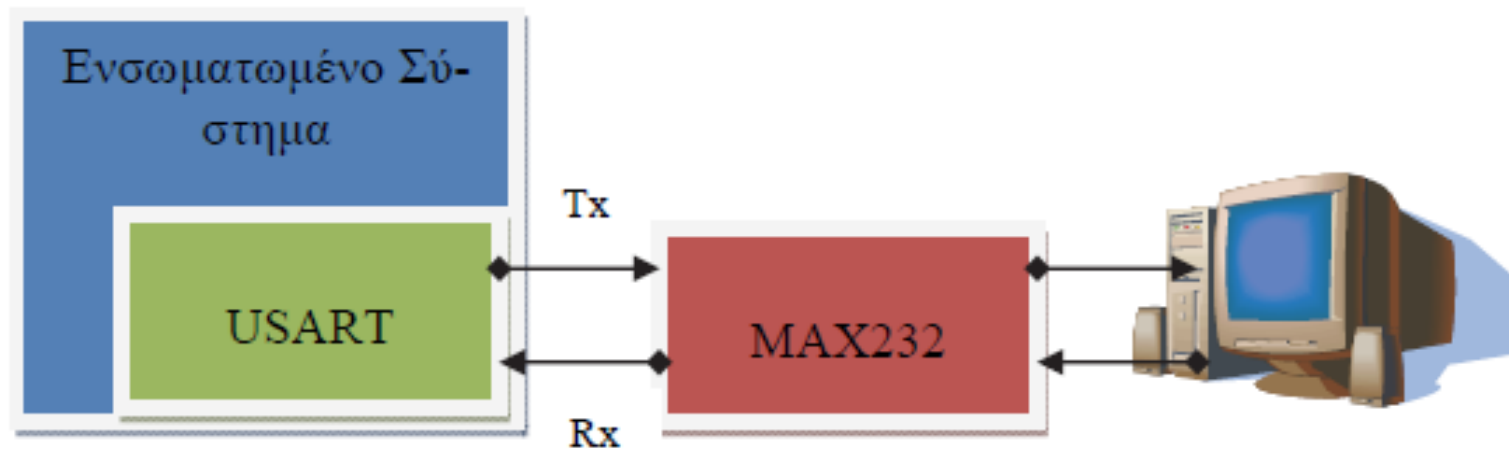
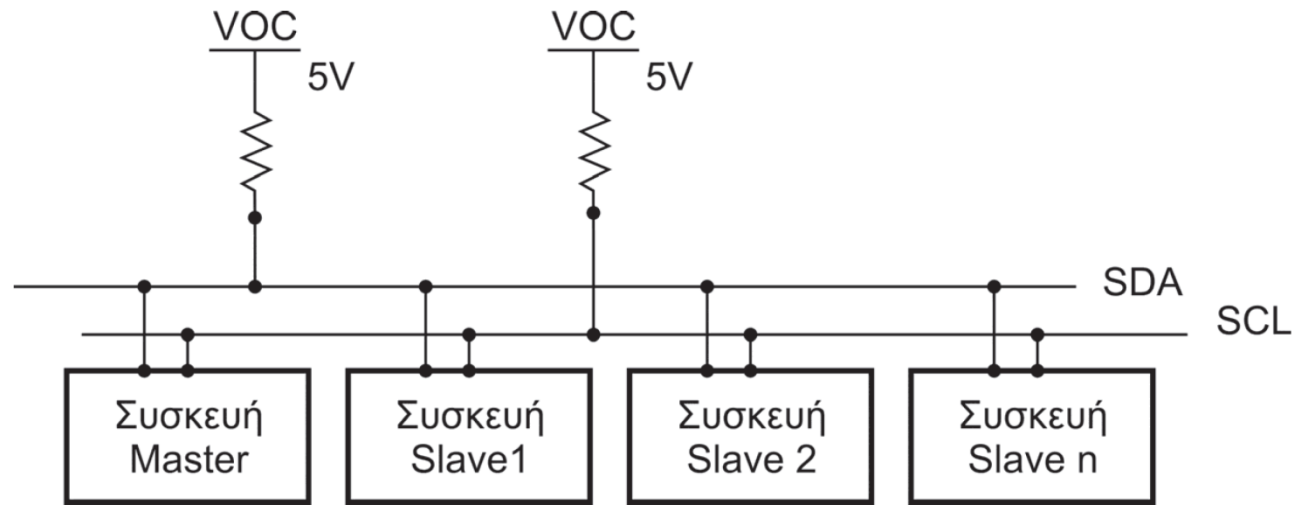


