικά ισχύουν τα παρακάτω:

Καλώδια κινητήρα	Μορφή συνδεσμολογίας
4 καλώδια	Διπολική
5 καλώδια	Μονοπολική,
6 καλώδια	Μονοπολική,
	Διπολική (σύνδεση σειράς)
8 καλώδια	Μονοπολική, Διπολική (σύνδεση σειράς),
	Διπολική (παράλληλη σύνδεση)

Στο σχ. 4 φαίνεται παραστατικά ο τρόπος οδήγησης των κινητήρων σύμφωνα με τον αριθμό των καλωδίων τους

Πόλοι

Οι πόλοι αποτελούν εκείνα τα σημεία του κινητήρα όπου συγκεντρώνεται η μαγνητική ροή. Συνήθως ο αριθμός των πόλων του στάτη συμπίπτει με τον αριθμό των πόλων του δρομέα. Υπάρχουν όμως και εξαιρέσεις όπου



020127-14

Σχ. 4. Βασικές συνδεσμολογίες κινητήρων με 4, 5, 6 και 8 καλώδια.

οι δύο αριθμοί διαφέρουν.

Γωνία περιστροφής(θήμα)

Η γωνία κατά την οποία στρέφεται ο άξονας του κινητήρα σε λειτουργία πλήρους θήματος ορίζεται από την παρακάτω σχέση:

$$\text{Γωνία} = 360 / (n_{ph} \times ph) = 360 / n$$

όπου n_{ph} = αριθμός των ισοδύναμων ζευγών πόλων
 ph = αριθμός των πόλων του δρομέα
 n = συνολικός αριθμός πόλων (σε όλες τις φάσεις)

Η σχέση αυτή έχει νόημα μόνο όταν ο αριθμός των πόλων του στάτη και του δρομέα είναι ίδιοι.

Τρόποι θηματισμού

Οι θηματικοί κινητήρες μπορούν να οδηγηθούν με πολλούς διαφορετικούς τρόπους, οι σημαντικότεροι των οποίων αναφέρονται στη συνέχεια:

Οδήγηση Κύματος

Κύριο χαρακτηριστικό αυτού του τρόπου είναι η οδήγηση μιας μόνο φάσης κάθε χρονική στιγμή. Αυτό σημαίνει πως για ένα μονοπολικό κινητήρα, αξιοποιείται μόνο το 25% των τυλιγμάτων του, ενώ για ένα διπολικό μόνο το 50%.

Order	Phase A	Phase B	Phase \bar{A}	Phase \bar{B}
1				
2				
3				
4				

Οδήγηση πλήρους θήματος

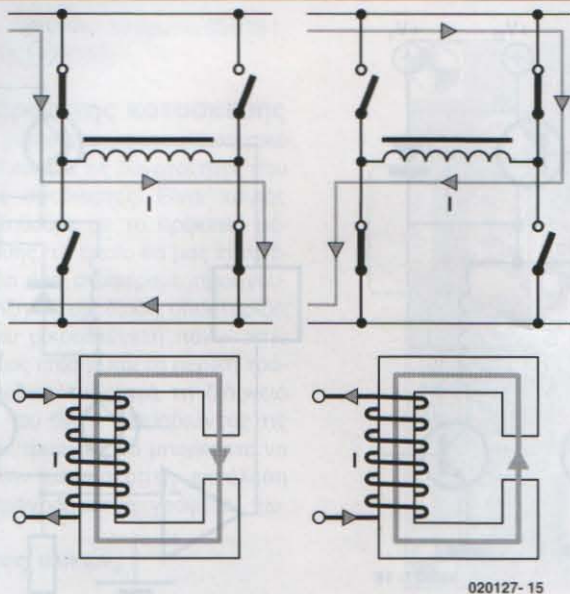
Σε αυτόν τον τρόπο ενεργοποιούνται κάθε χρονική στιγμή δύο διαφορετικές φάσεις. Έτσι, σε ένα μονοπολικό κινητήρα αξιοποιείται το 50% των τυλιγμάτων του, ενώ σε ένα διπολικό το 100%.

Order	Phase A	Phase B	Phase \bar{A}	Phase \bar{B}
1				
2				
3				
4				

Οδήγηση ημίσεως θήματος

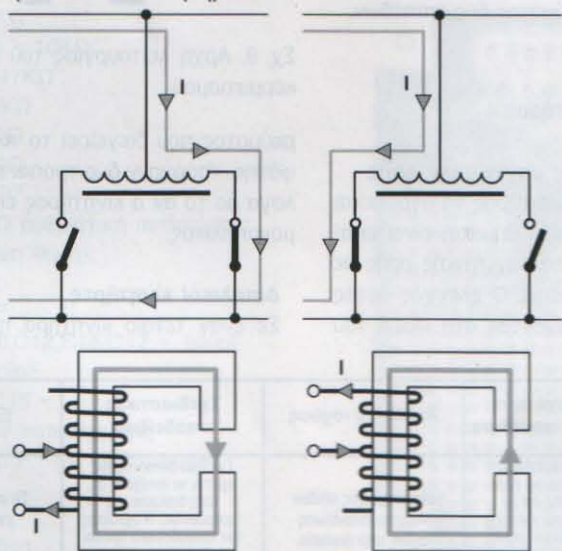
Όπως φαίνεται και από το χαρακτηριστικό γράφημα, ο τρόπος αυτός 'συμπλέκει' τα δύο προηγούμενα υποχρεώνοντας τον κινητήρα να στρέφεται κατά γωνία ίση με το μισό της ονομαστικής. Η αξιοποίηση των τυλιγμάτων για τα δύο είδη κινητήρων είναι ίση με 37,5% ή 75% (μέσες τιμές).

Order	Phase A	Phase B	Phase \bar{A}	Phase \bar{B}
1				
2				
3				
4				
5				
6				
7				
8				



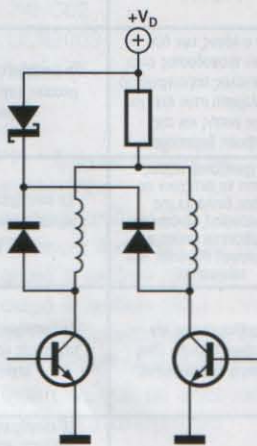
020127- 15

Σχ. 5. Οδήγηση διπολικών κινητήρων.



020127- 16

Σχ. 6. Οδήγηση μονοπολικών κινητήρων.



020127- 17

Σχ. 7. Μια αντίσταση σειράς χρησιμοποιείται πολλές φορές για τον περιορισμό του ρεύματος ενός τυλίγματος

Ανασταλτική ροπή

Η μέγιστη σταθερή ροή που μπορεί να εφαρμοσθεί στον άξονα ενός μη διεγερόμενου θηματικού κινητήρα χωρίς να προκαλείται συνεχής περιστροφή του.

Καμπύλη ταχύτητας / ροπής

Η χαρακτηριστική καμπύλη ταχύτητας / ροπής ενός θηματικού κινητήρα εξαρτάται άμεσα από το κύκλωμα οδήγησης, τον τρόπο διεγερσης και την αδράνεια του φορτίου.

Μέγιστη συχνότητα εκκίνησης

Η μέγιστη συχνότητα (ρυθμός επανάληψης) των παλμών που επιτρέπουν σε ένα αφόρτιστο κινητήρα να εκκινεί την περιστροφή και να συνεχίζει να στρέφεται χωρίς να 'χάνει' βήματα ή να σταματά χωρίς να κάνει περισσότερα βήματα από πλήθος των παλμών οδήγησης.

Ροπή pullout (ανατροπής)

Η μέγιστη ροπή που μπορεί να εφαρμοστεί στον άξονα ενός θηματικού κινητήρα που στρέφεται με σταθερή ταχύτητα χωρίς να προκαλείται απώλεια βημάτων.

Ροπή pull in (εκκίνησης)

Η μέγιστη ροπή για την οποία ένας θηματικός κινητήρας μπορεί να ξεκινήσει, να σταματήσει ή να αλλάξει φορά περιστροφής χωρίς να χάσει κάποιο βήμα.

Ακρίβεια

Η διαφορά μεταξύ της πραγματικής θέσης του άξονα σε σχέση με τη θεωρητική. Εκφράζεται σε ποσοστιαία αναλογία (τυπική τιμή +/- 5%) και δεν είναι αθροιστική.

Σφάλμα υστέρησης

Πρόκειται για το μέγιστο αθροιστικό σφάλμα της θεωρητικής από την πραγματική θέση όταν ο άξονας στρέφεται προς τη μια και την άλλη κατεύθυνση.

Συντονισμός

Οι θηματικοί κινητήρες έχουν κατασκευαστεί ώστε να στρέφουν τον άξονα τους κατά μια ορισμένη γωνία κάθε φορά που διεγείρονται από έναν παλμό. Μέσα σε αυτό το διάστημα πρέπει να εκκινήσουν την περιστροφή του άξονα, να τον οδηγήσουν στην νέα του θέση, να τον επιβραδύνουν και, τέλος, να τον σταματήσουν. Το σύνολο αυτών των ενεργειών προκαλεί ταλαντώσεις, υπερβάσεις της τελικής θέσης και δονήσεις. Ακόμα, το πιο πιθανό είναι πως για μερικές συχνότητες παλμών οδήγησης ο κινητήρας παραμένει αδρανής! Οι συχνότητες αυτές ονομάζονται συχνότητες συντονισμού. Οι σχεδιαστές φροντίζουν να σχεδιάζουν τα προϊόντα τους με τέτοιο τρόπο ώστε να ελαχιστοποιείται η πιθανότητα

Μικροβηματισμός

Εδώ τα πράγματα είναι λίγο διαφορετικά. Το ρεύμα των τυλιγμάτων μεταβάλλεται συνεχώς αναγκάζοντας τον άξονα του κινητήρα να στρέφεται κατά συγκεκριμένα κλάσματα της γωνίας που αντιστοιχεί στο πλήρες βήμα. Λόγω της ιδιόζουσας φύσης αυτού του τρόπου θα αποφύγουμε να δώσουμε περισσότερες λεπτομέρειες.

Τεχνικά στοιχεία και ορολογία

Ροπή συγκράτησης

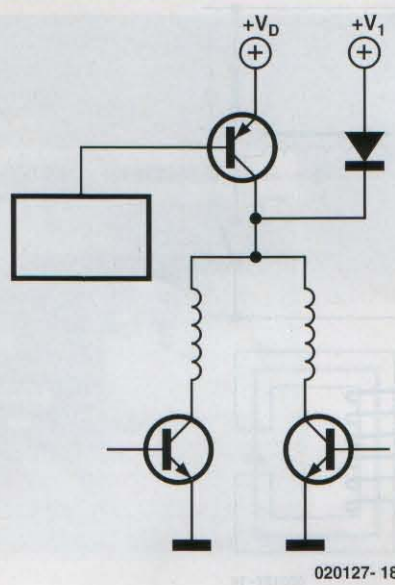
Η μέγιστη σταθερή ροπή που μπορεί να εφαρμοσθεί στον άξονα ενός διεγερόμενου θηματικού κινητήρα χωρίς να προκαλείται η συνεχής περιστροφή του.

