



ΑΝΩΤΑΤΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ
ΙΔΡΥΜΑ ΛΑΡΙΣΑΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

« Η Διδασκαλία εκπαιδευτικής ρομποτικής με τη χρήση
μικροελεγκτών(π.χ. ARDUINO, PIC) »

Βασιλική Παυλή

T-2716

ΕΠΙΒΛΕΠΩΝ: Σούλτης Γεώργιος , Ηλεκτρολόγος Μηχανικός, Επίκουρος Καθηγητής

Λάρισα 2013

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

Λάρισα, / / 201

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1.

2.

3.

«Δηλώνω υπεύθυνα ότι το παρόν κείμενο αποτελεί προϊόν προσωπικής μελέτης και εργασίας και πως όλες οι πηγές που χρησιμοποιήθηκαν για τη συγγραφή της δηλώνονται σαφώς είτε στις παραπομπές είτε στη βιβλιογραφία. Γνωρίζω πως η λογοκλοπή αποτελεί σοβαρότατο παράπτωμα και είμαι ενήμερος/η για την επέλευση των νομίμων συνεπειών»

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον εισηγητή της πτυχιακής μου εργασίας κ.

Σούλη Γεώργιο ο οποίος εμπιστεύτηκε πως θα φέρω εις πέρας αυτό το

δύσκολο έργο και ανέλαβε την επίβλεψη της εργασίας μου.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξη καθ' όλη τη

διάρκεια των σπουδών μου, τους φίλους και τα αγαπημένα μου πρόσωπα που με

στήριξαν κατά τη διάρκεια εκπόνησης της εργασίας αυτής.

ΠΕΡΙΛΗΨΗ

Η πτυχιακή αυτή εργασία έχει σαν στόχο να διερευνήσει την δυνατότητα διδασκαλίας της εκπαιδευτικής ρομποτικής κυρίως στην δευτεροβάθμια εκπαίδευση με τη χρήση των ARDUINO και PIC. Θα γίνει προσπάθεια να εκτιμηθούν οι διαφορές, τα πλεονεκτήματα και τα μειονεκτήματα από μια τέτοια διδασκαλία.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ.....	iv
ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ.....	vii
1 ΚΕΦΑΛΑΙΟ.....	11
1.1 Η εξέλιξη της ρομποτικής και της εκπαιδευτικής ρομποτικής.....	11
1.1.1 Η γέννηση και η εξέλιξη της ρομποτικής.....	11
1.1.2 Βασικά στοιχεία και αρχές των ρομπότ, ταξινόμηση σε είδη.....	23
1.1.3 Βασικά κατασκευαστικά και επιστημονικά στοιχεία.....	28
1.1.4 Η εκπαιδευτική ρομποτική – η εξέλιξη της LOGO.....	33
1.1.5 Συστήματα LEGO και γλώσσες εκπαιδευτικής ρομποτικής	40
1.1.6 Ο διαγωνισμός της εκπαιδευτικής ρομποτικής.....	48
2 ΚΕΦΑΛΑΙΟ.....	58
Οι μικροελεγκτές ARDUINO και PIC	58
2.1 Μικροελεγκτές.....	58
2.1.1 Ιστορία των μικροεπεξεργαστών-μικροελεγκτών.....	58
2.1.2 Η εξέλιξη των μικροεπεξεργαστών.....	59
2.1.3 Βασικά χαρακτηριστικά που καθιστούν προτιμότερο τον μικροελεγκτή.....	62
2.1.4 Διαφορά μεταξύ μικροελεγκτή και μικροεπεξεργαστή.....	64
2.1.5 Διαδεδομένες κατηγορίες μικροελεγκτών.....	65
2.1.6 Εργαλεία ανάπτυξης και γλώσσες προγραμματισμού.....	66
2.1.7 Κατασκευαστές μικροελεγκτών	66
2.2 Ο μικροελεγκτής ARDUINO.....	67
2.2.1 Ιστορία του ARDUINO.....	67
2.2.2 Τι είναι ο ARDUINO.....	69

2.2.3 Η γλώσσα προγραμματισμού που χρησιμοποιείται.....	69
2.2.4 Πλεονεκτήματα του ARDUINO.....	69
2.2.5 Δυνατότητες του ARDUINO.....	70
2.2.6 Εκδόσεις ARDUINO.....	70
2.2.7 Εφαρμογές του ARDUINO.....	70
2.3 Αναλυτική παρουσίαση του μικροελεγκτή ARDUINO.....	79
2.3.1 Hardware.....	79
2.3.2 Μοντέλα ARDUINO.....	80
2.3.3 Arduino Shields.....	81
2.3.4 Χαρακτηριστικά.....	83
2.3.5 Software.....	88
2.3.6 Το περιβάλλον ανάπτυξης	90
2.3.7 ARDUINO Example.....	95
2.4 Ο μικροελεγκτής PIC.....	107
2.4.1 Ιστορία του PIC.....	107
2.4.2 Τι είναι ο PIC.....	110
2.4.3 Η γλώσσα προγραμματισμού που χρησιμοποιείται.....	111
2.4.4 Πλεονεκτήματα του PIC.....	112
2.4.5 Εκδόσεις PIC.....	112
2.4.6 Εφαρμογές του PIC.....	113
2.5 Αναλυτική παρουσίαση του μικροελεγκτή PIC.....	121
2.5.1 Διάφοροι τύποι του PIC.....	121
2.5.2 Τύποι της μνήμης προγράμματος	123
2.5.3 Αρχιτεκτονική του επεξεργαστή.....	123
2.5.4 Ρεπερτόριο εντολών του PIC.....	129
2.5.5 Software.....	130

2.5.6 PIC Example.....	138
3 ΚΕΦΑΛΑΙΟ.....	149
3.1 Σχεδιασμός διδασκαλίας εκπαιδευτικής ρομποτικής με τον ARDUINO και PIC.....	149
3.1.1 Η διδασκαλία της εκπαιδευτικής ρομποτικής – στόχοι – αντικείμενο..	149
3.1.2 Λεπτομερής σχεδιασμός μιας διδασκαλίας με τους μικροελεγκτές....	157
3.1.3 Δυσκολίες και πλεονεκτήματα.....	170
3.1.4 Προτάσεις.....	171
ΑΝΑΦΟΡΕΣ	
ΒΙΒΛΙΟΓΡΑΦΙΑ	

ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ

Εικόνα 1.1: μηχανήμα ρομποτικής.....	12
Εικόνα 1.2: απεικόνιση ενός ρομπότ.....	13
Εικόνα 1.3: βιομηχανικός ρομποτικός βραχίονας.....	26
Εικόνα 1.4: βαδίζοντα ρομπότ, αυτόνομα έντροχα ρομπότ, AGVs, ROVs, εναέρια ρομπότ, AUVs.....	27
Εικόνα 1.5: ανδροειδές ρομπότ.....	29
Εικόνα 1.6: ρομπότ στην ιστορία.....	30
Εικόνα 1 7: ρομπότ στο διάστημα.....	30
Εικόνα 1.8: robot dog.....	31
Εικόνα 1.9: ρομποτικοί βραχίονες.....	32
Εικόνα 1.10: ενδεικτικό πρόγραμμα με επανάληψη (loop) για την κίνηση του ρομπότ σε τετράγωνο	42
Εικόνα 1.11: world robot Olympiad.....	48
Εικόνα 1.12: μέλη της πρώτης επιτροπής WOR.....	50
Εικόνα 1.13: WOR GEN II Football.....	54
Εικόνα 2.1: intel4004, intel8080, intel8086/8088, intel80286, intel80386, intel80486, intel Pentium, Pentium II, Pentium III, Pentium IV.....	61
Εικόνα 2.2: Arduino LilyPad.....	68
Εικόνα 2.3: Arduino Duemilano, Mega, Freeduino v1.16, Seeduino.....	71
Εικόνα 2.4: Laser harp.....	72
Εικόνα 2.5: Soundmachine.....	73
Εικόνα 2.6: τηλεκατευθυνόμενο όχημα.....	73
Εικόνα 2.7: Led Cube.....	74
Εικόνα 2.8: RGB combination door lock.....	75
Εικόνα 2.9: Senseless drawing bot.....	77

Εικόνα 2.10: LumiBots.....	78
Εικόνα 2.11: Mind-control.....	78
Εικόνα 2.12: περιβάλλον ανάπτυξης του Arduino.....	90
Εικόνα 2.13: λειτουργίες του IDE.....	91
Εικόνα 2.14: Turn all LED on and off.....	95
Εικόνα 2.15: Virtual color mixer.....	97
Εικόνα 2.16: LCD display.....	100
Εικόνα 2.17: κύκλωμα χτύπου.....	105
Εικόνα 2.18: μικροελεγκτής PIC.....	111
Εικόνα 2.19: PIC sonar (ultrasonic) range finding.....	114
Εικόνα 2.20: PIC based BRAM (beginner's robot autonomous mobile).....	115
Εικόνα 2.21: Versatile central heating program controller.....	116
Εικόνα 2.22: A versatile temperature data logger using PIC12F683 & DS1820...117	117
Εικόνα 2.23: Gas sensor	118
Εικόνα 2.24: RS232 communication with PIC microcontroller.....	119
Εικόνα 2.25: 3 LED bike light.....	119
Εικόνα 2.26: ενεργοποίηση του simulator.....	135
Εικόνα 2.27: Special function registers.....	137
Εικόνα 2.28: εικόνα προγράμματος	138
Εικόνα 2.29: Basic LCD test.....	139
Εικόνα 2.30: playing with LEDs.....	140
Εικόνα 2.31: RS-232 communication with PC.....	141
Εικόνα 2.32: PC's serial port terminal.....	142
Εικόνα 2.33: I2C communication with PCF8583 real time clock.....	144
Εικόνα 3.1: πλακέτα Arduino.....	159
Εικόνα 3.2: λειτουργία της πλακέτας.....	160

Εικόνα 3.3: περιβάλλον ανάπτυξης προγραμματισμού.....	161
Εικόνα 3.4: παράδειγμα Blink.....	162-3
Εικόνα 3.5: διαδικασία ολοκλήρωσης μεταγλώττισης.....	163
Εικόνα 3.6: συριακή επικοινωνία του Arduino με τον υπολογιστή.....	164
Εικόνα 3.7: κώδικας του hello word.....	164-5
Εικόνα 3.8: μήνυμα του εκτελέσιμου προγράμματος	165
Εικόνα 3.9: μικροελεγκτής PIC.....	166
Εικόνα 3.10: δομή του μικροελεγκτή PIC.....	166
Εικόνα 3.11: εγκατάσταση του προγράμματος στον υπολογιστή.....	167
Εικόνα 3.12: πρόγραμμα hello word.....	168
Εικόνα 3.13: πρόγραμμα που αναβοσβήνει τα LED.....	169
 Πίνακας 1: μοντέλα Arduino.....	 80-1
Πίνακας 2: Arduino shields, λεπτομέρειες.....	81-3
Πίνακας 3: χαρακτηριστικά εν συντομία	83
Πίνακας 4: χαρακτηριστικά για την παραδοσιακή και σύγχρονη θεωρία μάθησης	150-51

1.1 Η εξέλιξη της ρομποτικής και της εκπαιδευτικής ρομποτικής

1.1.1 Η γέννηση και η εξέλιξη της ρομποτικής

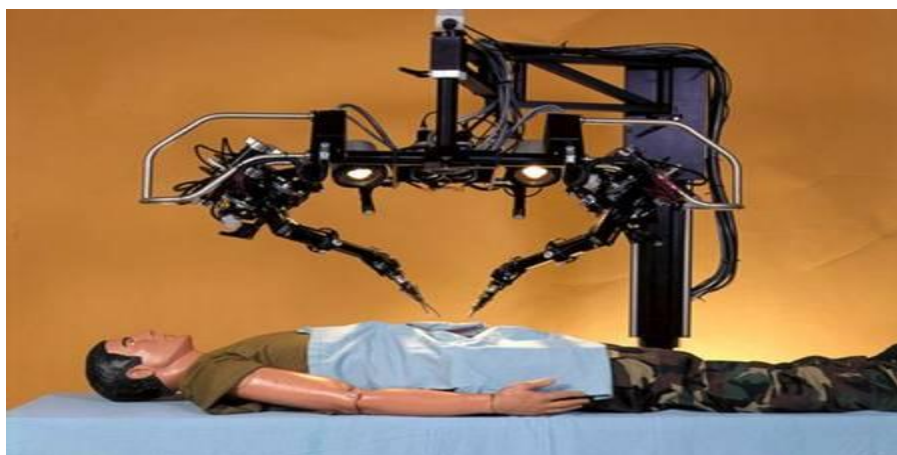
Νοημοσύνη

Η νοημοσύνη δεν είναι η ικανότητα μαθηματικών υπολογισμών αλλά η ικανότητα να κρίνεις, να αποφασίζεις πως θα πραγματοποιήσεις μια πράξη εξετάζοντας όλους τους επιμέρους παράγοντες και επιλέγοντας την καλύτερη σειρά ενεργειών, ακόμα και αν είναι μια νέα επιλογή.