USART



Σχήμα 7.46: Διασύνδεση ενσωματωμένου συστήματος με PC.

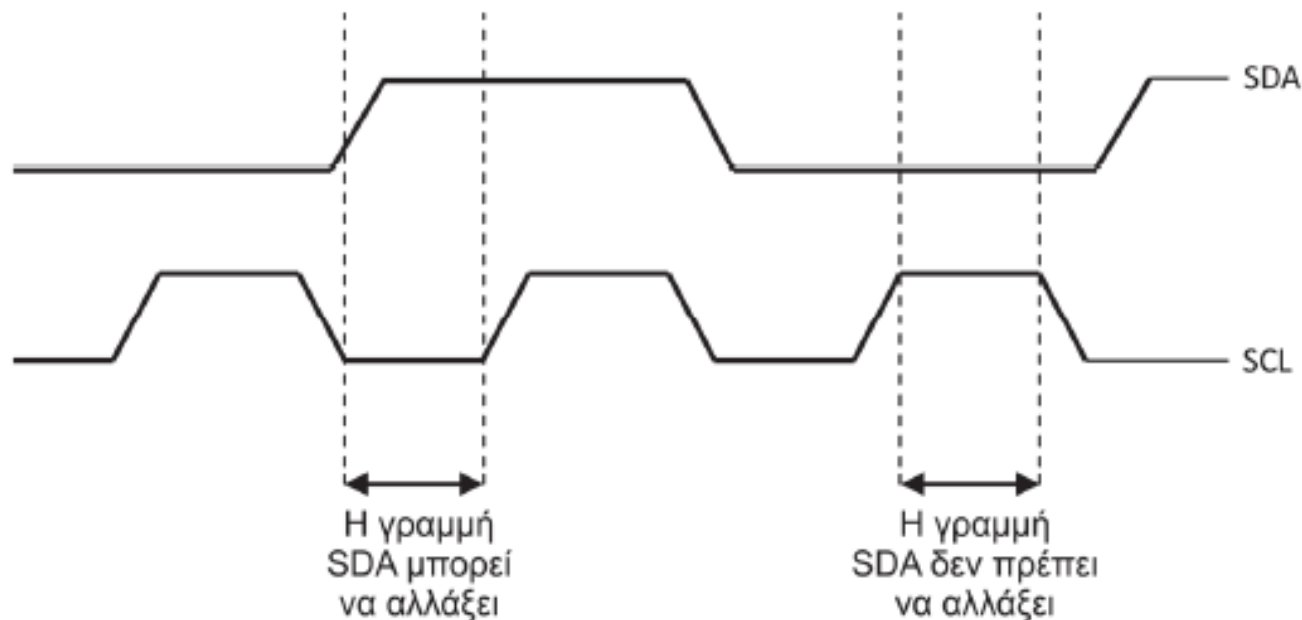
I²C



Σχήμα 7.47: Διασύνδεση συσκευών πάνω στον δίαυλο I²C.