τα ύπαρξης τέτοιων συχνοτήτων μέσα στα προβλεπόμενα όρια λειτουργίας του κινητήρα. Ένας τρόπος για να το πετύχουν είναι η προσθήκη μηχανικών συστημάτων απόσβεσης ή η οδήγησή τους με κατάλληλα ηλεκτρονικά κυκλώματα.

Η οδήγηση των βηματικών κινητήρων

Στο δεύτερο μέρος της σειράς που θα δημοσιευτεί τον επόμενο μήνα, παρουσιάζουμε ένα πολυκαναλικό κύκλωμα ισχύος ικανό να οδηγήσει βηματικούς κινητήρες μεγάλης ροπής. Επειδή όμως το κύκλωμα αυτό έχει ορισμένες ιδιομορφίες, που για να γίνουν αντιληπτές, απαιτούνται επιπλέον γνώσεις, προχωρούμε στη διατύπωση μερικών ακόμα ερμηνειών και λειτουργιών. Ένα τυπικό κύκλωμα οδήγησης βηματικών κινητήρων καλείται να εκτελεί αδιάκοπα δύο διαφορετικές εργασίες:

- να τροποποιεί τη μαγνητική ροή μέσα από μια φάση ελέγχοντας τη φορά κίνησης του ρεύματος ή τη διαδρομή που ακολουθεί, και
- να ελέγχει και να περιορίζει την ένταση του ρεύματος μέσα από τα τυλίγματα έτσι ώστε οι χρόνοι ανύψωσης και πτώσης να είναι όσο το δυνατόν μικρότεροι. Με αυτόν τον τρόπο επιτυγχάνεται η καλύτερη δυνατή από

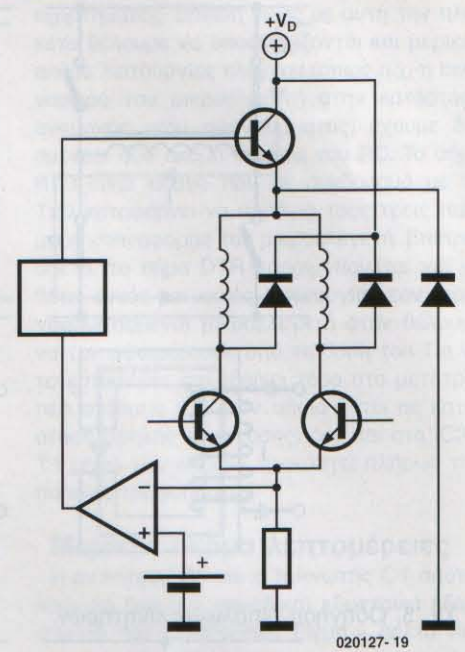


Σχ. 8. Περιορισμός ρεύματος δύο επιπέδων.

δοση του κινητήρα.

Έλεγχος φοράς της μαγνητικής ροής

Για να μπορεί ένας κινητήρας να στρέφεται συνεχώς (κάνοντας μικρά θήματα) είναι απαραίτητος ο έλεγχος της μαγνητικής ροής σε κάθε μια φάση ξεχωριστά. Ο έλεγχος αυτός επιτυγχάνεται επεμβαίνοντας στη φορά του



Σχ. 9. Αρχή λειτουργίας του κυκλώματος κερματισμού.

ρεύματος που διεγείρει το τυλίγμα της κάθε φάσης. Υπάρχουν δύο τρόποι επέμβασης, ανάλογα με το αν ο κινητήρας είναι διπολικός ή μονοπολικός.

Διπολικοί κινητήρες

Σε έναν τέτοιο κινητήρα η φορά κίνησης

Μέθοδος οδήγησης	Κόστος / Αριθμός εξαρτημάτων	Στάθμη ηλεκτρικού θορύβου	Ροπή / Ταχύτητα / Διακριτική ικανότητα	Απόδοση ισχύος	Σχεδιαστικές υποδείξεις	Εφαρμογές
1 Μονοπολική Σταθερής Τάσης	Χαμηλό, 4 τρανζίστορ ισχύος για κάθε κινητήρα.	Χαμηλή	Η ροπή συγκράτησης εξαρτάται από την τάση τροφοδοσίας και τη θερμοκρασία του κινητήρα. Μεγάλη κυμάτωση ροπής στον τρόπο λειτουργίας ημίσεως βήματος.	Ελάχιστη ισχύς εξόδου Μέγιστη κατανάλωση ισχύος στην ακινησία.	Για τρανζίστορ ισχύος πρέπει να αντέχουν σε τάση διπλάσια της τροφοδοσίας. Η σχεδίαση του τροφοδοτικού πρέπει να είναι σύμφωνα με τα χαρακτηριστικά των.	Σε μικρούς κινητήρες χαμηλής ισχύος / ταχύτητας.
2 Μονοπολική Σταθερής Τάσης L/R	Χαμηλό, 4 τρανζίστορ ισχύος για κάθε κινητήρα.	Χαμηλή	Η ροπή συγκράτησης εξαρτάται από την τάση τροφοδοσίας και τη θερμοκρασία του κινητήρα. Μεγάλη κυμάτωση ροπής στον τρόπο λειτουργίας ημίσεως βήματος.	Πολύ μικρή απόδοση. Μέγιστη κατανάλωση ισχύος στην ακινησία αυξανόμενη ανάλογα με το λόγο L/R.	Προβλήματα από τη θέρμανση των αντιστάσεων περιορισμού ρεύματος.	Σε κινητήρες χαμηλής / μεσαίας ταχύτητας και ισχύος.
3 Μονοπολική δύο επιπέδων με χρονιστή	Μέσο, 4 τρανζίστορ ισχύος για κάθε κινητήρα. Απαιτεί μικροελεγκτή ή κύκλωμα χρονισμού.	Μεσαία	Η ροπή συγκράτησης εξαρτάται από την τάση τροφοδοσίας και τη θερμοκρασία του κινητήρα.	Μεσαία απόδοση.	Αν ο λόγος των δύο τάσεων τροφοδοσίας είναι πολύ μεγάλος παρατηρούνται προβλήματα στον έλεγχο της ροπής και της ακρίβειας βηματισμού.	Σε κινητήρες χαμηλής / μεσαίας ταχύτητας και ισχύος.
4 Μονοπολική Σταθερού Ρεύματος	Μέσο / Υψηλό, 6 τρανζίστορ ισχύος για κάθε κινητήρα.	Μεσαία / Υψηλή	Η βέλτιστη ροπή αποδίδεται σε μεγάλη ταχύτητα. Στις χαμηλές ταχύτητες αποδίδουν το 70% της βέλτιστης. Πλήρης έλλειψη κυμάτωσης ροπής στον τρόπο λειτουργίας.	Μεσαία / Υψηλή απόδοση	Τα τρανζίστορ ισχύος πρέπει να αντέχουν σε τάση διπλάσια της τροφοδοσίας. Πρέπει να λαμβάνεται υπόψη η αυτεπαγωγή διαρροής των τυλιγμάτων.	Σε κινητήρες υψηλής ταχύτητας και μεσαίας ισχύος.
5 Διπολική Σταθερού Ρεύματος	Υψηλό, 8 τρανζίστορ ισχύος για κάθε κινητήρα. Διατίθενται ειδικά ολοκληρωμένα οδηγίες για κινητήρες χαμηλής / μεσαίας ισχύος.	Υψηλή	Βέλτιστη ροπή τόσο στις χαμηλές όσο και στις υψηλές ταχύτητες.	Υψηλή απόδοση. Μέγιστη ισχύς εξόδου. Απώλειες ισχύος στις αντιστάσεις αίσθησης ρεύματος.	Προβλήματα με τον ηλεκτρικό θόρυβο. Είναι βέβαιη η ανάπτυξη EMI.	Σε κινητήρες μεγάλης ταχύτητας και μεγάλης ισχύος.
6 Διπολική Σταθερού Ρεύματος με μικροβηματισμό	Πολύ υψηλό, 8 τρανζίστορ ισχύος για κάθε κινητήρα. Υψηλό κόστος κυκλωμάτων ελέγχου. Διατίθενται ειδικά ολοκληρωμένα οδηγίες για κινητήρες χαμηλής /	Υψηλή	Βέλτιστη ροπή τόσο στις χαμηλές όσο και στις υψηλές ταχύτητες. Απουσία συντονισμών στις χαμηλές ταχύτητες. Αυξημένη διακριτική ικανότητα.	Υψηλή απόδοση. Μέγιστη ισχύς εξόδου. Απώλειες ισχύος στις αντιστάσεις αίσθησης ρεύματος.	Προβλήματα με τον ηλεκτρικό θόρυβο. Είναι βέβαιη η ανάπτυξη EMI.	Σε κινητήρες μεγάλης ταχύτητας και μεγάλης ισχύος. Χρησιμοποιούνται εκεί όπου απαιτείται λειτουργία χωρίς συντονισμούς και μεγάλη διακριτική ικανότητα.

του ρεύματος αλλάζει κάθε φορά που αλλάζει και η πολικότητα της επιβαλλόμενης τάσης στο τυλίγμα της φάσης. Για να γίνει δυνατή μια τέτοια αλλαγή απαιτούνται τέσσερις (ηλεκτρονικοί) διακόπτες για κάθε φάση. Στο σχ. 5 φαίνεται η συνδεσμολογία τους.

Μονοπολικό κινητήριο

Οι κινητήρες αυτού του τύπου διαθέτουν ένα τυλίγμα με μεσαία λήψη. Τροφοδοτώντας με ρεύμα την λήψη και κατευθύνοντας το ρεύμα προς την μια ή την άλλη πλευρά του τυλίγματος καταφέρνουμε να αντιστρέψουμε τη φορά της μαγνητικής ροής. Σε μια τέτοια συνδεσμολογία απαιτούνται δύο μόνο (ηλεκτρονικοί) διακόπτες, συνδεδεμένοι όπως φαίνεται στο σχ. 6.

Έλεγχος ρεύματος

Ο έλεγχος του ρεύματος τροφοδοσίας του θηματικού κινητήρα αποτελεί αντικείμενο πρωταρχικής σημασίας, αν σκεφθούμε ότι μέσω αυτού καταφέρνουμε να μειώνουμε αισθητά τις απώλειες ισχύος στα τυλίγματα όπως επίσης και να προσδιορίζουμε με ακρίβεια την αναπτυσσόμενη ροπή. Για να κάνουμε πράξη τα παραπάνω χρησιμοποιούμε δύο διαφορετικές κατηγορίες κυκλωμάτων: αυτά που περιορίζουν το ρεύμα στις επιθυμητές τιμές και εκείνα που βασίζονται στον κερματισμό. Και τα δύο εργάζονται εξ ίσου καλά

τόσο σε πολιτικούς όσο και σε μονοπολικούς κινητήρες.