Ορισμός ρομποτικής

Η ρομποτική είναι ο κλάδος της επιστήμης όπου οι μηχανικοί ασχολούνται με τη σύλληψη, το σχεδιασμό, την κατασκευή και τη λειτουργία των ρομπότ, παράγωγος κλάδος της τεχνολογίας του αυτοματισμού. Τα ρομπότ είναι μηχανές, η χρήση των οποίων αποσκοπεί στην αντικατάσταση του ανθρώπου στην εκτέλεση έργου. Η αντικατάσταση αυτή αφορά τόσο στο φυσικό επίπεδο του έργου όσο και στο επίπεδο λήψης της απόφασης. Θεμέλια της ρομποτικής τεχνολογίας αποτελούν οι τεχνολογίες του **ψηφιακού ελέγχου** (NC)- μεθόδου προγραμματισμού εργαλειομηχανών για την εκτέλεση περίπλοκων κατεργασιών- και **της τηλεχειρικής** (tele-cherics)- μεθόδου χρήσεως μηχανικών βραχιόνων για την εξ-αποστάσεως εκτέλεση λεπτών χειρονακτικών χειρισμών σε επικίνδυνα αντικείμενα ή σε επικίνδυνους για τον άνθρωπο χώρους. Είναι αρκετά δύσκολο να οριστεί η διαχωριστική γραμμή μεταξύ ρομποτικών μηχανισμών και απλώς αυτοματοποιημένων μηχανών. Αρχικά το ρομπότ συγκροτείται από δύο συστήματα, το μηχανικό (στο οποίο περιλαμβάνεται το σύστημα κίνησης) και το ηλεκτρονικό (στο οποίο υπάγεται και η επαναπρογραμματιζόμενη μνήμη του).

Αναζητώντας κανείς τις ρίζες της ρομποτικής θα οδηγηθεί αρκετά πίσω στην ιστορία της ανθρωπότητας.



Εικόνα 1.1: μηχανήμα ρομποτικής

Ορισμός ρομπότ

Κατά γενικό κανόνα, όσο πιο περίπλοκη και εξειδικευμένη είναι η μηχανή τόσο μεγαλύτερη και η πιθανότητα χαρακτηρισμού της ως ρομπότ. Η λέξη «ρομπότ», γέννημα τού 20ού αιώνα, χρησιμοποιήθηκε για πρώτη φορά στο θεατρικό έργο R.U.R. (Rossum's Universal Robots) τού Τσέχου συγγραφέα Κάρελ Τσάπεκ ο οποίος την παρήγαγε από την τσεχική λέξη ρομπότα, που σημαίνει «αγγαρεία». Η παλαιότερη ελληνική λέξη αυτόματο χρησιμοποιείται πλέον περισσότερο για μηχανισμούς που μιμούνται τον άνθρωπο ή κάποιο ζώο, χωρίς αναγκαστικά να παράγουν ωφέλιμο έργο. Ο νεολογισμός «ανδροειδές» περιορίζεται σε ανθρωπόμορφους όχι όμως και ζωόμορφους, μηχανισμούς. Σύμφωνα με τον ορισμό του Ινστιτούτου Ρομπότ των ΗΠΑ, "ρομπότ είναι μια επαναπρογραμματιζόμενη πολύ-λειτουργική χειριστική διάταξη, σχεδιασμένη για τη μετακίνηση υλικών, εξαρτημάτων, εργαλείων και εξειδικευμένων διατάξεων, μέσω μεταβλητών, προγραμματισμένων κινήσεων για την εκτέλεση μιας σειράς εργασιών".



Εικόνα 1.2: απεικόνιση ενός ρομπότ

Ιστορία

Η ιστορία του ρομπότ έχει τις ρίζες της ήδη από αρχαίους μύθους και θρύλους. Σύγχρονες έννοιες που άρχισαν να αναπτύσσονται όταν η βιομηχανική επανάσταση επέτρεψε τη χρήση πιο πολύπλοκων μηχανών και την επακόλουθη εισαγωγή του ηλεκτρισμού κατέστησε δυνατές τις μηχανές ισχύος με μικρούς συμπαγείς κινητήρες.

Στην αρχαία Ελλάδα υπήρχαν αναφορές για διάφορους αυτοματισμούς που η προέλευσή τους είναι μυστήριο ακόμη και σήμερα. Οι αναφορές μιλούν για:

- **Χρυσούς Βοηθούς**
Ο Ήφαιστος είχε φτιάξει μερικές χρυσές γυναίκες να τον βοηθάνε στο εργαστήριο, να τον στηρίζουν για να περπατάει καλύτερα αλλά και για να έχει κάποιον να μιλάει.
- **Μηχανικοί Σκύλοι**
Οι θεοί ήταν ιδιαίτερα ευχαριστημένοι με τον βασιλιά Αλκίνοο και μέσω της τέχνης του Ηφαίστου, του χάρισαν χρυσούς και ασημένιους, αθάνατους και πανίσχυρους μηχανικούς σκύλους για την προστασία του παλατιού του.
- **Κινούμενες Κούκλες**
Ο Όμηρος και ο Πλάτυνας αναφέρουν ότι ο Δαίδαλος ανάμεσα στις άλλες εντυπωσιακές του κατασκευές έφτιαξε και κούκλες για τα παιδιά του Μίνωα. Σαν βασιλικά παιχνίδια δεν ήταν συνηθισμένες αλλά μπορούσαν να μιλάνε και να κινούνται. Λέγετε μάλιστα ότι αναγκαζόταν να τις δένουν για να μην τους φεύγουν μακριά και τις χάνουν. Το ίδιο λέγεται και για τους μηχανικούς ανθρωπόμορφους φύλακες του λαβύρινθου που κινούνταν με υδράργυρο.
- **Τάλως**
Ένα από τα πιο γνωστά ρομπότ στην Ελλάδα ήταν ο Τάλως. Κατασκευάστηκε από τον θεό Ήφαιστο σαν δώρο στον βασιλιά της Κρήτης Μίνωα. Ο Τάλως ήταν τεράστιος ανθρωπόμορφος και χάλκινος. Προστάτευε την Κρήτη από τους εχθρούς της και επέβλεπε την εφαρμογή των νόμων. Μπορούσε να κινείται πολύ γρήγορα και ήταν σε θέση να κάνει σε μια μέρα τον γύρο της Κρήτης (περίπου 250 km/h). Είχε τη δύναμη να εκσφενδονίζει τεράστιους βράχους εναντίων των αντιπάλων του ή να τους καίει με την αναπνοή του που πετούσε φωτιά. Με αυτόν τον τρόπο έδιωχνε τα εχθρικά πλοία, προστατεύοντας την Κρήτη.