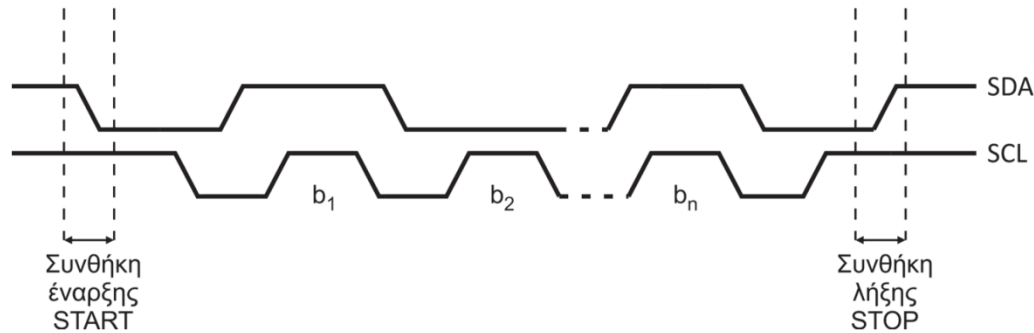
I²C

Τα δεδομένα στη γραμμή SDA πρέπει να μένουν σταθερά για το χρονικό διάστημα που η γραμμή SCL βρίσκεται σε υψηλή κατάσταση. Τα δεδομένα στη γραμμή SDA μπορούν να αλλάξουν μόνο όταν η γραμμή SCL βρίσκεται σε χαμηλή κατάσταση όπως απεικονίζεται στο Σχήμα 7.48.



Σχήμα 7.48: Διάγραμμα επιτρεπτών διαστημάτων αλλαγής της γραμμής SDA

I²C

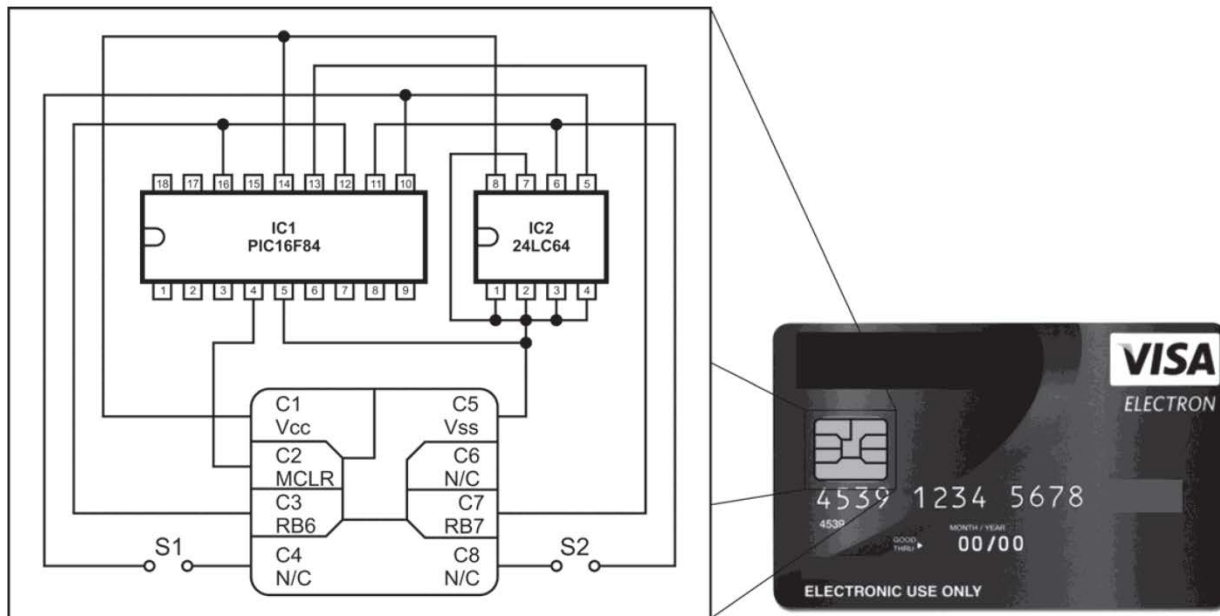


Σχήμα 7.49: Συνθήκες Start και Stop για το πρωτόκολλο επικοινωνίας I²C.