Περιορισμός ρεύματος μέσω αντίστασης

Τα κυκλώματα που βασίζονται σε αυτήν τη μέθοδο ελέγχου έχουν συνήθως σε μια αντίσταση τοποθετημένη σε σειρά με τα τυλίγματα του κινητήρα (σχ. 7). Η αντίσταση αυτή, εκτός του ότι προκαλεί πτώση τάσης, δρα ως αντίσταση περιορισμού υποβιβάζοντας το ρεύμα στην επιθυμητή τιμή. Το μειονέκτημα αυτής της μεθόδου εντοπίζεται στην άσκοπη κατανάλωση ισχύος στην αντίσταση.

Περιορισμός μέσω δύο πηγών

Αν τροφοδοτήσουμε τα τυλίγματα του κινητήρα μέσω δύο διαφορετικών πηγών τάσης (σχ. 8) είναι δυνατόν να εξασφαλίσουμε υψηλά ρεύματα εκκίνησης τα οποία κατόπιν περιορίζονται στις ονομαστικές τιμές τους. Το τρανζίστορ PNP γίνεται αγώγιμο για ένα προκαθορισμένο χρονικό διάστημα κατά την έναρξη της διέγερσης, ενώ κατόπιν οδηγείται στην αποκοπή. Η τροφοδοσία για το υπόλοιπο του κύκλου πραγματοποιείται από τη δεύτερη πηγή μέσω της διόδου. Το μειονέκτημα αυτής της μεθόδου δεν είναι άλλο από το κόστος της δεύτερης πηγής τροφοδοσίας.

Κερματισμός

Τα κυκλώματα κερματισμού αποτελούν σή-

μερα την καλύτερη λύση για τους θηματικούς κινητήρες, αφού με τη βοήθειά τους εξασφαλίζονται τόσο τα ισχυρά ρεύματα εκκίνησης όσο και η σταθερή τάση τροφοδοσίας τους. Η αρχή λειτουργίας των κυκλωμάτων αυτών είναι μάλλον απλή. Κατ' αρχήν απαιτείται μια πηγή τροφοδοσίας με τάση εξόδου πολλαπλάσια από την ονομαστική τάση λειτουργίας του κινητήρα. Το τρανζίστορ που μεσολαθεί μεταξύ της εξόδου του τροφοδοτικού και του τυλίγματος ανοιγοκλείνει με μεταβλητό λόγο διάρκειας προς περίοδο (Duty Cycle) έτσι ώστε το τυλίγμα να 'νοιωθεί' πάντα την ονομαστική τάση λειτουργίας του. Για τον έλεγχο του ρεύματος χρησιμοποιείται ένας αισθητήρας βασισμένος στην πτώση τάσης που αναπτύσσεται στα άκρα μιας αντίστασης τοποθετημένης σε σειρά με το τυλίγμα. Η τάση αυτή αποτελεί την ανάδραση για το κύκλωμα ελέγχου της τροφοδοσίας. Σαν μειονέκτημα αυτής της μεθόδου θα μπορούσαμε να θεωρήσουμε την απώλεια ισχύος στην παραπάνω αντίσταση.

(020127-1)

Στο επόμενο τεύχος:

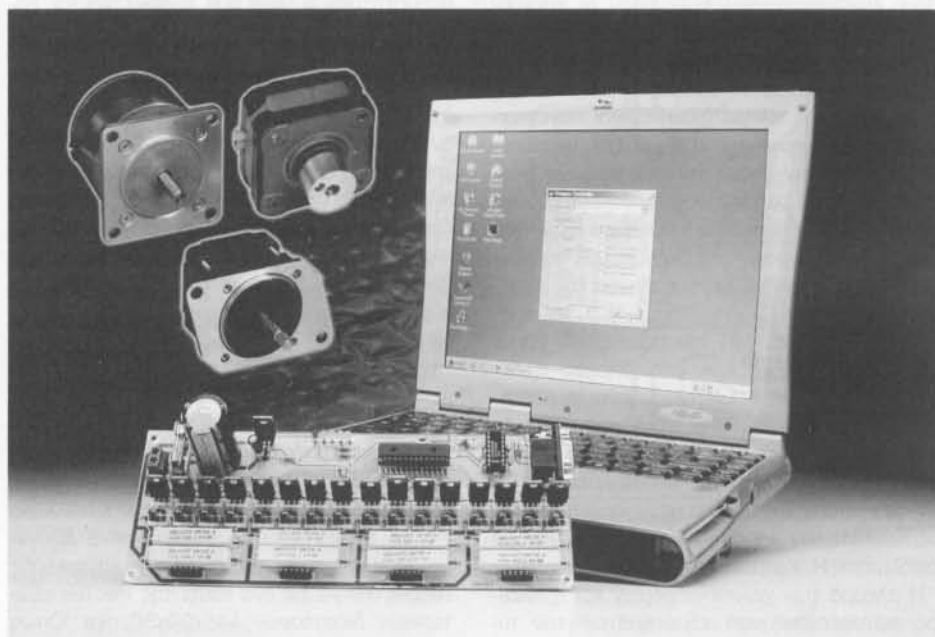
θα αναφερθούμε στο κύκλωμα ισχύος που συνδέεται σε οποιοδήποτε PC. Η κατασκευή βασίζεται σε μικροελεγκτή PIC και συνοδεύεται από το κατάλληλο λογισμικό υποστήριξης.

Τα πάντα για τους βηματικούς κινητήρες (2)

μέρος 2 (τελευταίο): ένας 4-κάναθος, μονοπολικός οδηγός βηματικών κινητήρων

Σχεδίαση Τ. Γ. Κωσταντίνου

Έχοντας πλέον καλύψει όλες τις βασικές αρχές των βηματικών κινητήρων και των συστημάτων οδήγησης, στο δεύτερο και τελευταίο μέρος του αφιερώματος ασχολούμαστε με μία απλή σχεδίαση ενός τετρακάναθου μονοπολικού οδηγού βηματικών κινητήρων, ο οποίος μάλιστα διαθέτει και μία πλήρη βαθμίδα διασύνδεσης για χειρισμό μέσω ενός κοινού προσωπικού υπολογιστή.



Στο δεύτερο λοιπόν μέρος του άρθρου περιλαμβάνονται όλες οι απαραίτητες λεπτομέρειες για την κατασκευή, δοκιμή και χρήση μίας 4-κάναθης βαθμίδας οδήγησης βηματικών κινητήρων, η οποία μάλιστα είναι σε θέση να προσαρμοστεί επί ακριβώς στις ανάγκες της κάθε εφαρμογής. Η συγκεκριμένη κατασκευή περιλαμβάνει μία διασύνδεση RS-232 για άμεση σύνδεση με τον υπολογιστή, μία γλώσσα υψηλού επιπέδου για εκτέλεση των εντολών που απευθύνονται προς τον ελεγκτή και τα ηλεκτρονικά μέρη οδήγησης που τροφοδοτούν τους κινητήρες. Εκτός από αυτά, το λογισμικό επικοινωνίας με τον υπολογι-

στή εξηγείται αρκετά αναλυτικά, επιτρέποντας με τον τρόπο αυτό την προσαρμογή του στις απαιτήσεις του καθενός. Το εν λόγω λογισμικό είναι συμβατό με όλες τις εκδόσεις του λειτουργικού συστήματος των windows (32-bit) και αναπτύχθηκε με την βοήθεια της Delphi Borland.

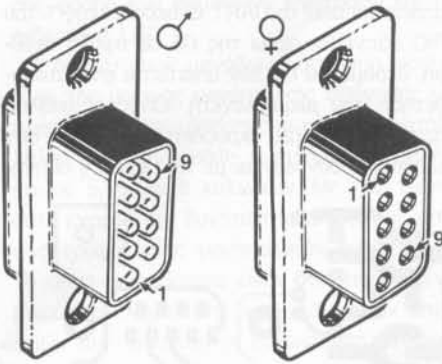
Η σειριακή διασύνδεση RS-232

Η γνωστή σε όλους μας σειριακή διασύνδεση RS-232 η οποία προδιεγράφη κάπου τέσσερις δεκαετίες νωρίτερα, εξακολουθεί να αποτελεί την πρώτη επιλογή για επικοινωνίες με τον υπολογιστή σε χαμηλό εύρος

ζώνης. Εάν λάβουμε μάλιστα υπ' όψη ότι σχεδόν όλοι οι σημερινοί προσωπικοί υπολογιστές διαθέτουν τουλάχιστον μία θύρα RS-232, καθώς επίσης και ότι πολλοί από του σημερινούς μικροελεγκτές διαθέτουν (ή μπορούν εύκολα να αποκτήσουν) ένα κύκλωμα σειριακής διασύνδεσης, τότε η θύρα RS-232 αποτελεί σίγουρα μία φτηνή αλλά και εύκολη επιλογή για ιδιο-κατασκευές.

Καταλήγοντας συνήθως σε ένα 9-πινο βύσμα τύπου sub-D 9 ακίδων το οποίο αναφέρεται ως COM1 ή COM2, η σειριακή διασύνδεση RS-232 διαθέτει εννέα επαφές και μέσω αυτής είναι δυνατόν να επιτευχθεί ημι-αμφίδρομη επικοινωνία με χρήση τριών μόνων από τους αντίστοιχους ακροδέκτες (2, 3 και 5). Αναλυτικά, η διάταξη των ακροδεκτών περιγράφεται στο Σχήμα 1. Σε αντίθεση με τις τυπικές στάθμες TTL, τα δεδομένα στην RS-232 οδεύουν διπολικά, χρησιμοποιώντας +3 έως +25 V για την αναπαράσταση του λογικού '0' και -3 έως -25 V για την αναπαράσταση του λογικού '1'. Με τον τρόπο αυτό καθίσταται δυνατή η επικοινωνία σε σχετικά μεγάλη απόσταση, με το κόστος βέβαια ότι απαιτούνται επιπρόσθετα ηλεκτρονικά μέρη για την μετατροπή των τάσεων από στάθμες RS-232 σε στάθμες TTL.

Από την πλευρά του υπολογιστή, η επικοινωνία με οποιαδήποτε σχεδόν θύρα σε επίπεδο υλικού είναι περίπου ίδια με το να προσπελάσουμε τα δεδομένα σε ένα αρχείο αποθηκευμένο σε κάποιο δίσκο. Η κάθε θύρα διαθέτει μία μοναδική διεύθυνση και για να ανακτήσουμε τα δεδομένα ανοίγουμε την θύρα, τα μεταφέρουμε και στην συνέχεια την κλεί-



020127 - 13

Σχήμα 1. Περιγραφή των ακροδεκτών της RS232.

νομε. Η μόνη επιπρόσθετη ενέργεια που απαιτείται πριν την χρήση είναι ο καθορισμός των παραμέτρων της θύρας (όπως για παράδειγμα ο ρυθμός μεταφοράς δεδομένων, το ψηφίο ισοτιμίας κ.λ.π.)