➤ **Θρόνος Παγίδα**

Ένα αυτόματο μηχανήμα κατασκεύασε ο Ήφαιστος για να εκδικηθεί την μητέρα του Ήρα που τον απέρριψε σαν άσχημο μωρό. Ήταν ένας εντυπωσιακά καλοφτιαγμένος θρόνος που όταν η Ήρα κάθισε πάνω του αυτόματα σφίχτηκαν γύρω της αλυσίδες κρατώντας την δέσμια. Κανένας δεν μπορούσε να την απελευθερώσει από τα δεσμά της και ο Ήφαιστος ούτε που δεχόταν να συζητήσει την απελευθέρωσή της. Τελικά ο Διόνυσος τον επισκέφτηκε και αφού τον μέθυσε τον έπεισε να απελευθερώσει την μάνα του από τα δεσμά. Οι υπόλοιποι θεοί αναγνωρίζοντας τις δυνάμεις του και τα ταλέντα του τον δέχτηκαν στον Όλυμπο.

Αν κοιτάξουμε την ιστορία πέρα από την αρχαία Ελλάδα, θα δούμε ότι από πολύ νωρίς έχουν γίνει αναφορές ή ακόμη και προσπάθειες για την ανάπτυξη και την δημιουργία ρομπότ και πολύπλοκων κατασκευών. Μερικά επιτεύγματα παρουσιάζονται στην πορεία ξεκινώντας από την χρονολογία

100-150 π.χ.

Γίνεται λόγος για τον μηχανισμό των Αντικυθήρων όπου είναι ο αρχαιότερος αυτοματισμός που σώζεται ως σήμερα. Αυτός ο μηχανισμός μπορούσε να προβλέψει τις θέσεις των πλανητών.

250π.χ.

Ο Έλληνας φυσικός και εφευρέτης Κτησίβιος από την Αλεξάνδρεια σχεδίασε ρολόγια νερού που έχουν κινητά στοιχεία σε αυτά. Τα ρολόγια νερού είναι ένα μεγάλο επίτευγμα. Μέχρι τότε οι Έλληνες χρησιμοποιούσαν τις κλεψύδρες και η εφεύρεση του Κτησίβιου άλλαξε το γεγονός αυτό διότι ο χρόνος μετρήθηκε ως αποτέλεσμα της δύναμης του νερού που έπεφτε σε αυτό με έναν σταθερό ρυθμό.

322 π.χ.

Ο Αριστοτέλης ανέλαβε μια παλαιότερη αναφορά στην Ιλιάδα του Ομήρου και σκέφτηκαν ότι τα αυτόματα θα μπορούσαν κάποια μέρα να επιφέρουν την ισότητα μεταξύ των ανθρώπων καθιστώντας την κατάργηση της δουλείας δυνατή στο βιβλίο του, Πολιτική.

350 π.χ.

Ο λαμπρός Έλληνας μαθηματικός, Αρχύτας από τον Τάραντα δημιούργησε ένα μηχανικό πουλί, που ονομάστηκε «το περιστέρι» όπου τροφοδοτούταν από ατμό και μπορούσε να διανύσει μέχρι και 200μ. Εξυπηρέτησε στις πρώτες μελέτες της πτήσης στην ιστορία, για να μην αναφέρουμε ίσως το πρώτο μοντέλο αεροπλάνου.

1^{ος} αιώνας

Ο Ήρων της Αλεξάνδρειας (**10-70 μ.χ.**), δημιούργησε κάποιες μηχανικές συσκευές μεταξύ των οποίων μία που υποτίθεται ότι μπορούσε να μιλήσει.

1200 μ.χ.

Ο Άραβας Al-Jazari κατασκεύασε το πρώτο ανθρωποειδές ρομπότ το οποίο ήταν προγραμματισμένο να παίζει ένα τύμπανο.

1495

Γίνεται αναφορά στον Leonardo Di Vinci όπου πριν αρχίσει την εργασία για το Μυστικό Δείπνο σχεδίασε το πρώτο ανθρωποειδές ρομπότ. Η μηχανική κατασκευή έμοιαζε με έναν πάνοπλο ιππότη και ήταν έτσι σχεδιασμένος ώστε οι κινήσεις που κάνει να φαίνεται σαν να υπάρχει άνθρωπος μέσα σε αυτό.

1738

Ο Jacques de Vaucanson άρχισε να φτιάχνει αυτόματα στο Grenoble της Γαλλίας. Αυτός δημιούργησε τρία αυτόματα. Το πρώτο του ήταν ο παίχτης φλάουτου όπου μπορούσε να παίζει 12 τραγούδια. Το δεύτερο αυτόματο που δημιούργησε ήταν αρκετά κοντά στο πρώτο και μπορούσε να παίζει φλάουτο, τύμπανο ή ένα ντέφι, αλλά το τρίτο δημιούργημά του ήταν το πιο διάσημο απ' όλα. Η πάπια ήταν ένα παράδειγμα απόπειρας του Vaucanson όπου το ονόμασε «κινούμενη ανατομία» ή ανατομία ανθρώπου ή ζώου με μηχανές. Η πάπια μετακινούνταν, έσκουζε, χτυπούσε τα φτερά της και ακόμα έτρωγε και χώνευε το φαγητό.