Το πρωτόκολλο επικοινωνίας I²C υποστηρίζει τέσσερις περιπτώσεις λειτουργίας για μία δεδομένη συσκευή πάνω στο δίαυλο:

1. Μετάδοση δεδομένων από τον Master όπου η συσκευή Master στέλνει δεδομένα σε μία συσκευή Slave.
2. Λήψη δεδομένων από τον Master όπου η συσκευή Master κάνει λήψη δεδομένων από μία συσκευή Slave.
3. Μετάδοση δεδομένων από τον Slave όπου μία συσκευή Slave στέλνει δεδομένα στον Master.
4. Λήψη δεδομένων από τον Slave όπου μία συσκευή Slave λαμβάνει δεδομένα από τον Master.

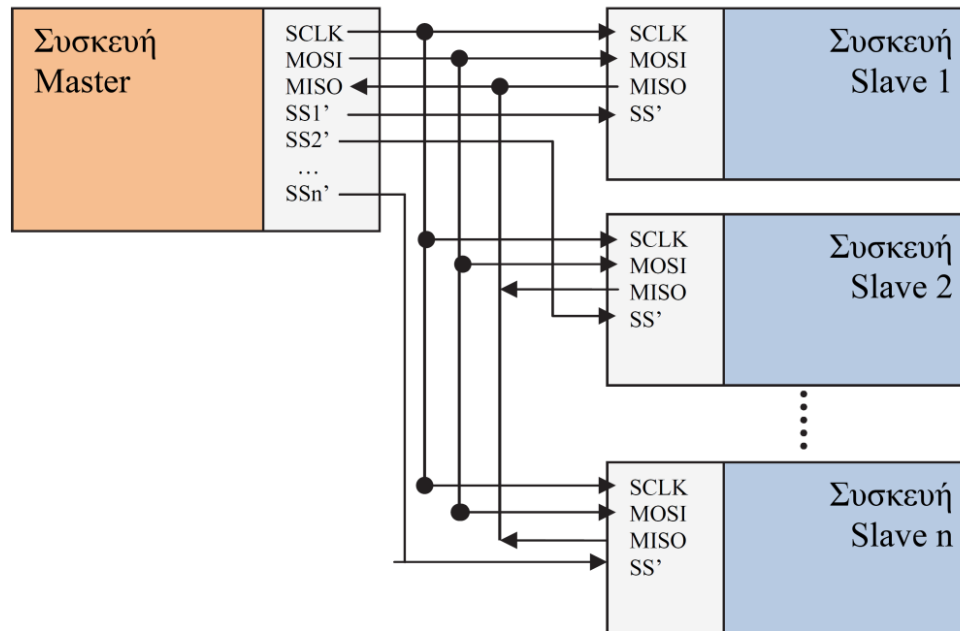
I²C



Σχήμα 7.50: Χρήση του I²C σε ενσωματωμένα κυκλώματα καρτών VISA

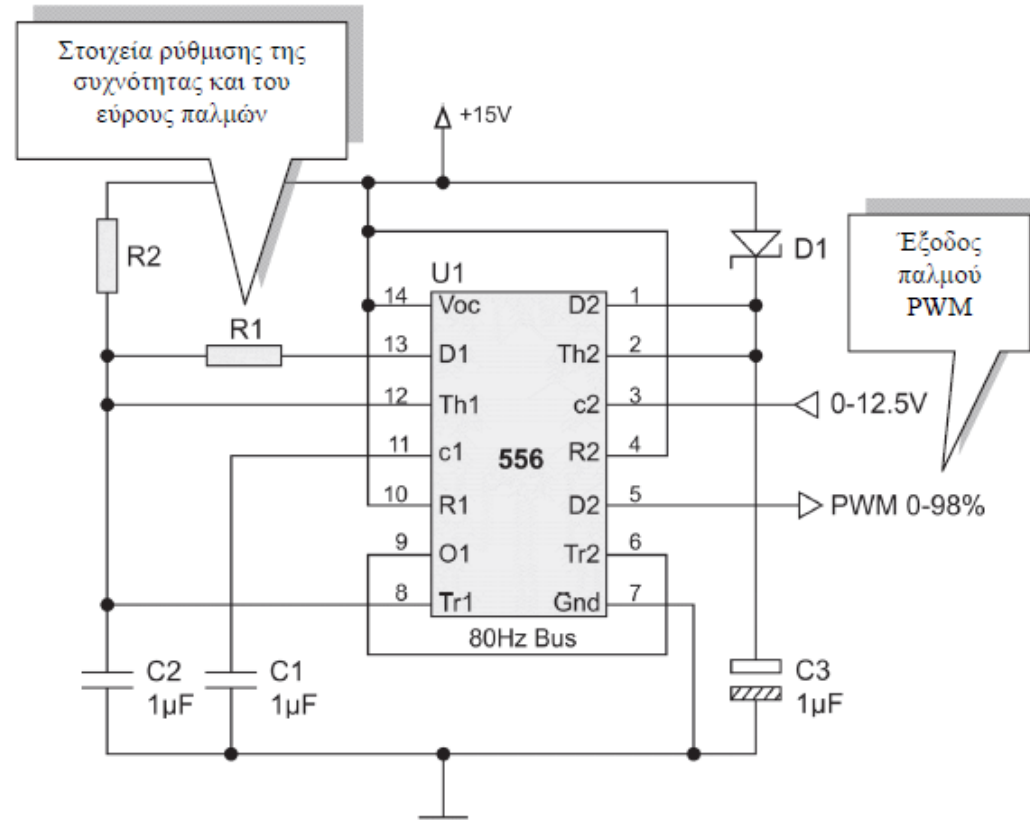
SPI