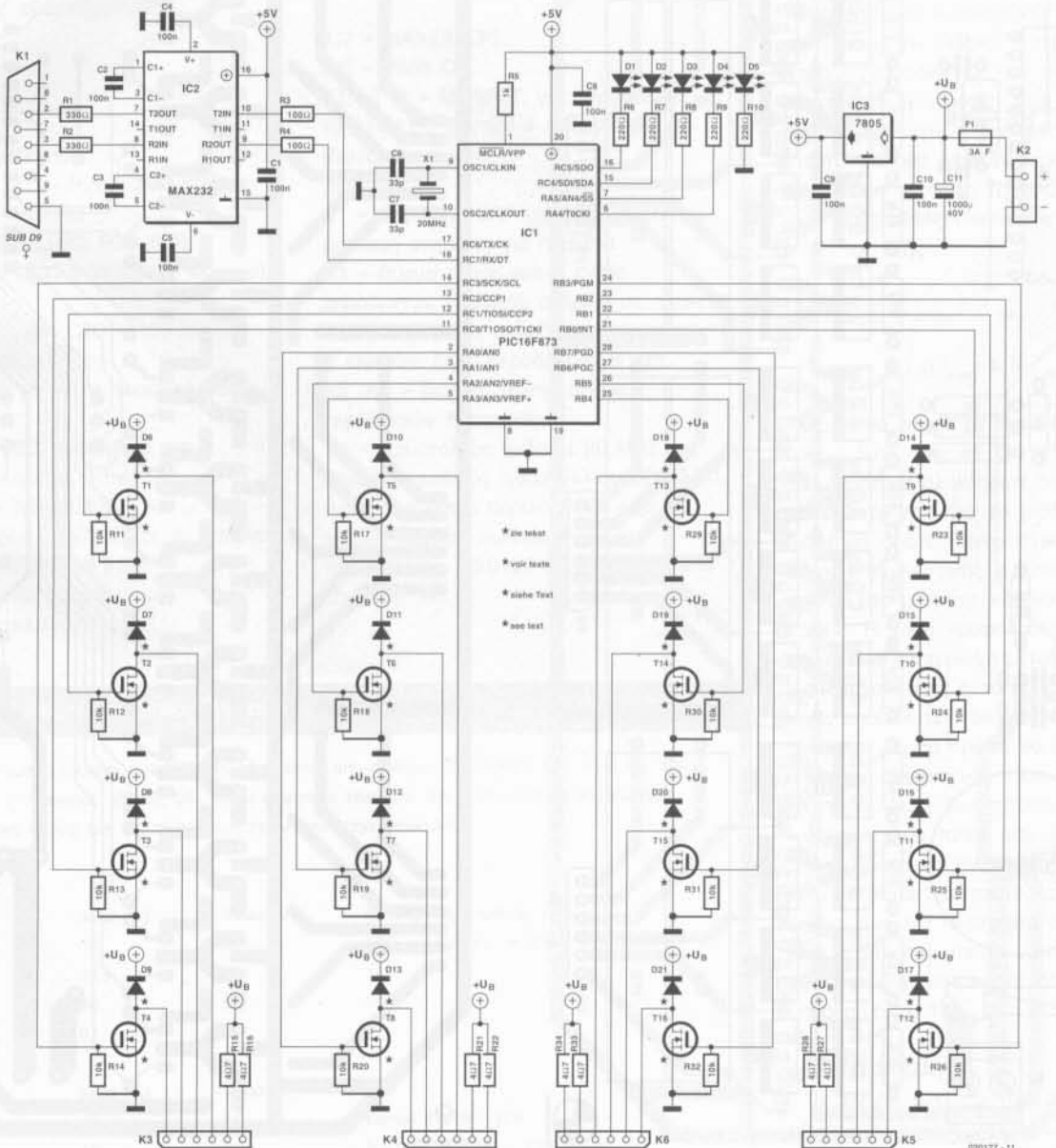
Ακροδέκτης	Σήμα
1	Ανίχνευση φέροντος (DCD)
2	Λήψη δεδομένων (RxD)
3	Εκπομπή δεδομένων (TxD)
4	Τερματικό δεδομένων έτοιμο (DTR)
5	Γείωση σήματος (SG)
6	Σύνολο δεδομένων έτοιμο (DSR)
7	Αίτηση προς αποστολή (RTS)
8	Ελεύθερος προς αποστολή (CTS)
9	Ένδειξη κλήσης (RI)

Το υλικό

Στο Σχήμα 2 απεικονίζεται το κυκλωματικό διάγραμμα της μονάδας οδήγησης θηματικών κινητήρων και της διασύνδεσης, απ' όπου και μπορούμε να διαπιστώσουμε ότι είναι αρκετά

απλό. Ξεκινώντας από την είσοδο RS232 (K1), συνεχίζουμε με τις γραμμές εκπομπής (Tx) και λήψης (Rx) οι οποίες συνδέονται σε ένα ολοκληρωμένο μετατροπής στάθμης (IC2). Όπως αναφέραμε και ενωρίτερα, το συγκεκριμένο ολοκληρωμένο αναλαμβάνει την μετατροπή των διπολικών τάσεων της RS232 (με διακύμανση για παράδειγμα μεταξύ -9 V / +9 V) σε στάθμες TTL (οι οποίες συνήθως ορίζονται στα +5 V και 0 V). Να σημειώσουμε ότι η διαδικασία αυτή υλοποιείται στο εσωτερικό του ολοκληρωμένου με την τεχνική μεταγωγής πυκνωτών, οι οποίοι δημιουργούν μία υψηλότερη διπολική τροφοδοσία (±9 V).

Στην συνέχεια, τα σήματα που έχουν πλέον μετατραπεί σε στάθμη TTL συνδέονται στους ακροδέκτες UART (Universal Asynchronous Receive and Transmit) του μικροελεγκτή PIC (IC1). Οι ακροδέκτες I/O της RS232 συνδέονται στην είσοδο του μετατροπέα μέσω των



020127 - 11

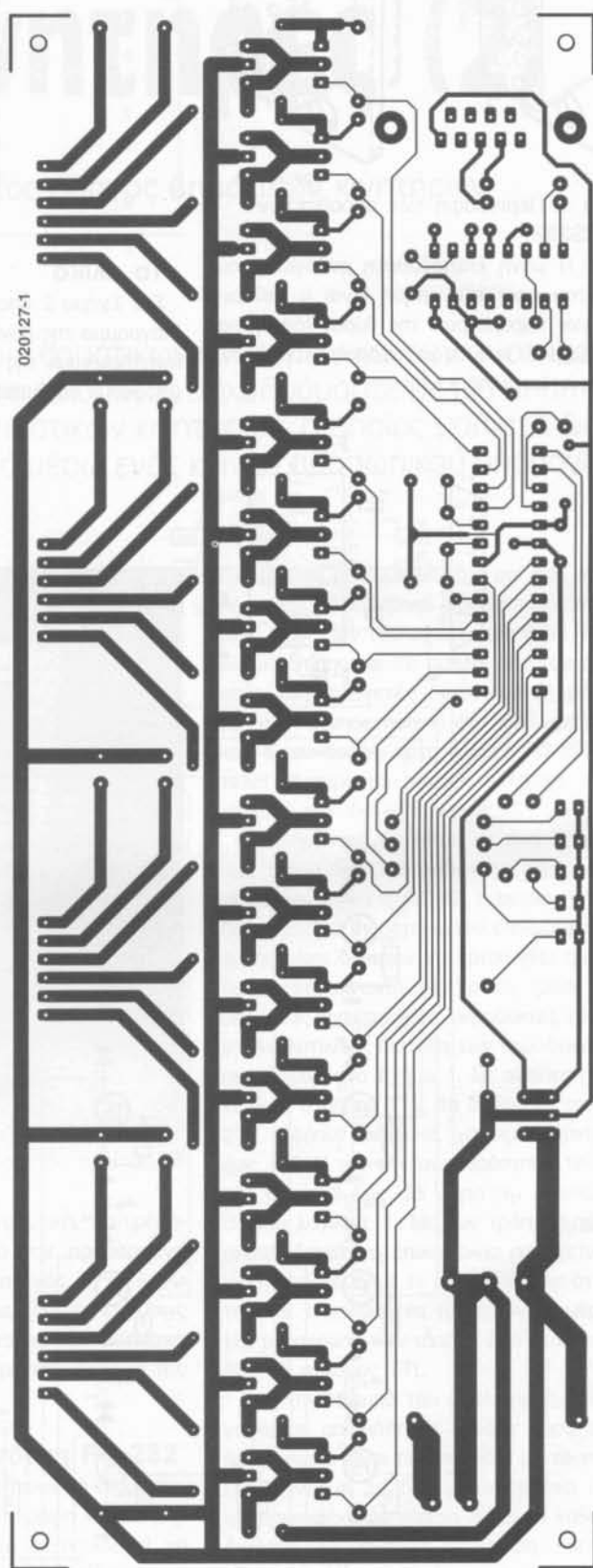
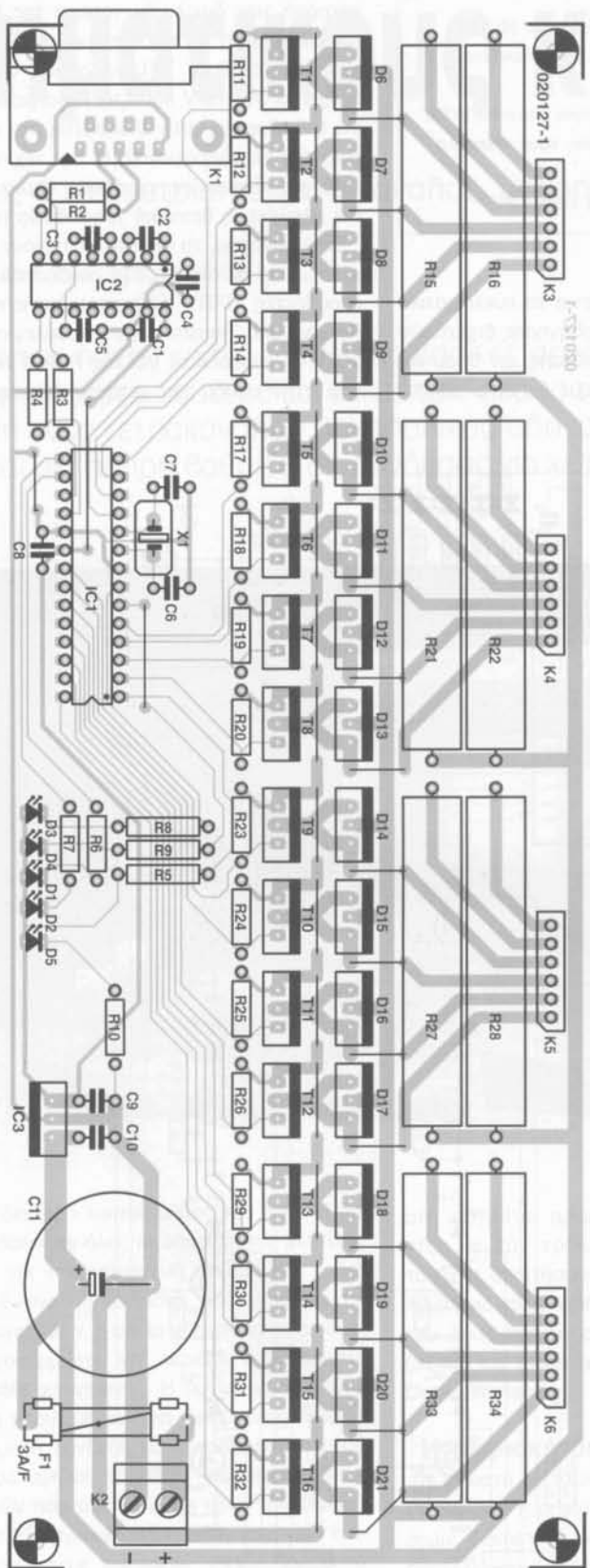
Σχήμα 2. Κυκλωματικό διάγραμμα της μονάδας οδήγησης.