1801

Ο Joseph Jacquard δημιούργησε έναν αυτόματο αργαλειό ελεγχόμενο από διάτρητες κάρτες. Οι διάτρητες κάρτες χρησιμοποιήθηκαν αργότερα ως μέθοδο εισαγωγής για ορισμένους υπολογιστές στις αρχές του 20^{ου} αιώνα.

1899

Ο Nikolai Tesla παρουσίασε το πρώτο τηλεκατευθυνόμενο όχημα. Στο τηλεκατευθυνόμενο σκάφος θα μπορούσε να δοθεί εντολή να ξεκινήσει, να σταματήσει, να στρίψει αριστερά και δεξιά, να ανάψουν τα φώτα και να σβήνουν, ακόμη και να βυθιστεί. Το κοινό δεν εντυπωσιάστηκε. Δημοσιεύματα του Τύπου μίλησαν για "ελέγχου του νου", η οποία δεν βοήθησε την επιστημονική φήμη του Tesla.

Μετά το **1920** η σύγχρονη διατύπωση ενός ανθρωποειδές μηχανήματος αναπτύχθηκε στο στάδιο όπου ήταν δυνατόν να υπάρξει ανθρώπινου μεγέθους ρομπότ με ικανότητα για κοντινές ανθρώπινες σκέψεις και κινήσεις, πριν από την πρώτη προβλεπόμενη χιλιετία. Οι **πρώτες χρήσεις** των σύγχρονων ρομπότ ήταν σε εργοστάσια, όπως τα βιομηχανικά ρομπότ – απλές φτιαγμένες μηχανές ικανές στις εργασίες κατασκευής που επέτρεψαν την παραγωγή χωρίς την ανάγκη για ανθρώπινη βοήθεια.

1926

Κυκλοφόρησε η ταινία του Fritz Lang "Metropolis". Το πρώτο θηλυκό ρομπότ που εμφανίζεται στην οθόνη ονομάστηκε «Μαρία» και θεωρήθηκε ο πρόδρομος του Star War C-3PO.

1932

Το πρώτο παιχνίδι ρομπότ παράχθηκε στην Ιαπωνία. Η «Λίλιπούπολη» ήταν το πρώτο παιχνίδι που περπάτησε και ήταν φτιαγμένο από λευκοσίδηρο και διαμορφωμένο μόλις στα 15 εκατοστά ύψος.

1936

Ο Alan Turing εισάγει την έννοια του θεωρητικού υπολογιστή που ονομάζεται μηχανή Turing. Παρά το γεγονός ότι έχουμε ουσιαστική πρόοδο στην λογική του υπολογιστή που γεννά επίσης νέα σχολεία στα Μαθηματικά.

1941

Ο θρυλικός συγγραφέας επιστημονικής φαντασίας Isaac Asimov γράφει το διήγημα «Liar!» Στο οποίο περιγράφει τους Τρεις Νόμους της Ρομποτικής. Διηγήματά του επαναμεταγλωττίστηκαν στον τόμο "I, Robot" το 1950 - αργότερα αναπαράχθηκε ως ταινία.

Οι τρεις νόμοι της ρομποτικής του Asimov:

- Ένα ρομπότ δεν μπορεί να τραυματίσει ένα ανθρώπινο ον ή, μέσω της αδράνειας, να επιτρέψει σε ένα ανθρώπινο ον να πάθει κακό
- Ένα ρομπότ πρέπει να υπακούει τις εντολές που δόθηκαν από τον άνθρωπο , εκτός εάν τέτοιες εντολές θα έρχονταν σε αντίθεση με τον πρώτο νόμο
- Ένα ρομπότ πρέπει να προστατεύει την ύπαρξή του όσο η προστασία αυτή δεν συγκρούεται με τον Πρώτο ή Δεύτερο Νόμο

1946

Πρώτη παρουσίαση της έννοιας ενός αποθηκευμένου προγράμματος (John von Neumann) και γενικά εκ νέου προγραμματισμού των ηλεκτρονικών υπολογιστών. Ο πρώτος γενικής χρήσης ψηφιακός υπολογιστής, που ονομάστηκε ανεμοστρόβιλος, λύνει το πρώτο πρόβλημα στο M.I.T

1947

Εφευρέθηκε το τρανζίστορ από τους Shockley, Bardeen και Brattain.

1948

Ο Βρετανικής καταγωγής πρωτοπόρος ρομποτικής William Grey Walter δημιούργησε αυτόνομα μηχανήματα που ονομάζονταν Elsie και Elmer που μιμούνται τη ζωή όπως και τη συμπεριφορά με πολύ μικρό κύκλωμα. Αυτά είναι τα πρώτα ρομπότ "χελώνες".

1954

Ο George Devol και Joe Engleberger σχεδίασαν το πρώτο προγραμματιζόμενο ρομπότ «βραχίονα». Αυτό έγινε και το πρώτο βιομηχανικό ρομπότ, συμπληρώνοντας επικίνδυνες και επαναλαμβανόμενες εργασίες σε μια γραμμή συναρμολόγησης στην General Motors (1962).

1956

Ο Alan Newell και Herbert Simon δημιούργησαν το Logic Theoretic, το πρώτο «έμπειρο σύστημα». Χρησιμοποιήθηκε για την επίλυση δύσκολων μαθηματικών προβλημάτων.

1957

Η Σοβιετική Ένωση εγκαινιάζει το «Sputnik», ο πρώτος τεχνητός δορυφόρος σε τροχιά. Αυτό σηματοδοτεί την έναρξη του αγώνα δρόμου του διαστήματος.

1961

Το πρώτο βιομηχανικό ρομπότ (Unimate) εγκαταστάθηκε στο εργοστάσιο αυτοκινήτων General Motors στο New Jersey. Το ρομπότ γραμμή συναρμολόγησης ελέγχεται βήμα-βήμα από εντολές που είναι αποθηκευμένες σε ένα μαγνητικό τύμπανο.

1963

Ο πρώτος τεχνητός ρομποτικός βραχίονας ελεγχόμενος από έναν υπολογιστή σχεδιάστηκε στο Νοσοκομείο Rancho Los Amigos στο Downey, στην California ως ένα εργαλείο για άτομα με ειδικές ανάγκες. Οι έξι αρθρώσεις του Rancho Arm του έδωσε την ευελιξία ενός ανθρώπινου βραχίονα.