Ο διάυλος SPI (Serial Peripheral Interface) είναι ένα σειριακό πρωτόκολλο επικοινωνίας σύγχρονης μετάδοσης δεδομένων. Σε αυτό το διάυλο επικοινωνίας μπορούν να συνδεθούν δύο είδη συσκευών, μία συσκευή Master και μία ή περισσότερες συσκευές Slave. Ο διάυλος SPI υποστηρίζει ταυτόχρονη μετάδοση και λήψη δεδομένων (Full Duplex) με χρήση τεσσάρων γραμμών επικοινωνίας πάνω στις οποίες συνδέονται οι συσκευές όπως απεικονίζεται στο Σχήμα 7.51.



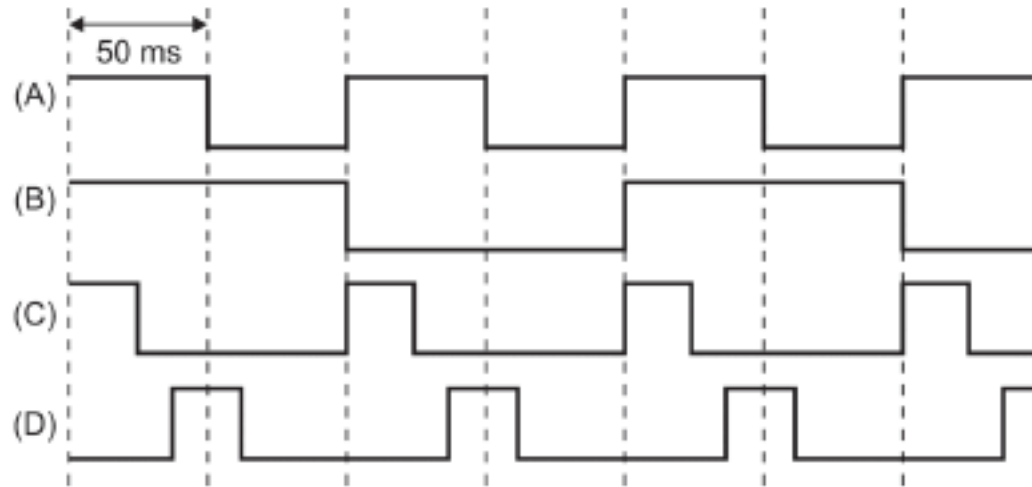
- MOSI – Γραμμή μετάδοσης δεδομένων από τον Master στον Slave (Master Out Slave In)
- MISO – Γραμμή μετάδοσης δεδομένων από τον Slave στον Master (Master In Slave Out)
- SCLK – Γραμμή χρονισμού των γραμμών MOSI και MISO (Serial CLock)