αντιστάσεων σειράς R1 και R2 και συνεχίζουν μέσω των αντιστάσεων R3 και R4 στην έξοδο, κύρια για λόγους προστασίας στην περίπτωση που κάτι πάει στραβά!

Ο γραμμικός σταθεροποιητής (IC3) χρησιμοποιείται για να παρέχει μία σταθεροποιη-

μένη τάση +5 V στον μικροελεγκτή και τον μετατροπέα στάθμης της RS232. Το IC1 χρησιμοποιεί τους πυκνωτές C6, C7 και τον κρυσταλλο χαλαζία X1 σε συνδυασμό με ένα εσωτερικό διαταθήμετο πολυδονητή, για να σχηματίσει ένα ταλαντωτή ακριβείας στα 20 MHz που

απαιτείται από το UART. Ο ακροδέκτης 1 του PIC οδηγείται μέσω της R5 σε υψηλή στάθμη, δεδομένου ότι δεν απαιτείται επανατοποθέτηση του μικροελεγκτή. Όλες οι υπόλοιπες θύρες I/O (20 ακροδέκτες) ορίζονται σαν έξοδοι και συνδέονται με τις μονάδες οδηγη-



Σχήμα 3. Η πλακέτα για την μονάδα οδήγησης βηματικών κινητήρων.

ως φάση των θηματικών κινητήρων και τα ενδεικτικά LED.

Η διάταξη των μονάδων οδήγησης φάσης είναι της μορφής μονοπολικής οδήγησης με περιορισμό μέσω αντίστασης και είναι κατάλληλη για θηματικούς κινητήρες χαμηλής ισχύος 5, 6 και 8 καλωδίων. Με τον τρόπο αυτό έχουμε την δυνατότητα απλής και φτηνής τροφοδοσίας μονοπολικών τυλιγμάτων. Το μόνο μειονέκτημα είναι ότι υπάρχει ένα μικρό πρόβλημα στην ισχύ, λόγω των απωλειών που εισάγουν οι ρυθμιστικές αντιστάσεις.

Για την τροφοδοσία των τυλιγμάτων των θηματικών κινητήρων, το κύκλωμα οδήγησης χρησιμοποιεί μονάδες σε λογική στάθμη MOSFET, οι οποίες και οδηγούνται απ' ευθείας από την έξοδο του μικροελεγκτή. Στην συγκεκριμένη περίπτωση είναι δυνατόν να

χρησιμοποιηθούν διάφορα FET και για τον λόγο αυτό παραθέτουμε και ένα αντίστοιχο πίνακα. Για την προστασία των MOSFET από τάσεις προερχόμενες από εκφόρτιση των πηνίων των κινητήρων, απαιτούνται δύο διόδους γρήγορης επαναφοράς μέσω των οποίων θα διαχετεύεται η αποθηκευμένη στα πηνία ενέργεια. Και στην περίπτωση αυτή, υπάρχει ένα ικανό πλήθος διόδων κατάλληλων για το κύκλωμα, ορισμένες από τις οποίες αναφέρονται στον ίδιο με τα MOSFET πίνακα. Οι ρυθμιστικές αντιστάσεις χρησιμοποιούνται για τον περιορισμό του ρεύματος που οδηγείται στα πηνία και αναπόφευκτα βέβαια καταναλώνουν ένα τμήμα της παρεχόμενης ισχύος. Οι τιμές των αντιστάσεων αυτών θα πρέπει να ανταποκρίνονται στον συγκεκριμένο θηματικό κινητήρα που χρησιμοποιείται. Είναι πολύ σημαντικό να διαθέτουμε τα κατάλληλα δεδο

Πίνακας 1 Τρεις ρυθμιστικών αντιστάσεων (παράδειγμα)

In this circuit, the choice of logic level FET (positions T1-T16) and fast recovery diodes (positions D6-D21) will be governed by availability and the power rating of the stepper motor(s) used.

FETs				
Type	I _{max} (A)	V _{max} (V)	R _i (mΩ)	Note
RFD14N05L	14	50	100	Farnell # 515-399, Fairchild
BUK100-50GL	13.5	50	125	
BUK101-50GS	30	50	30	
IRL12203N	61	30	7	
Diodes				
Type	I _{max} (A)	V _{max} (V)		
MBR1045CT	10	45		Farnell # 878-364
MBR1545CT	15	45		Farnell # 878-194
etc.				

μένα από τον κατασκευαστή που αφορούν τόσο την εμπέδηση του πηνίου, όσο και τις ονομαστικές τιμές ρεύματος και τάσης. Στην περίπτωση που δεν διαθέτουμε τα δεδομένα αυτά, δεν συνιστάται η εύρεση απλώς της αντίστασης μέσω ενός πολυμέτρου διότι στην ουσία δεν θα έχουμε καμία γνώση γύρω από τις πραγματικές ονομαστικές τιμές ισχύος του κινητήρα. Στον Πίνακα 1 παρουσιάζεται ένα παράδειγμα επιλογής των απαιτούμενων χαρακτηριστικών (τόσο η τιμή της αντίστασης όσο και της ισχύος) για την ρυθμιστική αντίσταση, για δύο διαφορετικούς κινητήρες με διαφορετικές τάσεις τροφοδοσίας.

Οι τιμές αυτές είναι δυνατόν να υπολογιστούν ως εξής:

$$R_{ballast} = V_{supply} / (I_{motor} - R_{motor})$$

$$R_{ballast} = 0,5 (I_{motor}^2 \times R_{ballast})$$

Ορισμένα σημεία τα οποία αξίζει να σημειώσουμε: λόγω του ότι ο κινητήρας οδηγείται σε κατάσταση ολόκληρων θημάτων, τα πηνία τροφοδοτούνται για το μισό μόνον χρονικό διάστημα, οπότε η ονομαστική ισχύς της αντίστοιχης ρυθμιστικής αντίστασης μπορεί να οριστεί στο μισό της κανονικά αναμενόμενης ισχύος. Η τάση τροφοδοσίας θα πρέπει να επιλεγεί στην περιοχή μεταξύ 10 V και 30 V (όσο μεγαλύτερη η τάση, τόσο περισσότερη ισχύς αποδίδεται στον κινητήρα). Η τάση αυτή εννοείται ότι θα πρέπει να είναι μεγαλύτερη από την ονομαστική τάση λειτουργίας του κινητήρα, διότι δεν πρέπει να λησμονούμε και την πτώση τάσης που υπάρχει στα άκρα της ρυθμιστικής αντίστασης. Ένα άλλο σημείο το οποίο θα πρέπει επίσης να σημειώσουμε, είναι ότι το μέγιστο ονομαστικό ρεύμα (ανά πηνίο) που μπορεί να οδηγηθεί από την συγκεκριμένη κατασκευή δεν θα πρέπει να υπερβαίνει το 1 A.

Κατασκευή

Το σύνολο των εξαρτημάτων του συγκεκριμένου κυκλώματος τοποθετείται επάνω σε μία πλακέτα, της οποίας τόσο η χάλκινη πλευρά

Κατάλογος εξαρτημάτων

Αντιστάσεις:

R1, R2 = 330 Ω

R3, R4 = 100 Ω

R5 = 1KΩ

R6 - R10 = 220 Ω

R11 - R14, R17 - R20, R23 - R26,

R29 - R32 = 10 KΩ

R15, R16, R21, R22, R27, R28,

R33, R34 = 18 Ω 5 W (θλίπε κείμενο)

Πυκνωτές:

C1 - C5, C8, C9, C10 = 100 nF

C6, C7 = 33 pF

C11 = 1000 μF 40 V ηλεκτρολυτικός

Ημιαγωγοί:

D1 - D4 = LED, πράσινο, 3 mm

D5 = LED, κόκκινο, 3 mm

D6 - D21 = MBR2060CT

(Farnell # 247 - 157) (δείτε το κείμενο)

IC1 = PIC16F873-20 / SP

(δεν διατίθεται έτοιμο

προγραμματισμένο)

IC2 = MAX232CPE

IC3 = 7805 CP

T1 - T16 = MOSFET, για παράδειγμα

RFD14N05L (Farnell # 516 - 399)

(δείτε το κείμενο)

Διάφορα:

F1 = ασφάλεια, 3AF (ταχείας τήξεως)

με βάση στήριξης για πλακέτα

K1 = Βύσμα τύπου sub-D εννέα

ακροδεκτών (θηλυκό), στήριξη σε πλακέτα

K2 = Τερματικός ακροδέκτης πλακέτας,

2 επαφών, θήμα ακροδεκτών 5 mm.

K3 -K6 = διάταξη μονής σειράς

ακροδεκτών 6 επαφών.

X1 = Κρύσταλλος χαλαζία 20 MHz

Πλακέτα, κωδικός παραγγελίας 020127-1,

Δισκέτα η οποία περιλαμβάνει όλα τα

αρχεία πηγαίου κώδικα, κωδικός

παραγγελίας 020127-11 ή δωρεάν από

το διαδίκτυο.

FET Λογικής Στάθμης και Δίοδοι Ταχείας Επαναφοράς

Στο παρόν κύκλωμα, η επιλογή των FET που οδηγούνται από στ.Αθμες TIL (θέσεις T1 - T16) και των διόδων ταχείας επαναφοράς (θέσεις D6 - D21) εξαρτάται τόσο από την διαθεσιμότητα όσο και από τα χαρακτηριστικά ισχύος των θηματικών κινητήρων που χρησιμοποιούνται.

FET

Τύπος	I _{max} (A)	V _{max} (V)	R _i (mΩ)	Σημειώσεις
RFD14N05L	14	50	100	Farnell # 516 - 399, Fairchild
BUK100-50GL	13,5	50	125	
BUK101-50GS	30	50	50	
IRL12203N	61	50	7	

Δίοδοι

Τύπος	I _{max} (A)	V _{max} (V)	
MBR1045CT	10	45	Farnell # 878 - 364
MBR1545CT	15	45	Farnell # 878 - 194 κ.λ.π.