1964

Ο IBM 360 γίνεται ο πρώτος υπολογιστής που παράχθηκε μαζικά.

1966

Ένα πρόγραμμα τεχνητής νοημοσύνης ονομαζόμενο ELIZA δημιουργήθηκε στο MIT από τον Joseph Weizenbaum. Η ELIZA λειτουργεί ως ψυχολόγος υπολογιστής που χειρίζεται καταστάσεις χρηστών να διαμορφώνουν ερωτήσεις. Ο Weizenbaum ταραχθηκε με το πόσο γρήγορα οι άνθρωποι πίστεψαν στο μικρό του πρόγραμμα.

1967

Ο Richard Greenblatt γράφει, το MacHack, ένα πρόγραμμα που παίζει σκάκι, απαντώντας σε ένα άρθρο που γράφτηκε από τον Hurbert Dreyfus, όπου πρότεινε, ως κριτική στις προσπάθειες στον τομέα της τεχνητής νοημοσύνης, ότι ένα πρόγραμμα υπολογιστή δεν θα μπορούσε ποτέ να τον νικήσει σε ένα παιχνίδι σκάκι. Όταν ολοκληρώθηκε το πρόγραμμα και ο Dreyfus καλείται να παίζει με τον υπολογιστή, που οδηγούσε το μεγαλύτερο μέρος του παιχνιδιού, τελικά έχασε στο τέλος σε ένα στενό αγώνα. Το πρόγραμμα του Greenblatt έγινε το θεμέλιο για πολλά μελλοντικά προγράμματα σκακιού με αποκορύφωμα το Big Blue το πρόγραμμα σκακιού που νίκησε τον Grand Master σκακιού, Gary Kasparov.

1969

Ο Victor Scheinman, ένας φοιτητής Μηχανολόγων Μηχανικών που εργαζόταν στο Εργαστήριο Τεχνητής Νοημοσύνης Stanford (Stanford Artificial Intelligence Lab) δημιουργεί το Stanford Arm. Ο σχεδιασμός του βραχίονα γίνεται πρότυπο και εξακολουθεί να επηρεάζει το σχεδιασμό στους ρομποτικούς βραχίονες σήμερα.

Την ίδια χρονική περίοδο οι Η.Π.Α. χρησιμοποιούν με επιτυχία την πληροφορική, την ρομποτική και τη διαστημική τεχνολογία για να προσγειωθεί ο Neil Armstrong στο φεγγάρι.

1970

Το Shakey δημιουργήθηκε στο Stanford Research Institute (SRI). Είναι το πρώτο κινητό ρομπότ που ελέγχονται από την τεχνητή νοημοσύνη. Εξοπλισμένο με συσκευές ανίχνευσης οδηγείται από ένα πρόγραμμα επίλυσης προβλημάτων που ονομάζεται TAINIES, το ρομπότ θα μπορούσε να βρει το δρόμο του γύρω από την εφαρμογή των πληροφοριών σχετικά με το περιβάλλον του σε μια διαδρομή. Το Shakey χρησιμοποιεί μια τηλεοπτική κάμερα, λέιζερ, και αισθητήρες χτυπήματος για τη συλλογή δεδομένων, τα οποία στη συνέχεια διαβιβάζεται σε έναν υπολογιστή DEC PDP-10 και PDP-15. Ο υπολογιστής εξέπεμπε σήμα εντολών πίσω στο Shakey - το οποίο στη συνέχεια μετακινούνταν με ταχύτητα 2 μέτρα ανά ώρα.

1977

Η πρώτη ταινία Star Wars απελευθερώνεται. Ταινία του George Lucas εισάγει τους θεατές στο R2D2 και C3PO. Η ταινία δημιούργησε την ισχυρότερη εικόνα ενός ανθρώπου στο μέλλον με ρομπότ από το 1960 και εμπνέει μια γενιά ερευνητών.

Η ASEA, μια ευρωπαϊκή εταιρεία ρομπότ, διαθέτει δύο μεγέθη ηλεκτρικών βιομηχανικών ρομπότ. Και τα δύο ρομπότ χρησιμοποιούν έναν ελεγκτή μικροϋπολογιστή για τον προγραμματισμό και τη λειτουργία τους.

Επίσης την ίδια χρονιά έχουμε τους διαστημικούς εξερευνητές Voyagers 1 και 2 να εκτοξεύονται από το διαστημικό κέντρο Κένεντι.

1981

Ο Takeo Kanade δημιούργησε το βραχίονα άμεσης προώθησης. Είναι το πρώτο που έχει εγκατεστημένους κινητήρες απευθείας στις αρθρώσεις του βραχίονα. Η εξέλιξη αυτή καθιστά τις αρθρώσεις πιο γρήγορες και πολύ πιο ακριβείς από ότι τους προηγούμενους ρομποτικούς βραχίονες.

1982

Η Fanuc της Ιαπωνίας και η General Motors σχηματίζουν μια κοινή επιχείρηση: GM Fanuc. Η νέα εταιρεία πρόκειται να εμπορεύονται τα ρομπότ στη Βόρεια Αμερική.

1986

Η LEGO και το MIT Media Lab συνεργάζονται να φέρουν το πρώτο LEGO βασισμένο στα εκπαιδευτικά προϊόντα στην αγορά. Η LEGO της LOGO χρησιμοποιείται από τις τάξεις χιλιάδων δασκάλων δημοτικού.

Η Honda ξεκινάει ένα ρομποτικό ερευνητικό πρόγραμμα το οποίο άρχισε με την παραδοχή ότι το ρομπότ θα πρέπει να συνυπάρχει και να συνεργάζεται με τους ανθρώπους, κάνοντας ότι ένα άτομο δεν μπορεί να κάνει και καλλιεργώντας μια νέα διάσταση στην κινητικότητα για να ωφελήσει τελικά την κοινωνία. Αυτοί άρχισαν με το πειραματικό "E-series" μεταξύ 1986 και 1991.

1989

Ένα ρομπότ που περπατάει ονομαζόμενο Genghis αποκαλύπτεται από την Ομάδα Mobile Robots του MIT. Γίνεται γνωστό για τον τρόπο που περπατά, γενικά αναφέρεται ως "Genghis βάδισμα».