PWM

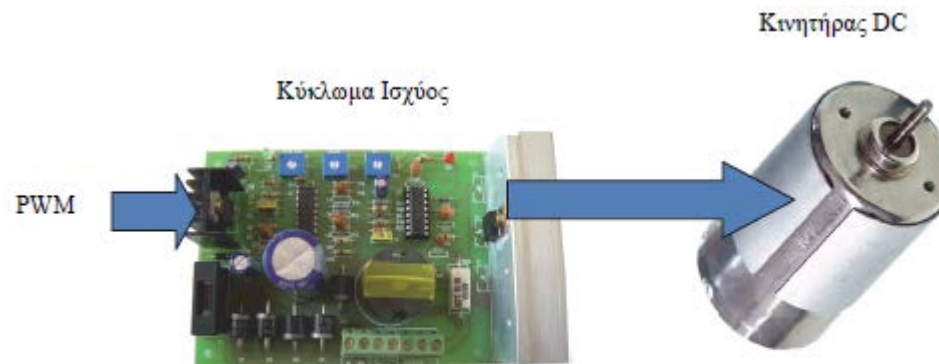


Σχήμα 7.52: Κυκλωματική διάταξη για την παραγωγή σήματος PWM με χρήση του ολοκληρωμένου κυκλώματος 555.

PWM



Σχήμα 7.53: Χρονικό διάγραμμα τεσσάρων σημάτων PWM με διαφορετικό κύκλο εργασίας (A) 20Hz, 50%, (B) 10 Hz, 50%, (C) 10Hz, 25%, (D) 10Hz, 12,5%.

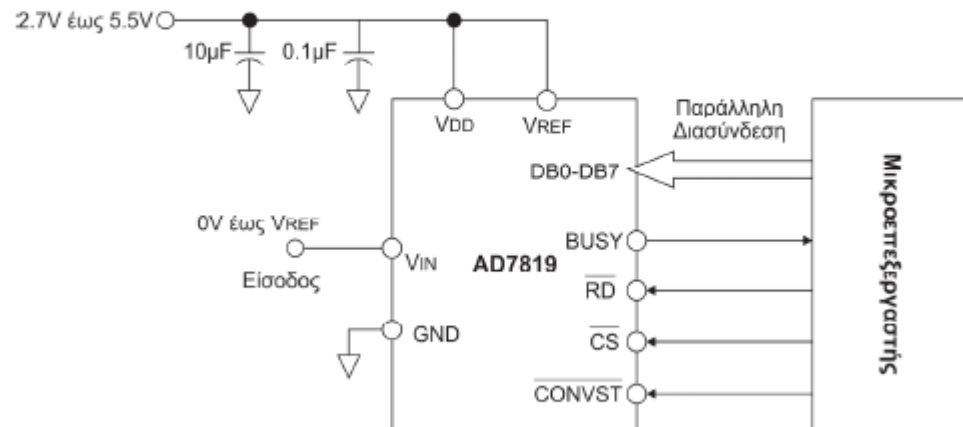


Σχήμα 7.54: Οδήγηση κινητήρα DC με κύκλωμα ισχύος και PWM.