Πρόγραμμα 1. Πηγαίος κώδικας για το υλικο-λογισμικό

```
// main.c - Main program code

#include <16f873.h>
#include <ports.h>
#include <protocol.h>
#include <clock=20000000>
#include <rs232(baud=38400, xmit=tx, rcv=rc)>

int astep=1, bstep=1, cstep=1, dstep=1;
long max=800, min=470;

// initialises the ports by defining whether the tri-state buffers should be input or output
void setup_ports(void) { set_tris_a(0x00);set_tris_b(0x00);set_tris_c(0xF0);set_uart_speed(38400); }

// resets one motor to initial state
void reset_motor(int motor) {
    if (motor==1) {output_low(a_1);output_low(a_2);output_low(a_3);output_low(a_4);output_high(led_a);}
    if (motor==2) {output_low(b_1);output_low(b_2);output_low(b_3);output_low(b_4);output_high(led_b);}
    if (motor==3) {output_low(c_1);output_low(c_2);output_low(c_3);output_low(c_4);output_high(led_c);}
    if (motor==4) {output_low(d_1);output_low(d_2);output_low(d_3);output_low(d_4);output_high(led_d);} }

// resets all ports to initial states
void reset_ports(void) { reset_motor(1);reset_motor(2);reset_motor(3);reset_motor(4);putc(ACKNOWLEDGE); }

// creates a delay which constitutes the step pulse duration
void delay_micro(long delay) { long n;for(n=1;n<=delay;n+=3)delay_us(6); }

// changes powered phases according to current step required
void power_motor(int axis, step) {
    if (axis==1) {
        if (step==1) {output_bit(a_1,1);output_bit(a_2,0);output_bit(a_3,0);output_bit(a_4,1);}
        if (step==2) {output_bit(a_1,0);output_bit(a_2,1);output_bit(a_3,0);output_bit(a_4,1);}
        if (step==3) {output_bit(a_1,0);output_bit(a_2,1);output_bit(a_3,1);output_bit(a_4,0);}
        if (step==4) {output_bit(a_1,1);output_bit(a_2,0);output_bit(a_3,1);output_bit(a_4,0);}
        output_low(led_a); }
    if (axis==2) {
        if (step==1) {output_bit(b_1,1);output_bit(b_2,0);output_bit(b_3,0);output_bit(b_4,1);}
        if (step==2) {output_bit(b_1,0);output_bit(b_2,1);output_bit(b_3,0);output_bit(b_4,1);}
        if (step==3) {output_bit(b_1,0);output_bit(b_2,1);output_bit(b_3,1);output_bit(b_4,0);}
        if (step==4) {output_bit(b_1,1);output_bit(b_2,0);output_bit(b_3,1);output_bit(b_4,0);}
        output_low(led_b); }
    if (axis==3) {
        if (step==1) {output_bit(c_1,1);output_bit(c_2,0);output_bit(c_3,0);output_bit(c_4,1);}
        if (step==2) {output_bit(c_1,0);output_bit(c_2,1);output_bit(c_3,0);output_bit(c_4,1);}
        if (step==3) {output_bit(c_1,0);output_bit(c_2,1);output_bit(c_3,1);output_bit(c_4,0);}
        if (step==4) {output_bit(c_1,1);output_bit(c_2,0);output_bit(c_3,1);output_bit(c_4,0);}
        output_low(led_c); }
    if (axis==4) {
        if (step==1) {output_bit(d_1,1);output_bit(d_2,0);output_bit(d_3,0);output_bit(d_4,1);}
        if (step==2) {output_bit(d_1,0);output_bit(d_2,1);output_bit(d_3,0);output_bit(d_4,1);}
        if (step==3) {output_bit(d_1,0);output_bit(d_2,1);output_bit(d_3,1);output_bit(d_4,0);}
        if (step==4) {output_bit(d_1,1);output_bit(d_2,0);output_bit(d_3,1);output_bit(d_4,0);}
        output_low(led_d); } }

// Moves a specified motor by a specified amount of steps in a specified direction.
int move(short direction, long steps, int axis, step) {
    long n, delay, accsteps;
    delay=max; accsteps=max-min;
    for(n=1;n<=steps;n++) {
        if(n<=accsteps)delay--;
        if(steps-n<=accsteps)delay++;
        if(direction==0)step--;else step++;
        if(step==0)step=4;
    }
}
```



```

        if(step==5)step=1;
        power_motor(axis, step); delay_micro(delay); reset_motor(axis); } return(step); }

// Reads in 2 bytes from the UART and returns a 16-bit integer (range 0-65535)
long readlong(void) { return(256*getc() + getc()); }

// Main Program
void main(void) {
    char incomm;
    long steps;
    setup_ports(); reset_ports();
    while(0==0) {
        output_low(led_a); output_low(led_b); output_low(led_c); output_low(led_d);
        incomm=getc();
        output_high(led_a); output_high(led_b); output_high(led_c); output_high(led_d);
        switch(incomm) {
            case RESET:      reset_ports();                break;
            case SETUP_ACC:  min=readlong(); max=readlong(); break;
            case MOVE_A_FW:  steps=readlong(); astep=move(0, steps, 1, astep); break;
            case MOVE_A_RV:  steps=readlong(); astep=move(1, steps, 1, astep); break;
            case MOVE_B_FW:  steps=readlong(); bstep=move(0, steps, 2, bstep); break;
            case MOVE_B_RV:  steps=readlong(); bstep=move(1, steps, 2, bstep); break;
            case MOVE_C_FW:  steps=readlong(); cstep=move(0, steps, 3, cstep); break;
            case MOVE_C_RV:  steps=readlong(); cstep=move(1, steps, 3, cstep); break;
            case MOVE_D_FW:  steps=readlong(); dstep=move(0, steps, 4, dstep); break;
            case MOVE_D_RV:  steps=readlong(); dstep=move(1, steps, 4, dstep); break; } putc(ACKNOWLEDGE); } }

// ports.h - defines pin assignments

#define tx    PIN_C6
#define rc    PIN_C7
#define a_1   PIN_C3
#define a_2   PIN_C2
#define a_3   PIN_C1
#define a_4   PIN_C0
#define b_1   PIN_A0
#define b_2   PIN_A1
#define b_3   PIN_A2
#define b_4   PIN_A3
#define c_1   PIN_B3
#define c_2   PIN_B2
#define c_3   PIN_B1
#define c_4   PIN_B0
#define d_1   PIN_B7
#define d_2   PIN_B6
#define d_3   PIN_B5
#define d_4   PIN_B4
#define led_a PIN_A5
#define led_b PIN_A4
#define led_c PIN_C5
#define led_d PIN_C4

// protocol.h - defines communication protocol

#define RESET          1
#define ACKNOWLEDGE  2
#define SETUP_ACC     10
#define MOVE_A_FW     20
#define MOVE_A_RV     21
#define MOVE_B_FW     22
#define MOVE_B_RV     23
#define MOVE_C_FW     24
#define MOVE_C_RV     25
#define MOVE_D_FW     26
#define MOVE_D_RV     27

```

Πρόγραμμα 2. Δοκιμαστικό πρόγραμμα που εκτελείται στον υπολογιστή

```
unit main;

interface

uses Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs, StdCtrls, QCom32, Buttons, ExtCtrls;

type
  TForm1 = class(TForm)
    QCPort: T_QCom32;
    Commport: TComboBox;
    xclgroup: TRadioGroup;
    setup_acc, move_a_rv, move_a_fw, move_b_rv,    move_b_fw, move_c_rv, move_c_fw, move_d_rv, move_d_fw, reset: TRadioButton;
    parameter1, parameter2: TEdit;
    commportlabel, parameterlabel: TLabel;
    Executebutton: TBitBtn;
    autoreset: TCheckBox;
    procedure CommportChange(Sender: TObject);
    procedure ExecutebuttonClick(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure setup_accClick(Sender: TObject);
    procedure move_a_fwClick(Sender: TObject);
    procedure move_a_rvClick(Sender: TObject);
    procedure move_b_fwClick(Sender: TObject);
    procedure move_b_rvClick(Sender: TObject);
    procedure move_c_fwClick(Sender: TObject);
    procedure move_c_rvClick(Sender: TObject);
    procedure resetClick(Sender: TObject);
    procedure move_d_fwClick(Sender: TObject);
    procedure move_d_rvClick(Sender: TObject);
  private { Private declarations }
  public { Public declarations }
  end;

var Form1: TForm1;

Implementation {$R *.DFM}

  procedure TForm1.resetClick(Sender: TObject); begin parameter1.Enabled := FALSE; parameter2.Enabled := FALSE; end;

  procedure TForm1.setup_accClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := TRUE; end;

  procedure TForm1.move_a_fwClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.move_a_rvClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.move_b_fwClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.move_b_rvClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.move_c_fwClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.move_c_rvClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.move_d_fwClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.move_d_rvClick(Sender: TObject); begin parameter1.Enabled := TRUE; parameter2.Enabled := FALSE; end;

  procedure TForm1.CommportChange(Sender: TObject);
    begin QCPort.Port := Commport.ItemIndex + 1; end;

  procedure TForm1.FormShow(Sender: TObject); begin QCPort.Port := 1; Commport.ItemIndex := 0; end;
```



```

procedure TForm1.ExecutebuttonClick(Sender: TObject);
var
  commandcode : char;
  command      : string;
begin
  Executebutton.Enabled := FALSE;

  if reset.Checked then commandcode := char(1);
  if setup_acc.Checked then commandcode := char(10);

  if move_a_fw.Checked then commandcode := char(20);
  if move_a_rv.Checked then commandcode := char(21);
  if move_b_fw.Checked then commandcode := char(22);
  if move_b_rv.Checked then commandcode := char(23);
  if move_c_fw.Checked then commandcode := char(24);
  if move_c_rv.Checked then commandcode := char(25);
  if move_d_fw.Checked then commandcode := char(26);
  if move_d_rv.Checked then commandcode := char(27);

  QCPort.Open; setlength(command, 1);
  command[1] := commandcode; QCPort.Write(command);

  if (parameter1.enabled) then
  begin setlength(command, 2);
    command[1] := char(strtoint(parameter1.text) div 256);
    command[2] := char(strtoint(parameter1.text) mod 256);
    QCPort.Write(command); end;

  if (parameter2.enabled) then begin
    setlength(command, 2);
    command[1] := char(strtoint(parameter2.text) div 256);
    command[2] := char(strtoint(parameter2.text) mod 256);
    QCPort.Write(command); end;

  while(QCPort.Read = '') do;

  if autoreset.Checked then begin
    setlength(command, 1);
    command[1] := char(1);
    QCPort.Write(command);
    while(QCPort.Read = '') do; end;

  QCPort.Close; Executebutton.Enabled := TRUE;
end;
end.