1992

Ο Dr. John Adler ήρθε με την ιδέα του CyberKnife ένα ρομπότ που εξετάζει τον ασθενή με ακτίνες X για να κοιτάξει για όγκο και ελευθερώνει μια προ-προγραμματισμένη δόση ραδιενέργειας όταν βρεθεί κάποιος όγκος.

1993

Ο Dante ένα ρομπότ με 8 πόδια που περπατάει αναπτύχθηκε στο Πανεπιστήμιο Carnegie Mellon κατέβηκε στο Mt. Erebus, στην Ανταρκτική. Η αποστολή του είναι να συλλέγει δεδομένα από ένα σκληρό περιβάλλον παρόμοιο με αυτό που θα μπορούσαμε να βρούμε σε άλλο πλανήτη. Η αποστολή αποτυγχάνει όταν, μετά από μια σύντομη αξιοπρεπή, ασφαλή πτώση του Dante μέσα στον κρατήρα.

Την επόμενη χρονιά όμως έχουμε το 8-άποδο ρομπότ από το πανεπιστήμιο Carnegie, τον Dante II, που κατέβηκε με επιτυχία στο Mt Spurr να συλλέξει δείγματα ηφαιστειακών αερίων.

1996

Ένα Robo Tuna σχεδιάστηκε και κατασκευάστηκε από τον David Barrett για τη διδακτορική του διατριβή στο MIT. Χρησιμοποιήθηκε για να μελετήσει τον τρόπο με τον οποίο τα ψάρια κολυμπούν.

Επίσης ο Chris Campbell και ο Stuart Wilkinson μετέτρεψαν ένα ατύχημα ζυθοποιίας σε έμπνευση στο πανεπιστήμιο της νότιας Φλόριντας. Το αποτέλεσμα είναι το Gastrobot, ένα ρομπότ που πέπτει οργανική μάζα για να παράγει διοξείδιο του άνθρακα που στη συνέχεια χρησιμοποιείται για ενέργεια. Ονόμασαν την δημιουργία τους "μηχανή μετεωρισμού".

Ακόμα την ίδια χρονιά η Honda κάνει το ντεμπούτο της με το P3, τον καρπό της δεκαετίας μακράς προσπάθειάς της να δημιουργήσει ένα ανθρωποειδές ρομπότ.

Ενώ επιπλέον γίνεται και η τρίτη ετήσια εκδήλωση Robot Wars που πραγματοποιήθηκε στο Fort Mason Center, στο Σαν Φρανσίσκο στην Καλιφόρνια.

1997

Το πρώτο τουρνουά ποδοσφαίρου RoboCup έχει πραγματοποιηθεί στη Ναγκόγια, στην Ιαπωνία. Ο στόχος του RoboCup είναι να έχει μια πλήρως αυτοματοποιημένη ομάδα από ρομπότ για να νικήσει την καλύτερη ποδοσφαιρική ομάδα του κόσμου το έτος 2050.

Η αποστολή Pathfinder προσγειώνεται στον Άρη. Ο ρομποτικός πειρατής της Sojourner, κυλάει σε μια κεκλιμένη επιφάνεια στο αρειανό έδαφος από στις αρχές

Ιουλίου. Συνέχιζε να μεταδίδει δεδομένα από την επιφάνεια του Άρη μέχρι το Σεπτέμβριο.

Ο πρώτος κόμβος του Διεθνούς Διαστημικού Σταθμού τοποθετείται σε τροχιά. Κατά τα επόμενα χρόνια περισσότερα στοιχεία θα συμμετάσχουν σε αυτήν, συμπεριλαμβανομένου ενός ρομποτικού βραχίονα, σχεδιασμένο από την канаδική εταιρεία MD Robotics.

1998

Το Tiger Electronics παρουσιάζει το Furby στην αγορά Χριστουγεννιάτικων παιχνιδιών. Χρησιμοποιώντας μια ποικιλία αισθητήρων αυτό το "εφέ-pet" μπορεί να αντιδράσει με το περιβάλλον του και να επικοινωνήσει χρησιμοποιώντας πάνω από 800 φράσεις στα αγγλικά και στη γλώσσα του "Furbish".

Η LEGO κυκλοφορεί τη πρώτη της εφεύρεση ρομποτικού συστήματος TM 1.0. LEGO ονομάστηκε το προϊόν της σειράς MINDSTORMS μετά τη δημιουργική εργασία του Seymour Papert το 1980.

1999

Η Sony κυκλοφορεί την πρώτη έκδοση του AIBO, ένα ρομποτικό σκύλο με την ικανότητα να μαθαίνει, να ψυχαγωγεί και να επικοινωνεί με τον ιδιοκτήτη του. Πιο εξελιγμένες εκδόσεις ακολούθησαν.

2000

Η Honda λανσάρει ένα νέο ανθρωποειδές ρομπότ τον "ASIMO", την επόμενη γενιά της σειράς του ανθρωποειδούς ρομπότ.

Η Lego κυκλοφορεί το Mindstorms Robotic Invention System TM 2.0.

Τον Οκτώβριο, ο ΟΗΕ εκτιμά ότι υπάρχουν 742.500 βιομηχανικά ρομπότ σε χρήση σε όλο τον κόσμο. Περισσότερα από τα μισά από αυτά που χρησιμοποιούνται στην Ιαπωνία.

2001

Τον Αύγουστο, η FDA καθαρίζει το CyberKnife για τη θεραπεία όγκων οπουδήποτε στο σώμα.

Φτιαγμένο από το MD Ρομποτικής του Καναδά, το σύστημα απομακρυσμένου χειρισμού Διαστημικό Σταθμό (SSRMS) έχει ξεκινήσει με επιτυχία σε τροχιά και αρχίζει τη λειτουργία για να ολοκληρώσει τη συναρμολόγηση του Διεθνούς Διαστημικού Σταθμού.

2003

Στις 10 Ιουνίου - NASA εγκαινιάζει το MER-A "Spirit" rover που προορίζονται για τον Άρη.

Στις 7 Ιουλίου - NASA εγκαινιάζει το MER-B "Opportunity".

Η SONY απελευθερώνει το AIBO ERS-7 ,είναι 3ης γενιάς ρομποτικό κατοικίδιο ζώο.

Το εργαστήριο Ρομποτικής στο Πανεπιστήμιο του Όκλαντ χρησιμοποιούν ρομπότ τους για την έρευνα πλοήγησης B21r.