ADC

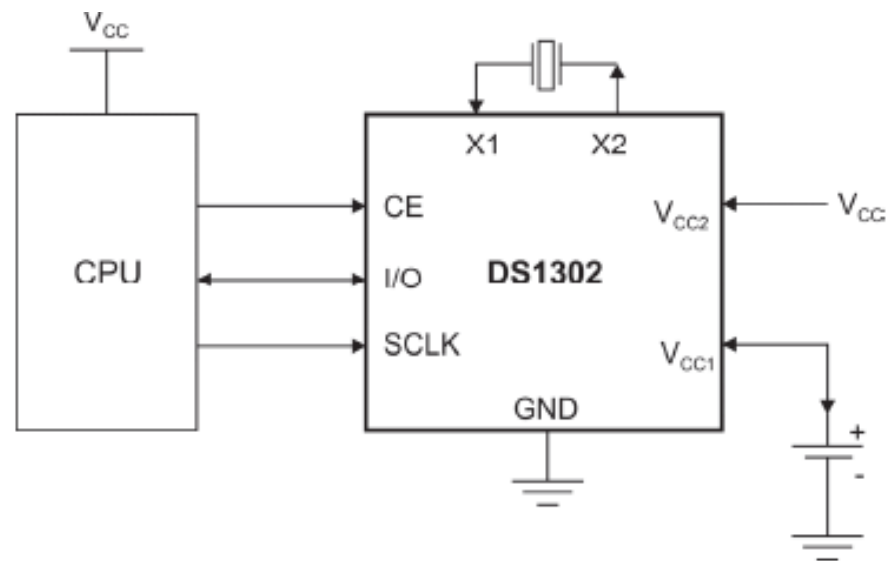
Μία εξωτερική περιφερειακή μονάδα μετατροπέα A/D που χρησιμοποιεί παράλληλη επικοινωνία (όπως το AD7819, Σχ. 7.20), αποτελείται συνήθως τα ακόλουθα σήματα ελέγχου:

- CS (Chip select): επιλογή του ολοκληρωμένου κυκλώματος του μετατροπέα από τον επεξεργαστή του συστήματος.
- RD (Read Pin): το σήμα αυτό δηλώνει την ανάγνωση της ψηφιακής λέξης, που βρίσκεται στους διαδρόμους δεδομένων του μετατροπέα, από τον επεξεργαστή του συστήματος.
- BUSY: το σήμα αυτό ενημερώνει τον επεξεργαστή ότι ο μετατροπέας εκτελεί κάποια λειτουργία μετατροπής ενός σήματος.
- CONVST (Convert Start): αυτή η γραμμή ελέγχου χρησιμοποιείται από τον επεξεργαστή για να οδηγήσει τον μετατροπέα έτσι ώστε αυτός να ξεκινήσει τη διαδικασία μετατροπής.



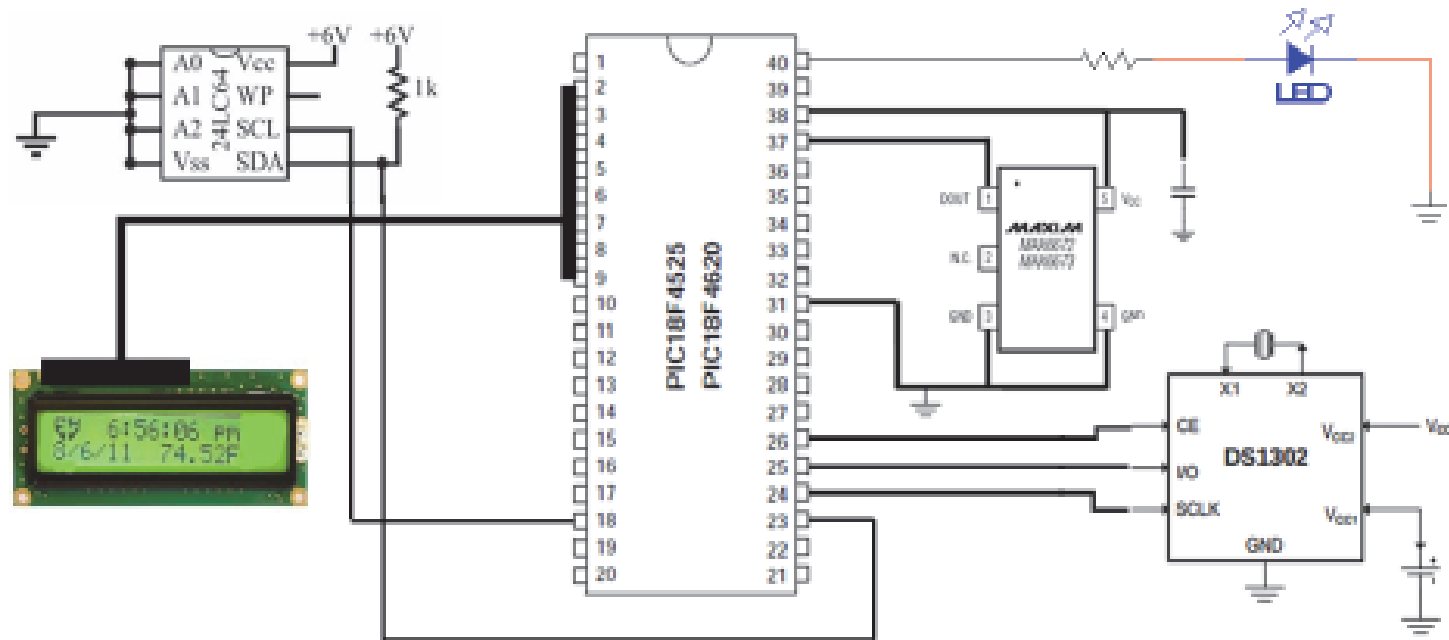
Σχήμα 7.20: Συνδεσμολογία μετατροπέα AD7819 σε μικροπολογιστικό σύστημα.