```

όσο και η διάταξη των εξαρτημάτων απεικονίζεται στο Σχήμα 3. Για τα ολοκληρωμένα IC1 και IC2 καλό είναι να χρησιμοποιηθούν βάσεις DIL, ενώ το IC3 κολλείται απ' ευθείας επάνω στην πλακέτα. Όπως συνήθως και κυρίως για δική μας διευκόλυνση, στερεώνουμε πρώτα τα χαμηλά εξαρτήματα όπως είναι οι συρμάτινες γέφυρες, οι αντιστάσεις οι βάσεις DIL, οι κεραμικοί πυκνωτές κ.λ.π. Ιδιαίτερη προσοχή χρειάζεται όσον αφορά την ορθή πολικότητα, κατά την στερέωση όλων των ημιαγωγών και των ηλεκτρολυτικών πυκνωτών. Οι αντιστάσεις καλό περιορισμού ρεύματος καλό είναι να τοποθετηθούν ελαφρώς ανυψωμένες σε σχέση με την πλακέτα, δεδο-

μένου ότι κατά την λειτουργία θερμαίνονται. Συνιστάται μάλιστα η τοποθέτηση κεραμικών διαχωριστικών τα οποία θα κρατούν τις συγκεκριμένες αντιστάσεις σε απόσταση από την πλακέτα.

Στην περίπτωση που δεν χρειάζονται όλα τα κανάλια της βαθμίδας οδήγησης βηματικών κινητήρων, είναι δυνατή η τοποθέτηση εξαρτημάτων μόνον στα κανάλια τα οποία πρόκειται να χρησιμοποιηθούν.

Αφού ολοκληρωθεί η στερέωση των εξαρτημάτων, μπορούμε να τοποθετήσουμε τα ολοκληρωμένα του μικροελεγκτή PIC και του MAX232 στις αντίστοιχες βάσεις. Όποιος μάλιστα επιθυμεί έχει την δυνατότητα να

προγραμματίσει ο ίδιος το PIC της κατασκευής, με την βοήθεια του πηγαίου κώδικα που διατίθεται με κωδικό αριθμό 020127-11 σε δισκέτα, ή δωρεάν από το διαδίκτυο του δικτυακού τόπου του περιοδικού. Για του πλέον φιλόδοξους αναγνώστες που επιθυμούν να τροποποιήσουν τον κώδικα και να προσθέσουν επί πλέον λειτουργίες, παρατίθεται στην συνέχεια μία πλήρης σύνοψη του κώδικα μαζί με μερικές βασικές οδηγίες. Καλό είναι βεβαίως πριν προχωρήσει κανείς σε τροποποίηση του κώδικα, να δοκιμάσει την κατασκευή με το αρχικό υλικό-λογισμικό (firmware).



Σχήμα 4. Το πρόγραμμα αποστολής 'εντολών' προς τον θηματικό κινητήρα, από την πλευρά του υπολογιστή.

Το λογισμικό του ελεγκτή

Το σύνολο των λειτουργιών του PIC, συνίσταται στην εκτέλεση εντολών τις οποίες λαμβάνει μέσω της διασύνδεσης RS232 από τον υπολογιστή. Αρμοδιότητα του μικροελεγκτή είναι η δημιουργία της ακολουθίας βημάτων τα οποία θα ελέγξουν την ισχύ που αποδίδεται στον κινητήρα. Διαθέτοντας λοιπόν διασύνδεση σε τόσο χαμηλό επίπεδο, ο ακριβής χρονισμός των παλμών είναι κάτι παραπάνω από σίγουρος.

Ένα ερώτημα που λογικά τίθεται, είναι γιατί να κάνουμε τόση φασαρία και να μην ελέγχουμε τους παλμούς απ' ευθείας από τον υπολογιστή. Παρότι ο έλεγχος αυτής της μορφής θα ήταν εφικτός με τα παλιά προγράμματα σε περιβάλλον DOS, δυστυχώς πλέον δεν είναι. Αυτό οφείλεται στον πολύ-διεργασιακό χαρακτήρα των 32-ψηφίων λειτουργικών συστημάτων των Windows, όπου ο χρονικός καταμερισμός του επεξεργαστή αποκλείει την δυνατότητα σταθερού και ακριβούς χρονισμού.

Ο προγραμματισμός του λογισμικού για την κατασκευή έγινε με την βοήθεια ενός σχετικά φθηνού μεταφραστή σε γλώσσα C ο οποίος διατίθεται από την CCS και είναι πλήρως συμβατός με το περιβάλλον MLAB της Microchip. Για όποιον επιθυμεί περισσότερες πληροφορίες γύρω από τον συγκεκριμένο μεταφραστή, να αναφέρουμε ότι στον δικτυακό τόπο της CCS υπάρχει μία ολοκληρωμένη αναφορά στην γλώσσα.

Το σύνολο του κώδικα περιλαμβάνεται σε τρία αρχεία: τα main.c, protocol.h και ports.h. Το κυρίως πρόγραμμα βρίσκεται το main.c, ενώ η εκχώρηση μεταβλητών στους ακροδέκτες γίνεται στο ports.h και το πρωτόκολλο επικοινωνίας καθορίζεται στο protocol.h. Ολόκληρος ο πηγαίος κώδικας που αφορά το υλικο-λογισμικό δίνεται στον Κατάλογο 1.

Το πρωτόκολλο επικοινωνίας που χρησιμοποιεί η παρούσα κατασκευή είναι πολύ απλό. Για κάθε εντολή αποστέλλεται μία τιμή με μέγεθος 1 ψηφιολέξη (byte) και στην περίπτωση όπου η εντολή απαιτεί επιπρόσθετα δεδομένα, αυτά αποστέλλονται στην συνέχεια. Εάν για παράδειγμα θέλουμε να δώσουμε στον μικροελεγκτή την εντολή να προωθήσει τον κινητήρα προς μια κατεύθυνση για 1000 βήματα θα χρειαστούμε τρεις ψηφιολέξεις, όπου η πρώτη προσδιορίζει την εντολή και οι δύο επόμενες καθορίζουν το πλήθος των βημάτων (μέσα στην περιοχή από 0 έως 65535). Το συνολικό μήκος της εκπομπής για κάθε εντολή, εξαρτάται από την πρώτη ψηφιολέξη η οποία προσδιορίζει και την εντολή. Μετά από την εκτέλεση της εντολής ο μικροελεγκτής θα απαντήσει με ένα σήμα 'επιβεβαίωσης λήψης' (acknowledge), ενημερώνοντας με τον τρόπο αυτό τον υπολογιστή ότι είναι έτοιμος να δεχτεί τις επόμενες εντολές.

Το πρώτο άρθρο του κυρίως προγράμματος εκχωρεί σε πρώτη φάση τις αρχικές τιμές και επανατοποθετεί όλες τις θύρες I/O, συμπεριλαμβανομένης και της UART η οποία ορίζεται σε ρυθμό μεταφοράς δεδομένων 38.400 bit/s. Στην συνέχεια το πρόγραμμα εισέρχεται σε ένα αέναο βρόγχο επανάληψης, όπου αναμένει την λήψη μίας ψηφιολέξης από το UART. Με την λήψη μίας τέτοιας ψηφιολέξης το έλεγχος του προγράμματος περνά στο κατάλληλο τμήμα επεξεργασίας εντολών, όπου στην συνέχεια ενδέχεται να γίνει λήψη και κάποιων επί πλέον ψηφιολέξεων από το UART.

Το σύνολο των διαθεσίμων εντολών έχει ως εξής:

RESET (byte 1): επανατοποθετεί όλες τις θύρες I/O.

SETUP_ACC (byte 10): Ακολουθούμενη από άλλες τέσσερις ψηφιολέξεις ορίζει την μέγιστη και ελάχιστη καθυστέρηση βημάτων κατά την κίνηση του θηματικού κινητήρα (και τα δύο μεγέθη είναι ακέραιοι αριθμοί με 16 ψηφία). Κατά την εκτέλεση μίας εντολής MOVE οι καθυστερήσεις μεταξύ των βημάτων θα έχουν στην αρχή την μέγιστη τιμή, η οποία στην συνέχεια θα αρχίσει να ελαττώνεται σταδιακά μέχρι να επιτευχθεί το χαμηλότερο όριο. Όλα τα βήματα που θα ακολουθήσουν στην συνέχεια θα έχουν αυτήν την ελάχιστη καθυστέρηση, ενώ προς το τέλος του κύκλου της εντολής η καθυστέρηση μεταξύ των βημάτων θα ξαναρχίσει να αυξάνει μέχρι να φτάσει και πάλι στην μέγιστη τιμή. Είναι προφανές ότι οι παραπάνω ενέργειες υλοποιούν την απαιτούμενη επιτάχυνση και επιβράδυνση σε κάθε εντολή MOVE.

MOVE_A_FW (byte 20): Ακολουθούμενη από άλλες δύο ψηφιολέξεις (ένας ακέραιος 16 ψηφίων), προσδιορίζει το σύνολο των βη-

μάτων που θα εκτελέσει ο κινητήρας προς την ορθή φορά κίνησης.

MOVE_A_RV (byte 21): Ακολουθούμενη από άλλες δύο ψηφιολέξεις (ένας ακέραιος 16 ψηφίων), προσδιορίζει το σύνολο των βημάτων που θα εκτελέσει ο κινητήρας προς την ανάστροφη φορά κίνησης.

MOVE_B_FW (byte 22)

MOVE_B_RV (byte 23)

MOVE_C_FW (byte 24)

MOVE_C_RV (byte 25)

MOVE_D_FW (byte 26)

MOVE_D_RV (byte 27)

Οι εντολές αυτές είναι ανάλογες των MOVE_A_FW και MOVE_A_RV, με την διαφορά ότι αναφέρονται στους κινητήρες B, C και D αντίστοιχα.

Στην περίπτωση όπου προγραμματίζουμε τον δικό μας μικροελεγκτή PIC, δεν θα πρέπει να λησμονήσουμε να απενεργοποιήσουμε το χαρακτηριστικό DEBUG_MODE. Φροντίζουμε επίσης να είναι ενεργοποιημένη η επιλογή POWER_ON_RESET και απενεργοποιημένα τα χαρακτηριστικά WATCH-DOG_TIMER και BROWN_OUT_DETECT. Ένα άλλο σημείο το οποίο θα πρέπει να φροντίσουμε, είναι η συχνότητα του ρολογιού η οποία θα πρέπει να οριστεί στα 20 MHz.

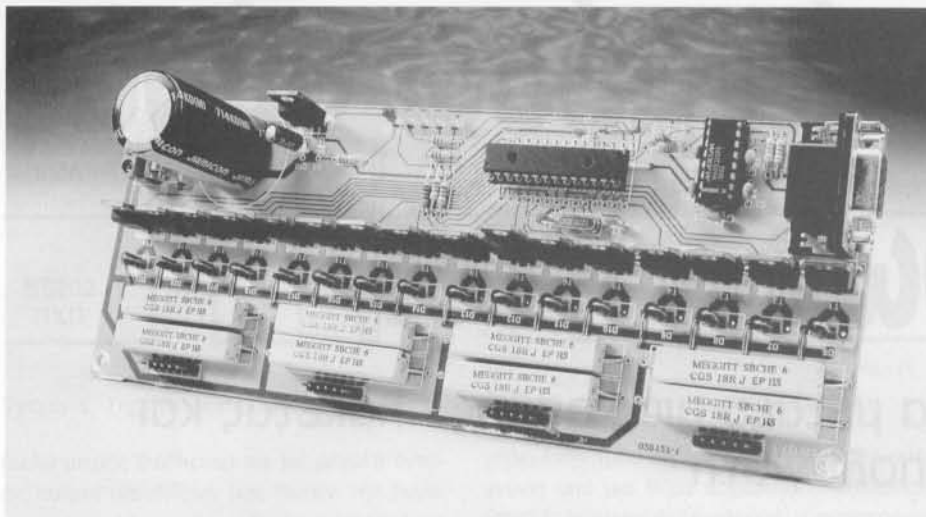
Υπάρχουν διάφορα περιβάλλοντα ανάπτυξης και προγραμματισμού τα οποία θα μπορούσαμε να συστήσουμε, μεταξύ των οποίων έχουμε το Προγραμματιστή PIC του περιοδικού Ελεktor (Οκτώβριος 2003) και την μονάδα PIC-START και ICD της Microchip (η οποία χρειάζεται και μία επί πλέον διάταξη 28 ακροδεκτών). Μία εναλλακτική πρόταση έρχεται από την Taylor Ltd, η οποία διαθέτει ένα αρκετά φθηνό ισοδύναμο του ICD (λιγότερο από 70 Ευρώ), πλήρως συμβατό με το λογισμικό της Microchip το οποίο διατίθεται για κατέβασμα δωρεάν.

Το λογισμικό στον υπολογιστή

Το λογισμικό στην πλευρά του υπολογιστή, προγραμματίστηκε σε Borland Delphi 4.0. Για να έχουμε πρόσβαση στην σειριακή θύρα, χρησιμοποιήσαμε μία 'freeware' έκδοση της VCL (Visual Component Library) η οποία καλείται QCCOM32.

Μέσα στην δεύτερη λίστα (Listing 2) περιλαμβάνεται ένα δοκιμαστικό πρόγραμμα το οποίο περιγράφει την αποστολή των εντολών προς τον ελεγκτή θηματικών κινητήρων μέσω της θύρας RS-232. Και αυτό το αρχείο διατίθεται δωρεάν κάτω από την σελίδα 'Free Downloads' του δικτυακού τόπου του περιοδικού.

Είναι σημαντικό να βεβαιωθούμε ώστε τα χαρακτηριστικά της QCCOM32 να συμφωνούν επ' ακριβώς με τις αρχικές τιμές που εκχωρούνται στο UART από το υλικο-λογι-



Σχήμα 5. Το πρωτότυπο του οδηγού βηματικών κινητήρων, έτοιμο.

σμικού, ειδικά όσον αφορά τον ρυθμό μεταφοράς δεδομένων (bitrate = 38400). Για κάθε εντολή που αποστέλλεται προς τον ελεγκτή υπάρχει μία διαδικασία, όπου η θύρα ανοίγει, εκπέμπονται τα απαιτούμενα δεδομένα, στην συνέχεια το πρόγραμμα αναμένει ένα σήμα επιβεβαίωσης (acknowledge) και τέλος η θύρα κλείνει.

Δοκιμή και χρήση στην πράξη

Πριν τροφοδοτήσουμε το κύκλωμα είναι σημαντικό να βεβαιωθούμε ότι όλα τα εξαρτήματα είναι σωστά κολλημένα και τα σημεία των κολλήσεων είναι καθαρά. Αποσυνδέουμε λοιπόν τους βηματικούς κινητήρες και τροφοδοτούμε το κύκλωμα: αρχικά και με την βοήθεια ενός αμπερομέτρου ελέγχουμε το ρεύμα που αναλώνεται από το κύκλωμα, ενώ στην συνέχεια χρησιμοποιούμε ένα βολτόμετρο για να ελέγξουμε τις γραμμές της τάσης τροφοδοσίας. Στην περίπτωση που οτιδήποτε από τα παραπάνω δείχνει να ξεφεύγει από τα φυσιολογικά όρια, διακόπτουμε άμεσα την τροφοδοσία και ελέγχουμε την πλακέτα και τους συνδέσμους.

Κάτω από κανονικές συνθήκες, την ώρα που τροφοδοτείται το κύκλωμα πρέπει να ανάψουν και τα πέντε LED. Εάν λοιπόν διαπιστώσουμε ότι όντως ανάβουν και τα πέντε LED, τότε ο μικροελεγκτής έχει ενεργοποιηθεί και λειτουργεί σωστά. Στην περίπτωση όμως όπου έχουμε μόνον το ένα LED να ανάβει, τότε το κύκλωμα τροφοδοτείται σωστά αλλά το υλι-

κο-λογισμικό του μικροελεγκτή δεν εκτελείται σωστά. Εάν δεχθούμε ότι ο προγραμματισμός του PIC έχει γίνει σωστά, θα πρέπει να βεβαιωθούμε ότι στους αντίστοιχους ακροδέκτες λαμβάνει την σωστή τροφοδοσία. Εάν και εκεί οι τάσεις δείχνουν να είναι σωστές, αυτό που μένει να ελέγξουμε είναι τα εξωτερικά εξαρτήματα του ταλαντωτή (X1, C6 και C7), για να βεβαιωθούμε ότι έχουν τοποθετηθεί ορθά. Εάν και πάλι δεν έχουμε αποτέλεσμα, θα πρέπει να δοκιμάσουμε να αντικαταστήσουμε ή να επανα-προγραμματίσουμε το ολοκληρωμένο του μικροελεγκτή.

Αφού λοιπόν φτάσουμε στο σημείο όπου το κύκλωμα ξεκινάει κανονικά, παίρνουμε ένα σειριακό καλώδιο με αντιστοιχία ακροδεκτών 1:1 (όχι διασταυρούμενοι), εξοπλισμένο με βύσματα D-9 θηλυκό σε D-9 αρσενικό, με το οποίο συνδέουμε το κύκλωμα με την σειριακή θύρα του υπολογιστή. Τρέχουμε στον υπολογιστή το δοκιμαστικό πρόγραμμα επιλέγουμε την σωστή ρύθμιση όσον αφορά την θύρα COM και στην συνέχεια δοκιμάζουμε οποιαδήποτε από τις εντολές. Με την αποστολή κάποιας εντολής, θα πρέπει τα τέσσερα από τα πέντε LED να σβήνουν και το ένα που απομένει να υποδεικνύει το κανάλι που χρησιμοποιείται, ενώ όταν ολοκληρωθεί η εκτέλεση της εντολής τα τέσσερα LED θα πρέπει να επανέρχονται. Εάν όλα λειτουργούν όπως παραπάνω, μπορούμε αφού διακόψουμε την τροφοδοσία του κυκλώματος του ελεγκτή να συνδέσουμε σε κάποιο κανάλι ένα βηματικό κινητήρα. Εδώ χρειάζεται να προσέξουμε ώστε όλες οι φάσεις αλλά και οι κοινές απολήξεις να συνδέονται σωστά. Στην συνέχεια τροφοδοτούμε ξανά και ξανα-δοκιμάζουμε. Ο κινητήρας θα πρέπει να περιστρέφεται ομαλά, επιταχύνοντας κατά την εκκίνηση και επιβραδύνοντας κατά την παύση. Στην περίπτωση όπου η κίνηση δείχνει να γίνεται απότομα ή με μικρά 'πηδήματα', ελέγχουμε εκ νέου τόσο την ορθή σύνδεση των καλωδίων

του κινητήρα όσο και την ορθότητα των παραμέτρων που αφορούν τον ρυθμό επιτάχυνσης (να μην είναι υπερβολικά μεγάλος για τον συγκεκριμένο κινητήρα). Στο σημείο αυτό θα πρέπει να υπενθυμίσουμε ότι μικρότερες τιμές καθυστέρησης σημαίνουν γρηγορότερη περιστροφή. Εάν ορίσουμε την ταχύτητα περιστροφής με ισοδύναμες καθυστερήσεις (για παράδειγμα 800-800), δεν θα υπάρχει καμία επιτάχυνση ή επιβράδυνση κατά την περιστροφή. Οι περισσότεροι πάντως βηματικοί κινητήρες θα πρέπει να λειτουργούν χωρίς πρόβλημα με τιμές καθυστέρησης βημάτων 500-1000.

Όταν ολοκληρωθεί ο έλεγχος σε όλα τα κανάλια και αυτός αποδειχθεί σωστό, μπορούμε να προχωρήσουμε σε τροποποιήσεις είτε στο λογισμικό εντολών (στον υπολογιστή) είτε στο λογισμικό ελέγχου (PIC), με στόχο την βελτίωση αλλά και προσαρμογή της κατασκευής στις ανάγκες μας. Μία για παράδειγμα καλή μετατροπή θα ήταν να πολυπλέξουμε τους κινητήρες, δίνοντας την δυνατότητα ταυτόχρονου ελέγχου περισσότερων του ενός κινητήρων. Εφαρμογές πάντως της συγκεκριμένης κατασκευής θα μπορούσαμε να βρούμε στον χώρο της ρομποτικής, στην ακριβή περιστροφή μηχανικών μερών σε τηλεσκόπια, σε ρομπότ, κάμερες κ.λπ. ή ακόμη και στον έλεγχο εργαλείων κοπής και χάραξης. (020127-2)

Χρήσιμες διευθύνσεις

Φυλλάδιο δεδομένων σχετικά με τον μικροελεγκτή PIC 16F87X της Microchip: www.microchip.com/download/lit/pline/picmicro/familiew/16f87x/30292c.pdf

Απ' ευθείας διεύθυνση για κατέβαση του VCL QCCOM32 για I/O στην RS232 σε Borland Delphi: <http://geocities.com/scottpinkham/delphi/qccom32.zip>

Χαμηλού κόστους εργαλεία ανάπτυξης PIC συμβατά με το περιβάλλον MLAB της Microchip www.taylec.co.uk

Ένας μεταφραστής PIC συμβατός με το περιβάλλον MLAB της Microchip www.ccinfo.com

VCL για πρόσβαση και έλεγχο θυρών σε επίπεδο υλικού: www.programmesheaven.com

www.torry.net, www.codeguru.com

Χρήσιμη βιβλιογραφία

Εγχειρίδιο της σειριακής θύρας (Serial Port Complete) του Jan Axelson, ISBN: 0965081923
Προγραμματισμός PIC
Ελεκτορ, Οκτώβριος 2003

Δωρεάν από το διαδίκτυο

Λογισμικό για τον υπολογιστή και το PIC (αρχεία πηγαίου κώδικα)
Αριθμός αρχείου 020127-11.zip
Το σχέδιο της πλακέτας σε μορφή PDF, αριθμός αρχείου 020127-1.zip