RTC



Σχήμα 7.61: Το ολοκληρωμένο κύκλωμα RTCC DS1302 και ο τρόπος συνδεσμολογίας του με έναν επεξεργαστή.

PWM



Σχήμα 7.64: Διασύνδεση περιφερειακών μονάδων με μικροελεγκτή για την υλοποίηση ενσωματωμένου συστήματος απόκτησης δεδομένων θερμοκρασίας.

WDT

1. Υπολογίζουμε το χρόνο υπερχειλίσσης του WDT σύμφωνα με τη συχνότητα του ρολογιού του συστήματος και επιλέγουμε μία κατάλληλη κλίμακα υποδιαίρεσης.
2. Μηδενίζουμε τον μετρητή του WDT και ρυθμίζουμε τους καταχωρητές που σχετίζονται με τη λειτουργία του.
3. Ενεργοποιούμε τον χρονιστή WDT.
4. Φροντίζουμε να μηδενίζουμε τον χρονιστή ανά τακτά χρονικά διαστήματα μέσα στον κώδικα του προγράμματός μας ώστε να μην φτάσει ποτέ σε υπερχειλίση που να οφείλεται σε καθυστέρηση λόγω της εκτέλεσης των εντολών του προγράμματος.

Η χρήση του WDT μπορεί να γίνει προβληματική αν δεν μελετηθεί σωστά ο χρόνος που πρέπει να περάσει για να μην φτάσει σε υπερχειλίση λόγω καθυστέρησης στην εκτέλεση των εντολών του προγράμματος. Στη συνέχεια απεικονίζεται ένα παράδειγμα σωστής και λάθος χρήσης του WDT μέσα σε ένα πρόγραμμα. Αν ο συνολικός χρόνος $t1+t2$ που απαιτείται για την εκτέλεση των δύο τμημάτων εντολών του παρακάτω παραδείγματος είναι μεγαλύτερος από τον χρόνο που απαιτείται για να υπερχειλίσει ο χρονιστής τότε πρέπει να μηδενίσουμε τον χρονιστή σε κάποιο σημείο του κώδικα όπου ο χρόνος που έχει περάσει μέχρι αυτό δεν υπερβαίνει τον χρόνο υπερχειλίσσης. Έτσι, αν οι χρόνοι $t1$ και $t2$ είναι μικρότεροι από το χρόνο υπερχειλίσσης του χρονιστή τότε μπορούμε να μηδενίσουμε τον χρονιστή σε κάποιο σημείο μεταξύ των τμημάτων εντολών 1 και 2. Στην περίπτωση που δεν μηδενίσουμε τον χρονιστή έγκαιρα το σύστημα θα κάνει επανεκκίνηση.