2004

Epson, είναι το μικρότερο ιπτάμενο ρομπότ που κυκλοφόρησε. Με βάρος 0,35 ουγκιές (10 γραμμάρια) και τη μέτρηση 2,8 ίντσες (70 mm) σε ύψος, το Ιπτάμενο ρομπότ αποκαλείται ως το ελαφρύτερο και μικρότερο ρομπότ ελικόπτερο του κόσμου. Προορίζεται να χρησιμοποιηθεί ως ιπτάμενη κάμερα κατά τη διάρκεια φυσικών καταστροφών.

Το εργαστήριο Ρομποτικής στο Πανεπιστήμιο του Auckland άρχισε να χρησιμοποιεί τελευταίας διδασκαλίας ρομπότ το Shuriken . Αυτό το ρομπότ των 3τροχών μπορεί να κινηθεί προς οποιαδήποτε κατεύθυνση και μπορεί να περιστραφεί.

Στις 4 Ιανουαρίου το ρομπότ Spirit rover προσγειώνεται στον Άρη ενώ στις 23 Ιανουαρίου το δεύτερο ρομπότ Opportunity rover προσγειώνεται με ασφάλεια στο Meridium Planum.

2005

Το Πανεπιστήμιο του Auckland Robotics Group αποκτά ένα νέο, ειδικό εργαστήριο, και χρηματοδοτήθηκε με 7 νέα ρομπότ Pioneer, που συμπληρώνουν την B21r και διάφορα άλλα ερευνητικά ρομπότ, καθώς και εξοπλισμό ρομπότ οπτικοποίησης και εξοπλισμό αλληλεπίδρασης ανθρώπου-ρομπότ.

Ερευνητές στο Πανεπιστήμιο του Cornell ισχυρίζονται ότι έχουν δημιουργήσει το πρώτο ρομπότ αυτο-αναπαραγόμενο, χρησιμοποιώντας μια σειρά ηλεκτρονικών κύβων.

Το ρομπότ Shuriken παίρνει μια αναβάθμιση στο Πανεπιστήμιο του Auckland. Τώρα έχει οπτική μέτρηση χιλιομετρικών αποστάσεων, μια μαγνητική πυξίδα και ένα δεύτερο επεξεργαστή.

2008

Μετά τη πρώτη παρουσίαση το 2002, το δημοφιλές Roomba, ρομποτική ηλεκτρική σκούπα, έχει πουλήσει πάνω από 2,5 εκατομμύρια μονάδες αποδεικνύοντας ότι υπάρχει μεγάλη ζήτηση για αυτό το είδος των εγχώριων ρομποτικής τεχνολογίας.

2010

Το HRP-4 δημιουργήθηκε από τους Kawada και AIST. Είναι δίποδο ρομπότ. Αυτό το ρομπότ έχει μεγάλη ευελιξία και ισορροπία και μοιάζει με έναν άνθρωπο όταν λειτουργεί.

2012

Ο Baxter είναι το ρομπότ της χρονιάς για το 2012. Δημιουργήθηκε από τον Rodney Brooks. Αυτό το ρομπότ κάνει ελαφρά επαναλαμβανόμενα καθήκοντα, όπως η συσκευασία και τη διαλογή.

2013

Ο Grover είναι ένα Greenland Rover. Δημιουργήθηκε από ομάδες φοιτητών που παρακολουθούν μηχανικής στρατοπέδων εκκίνησης. Ο GROVER δημιουργήθηκε για να μεταφέρει ένα επίγειο διαπεραστικό ραντάρ να αναλύει το χιόνι.

Ιστορικά βλέπουμε ότι η επιστήμη της ρομποτικής εξελίσσεται συνέχεια. Κατά την ιστορική αναδρομή είδαμε ότι τα ψηφιακά ελεγχόμενα βιομηχανικά ρομπότ και τα ρομπότ που κάνουν χρήση της τεχνητής νοημοσύνης έχουν κατασκευαστεί από το **1960**.

Εμπόδια στο δρόμο της ρομποτικής

Τα ρομπότ εξελίσσονται μέρα με τη μέρα χάρη στη ραγδαία ανάπτυξη της τεχνολογίας και την επιμονή των επιστημόνων. Σε καμία περίπτωση όμως από την μια μέρα στην άλλη τα ανθρωπόμορφα ρομπότ της επιστημονικής φαντασίας δεν γίνονται πραγματικότητα.

Όσο και αν έχει προχωρήσει η ρομποτική, οι επιστήμονες συναντούν εμπόδια, τα οποία δεν ξέρουν αν θα ξεπεράσουν ποτέ. Τα κυριότερα από αυτά τα προβλήματα είναι τα εξής:

Η Επιδεξιότητα: καθημερινά οι άνθρωποι κάνουν κινήσεις που απαιτούν μεγάλη προσοχή και επιδεξιότητα, παρόλο που μας φαίνεται σχετικά απλές. Όπως για

παράδειγμα να δέσουμε τα κορδόνια μας ή να μεταφέρουμε ένα βάζο χωρίς να το σπάσουμε. Τα ρομπότ δεν μπορούν να τα κάνουν όλα αυτά.

Η Όραση: σίγουρα τα ρομπότ μπορούν να βλέπουν και μάλιστα εξαιρετικά καθαρά χάρη στις υπερσύγχρονες κάμερες που έχουν ενσωματωμένες. Είναι όμως προγραμματισμένα να βλέπουν συγκεκριμένα πράγματα. Για παράδειγμα αν δουν ένα χρώμα ή ένα αντικείμενο που οι επιστήμονες δεν έχουν συμπεριλάβει στο πρόγραμμά τους, δεν θα μπορούν να το αναγνωρίσουν.

Τα συναισθήματα και ο αυθορμητισμός: τα ρομπότ είναι προγραμματισμένα να πραγματοποιούν αποτελεσματικά κάποιες συγκεκριμένες ενέργειες. Η ζωή όμως δεν πηγαίνει πάντα σύμφωνα με το πρόγραμμα. Συχνά χρειάζεται να διακόψουμε κάτι που κάνουμε για να βοηθήσουμε έναν συνάνθρωπό μας κάτι που δεν θα σκεφτεί ποτέ να κάνει ένα ρομπότ.

Θα πρέπει να περάσουν πολλά χρόνια μέχρι να αποκτήσουμε ανθρωπόμορφα ρομπότ που θα λειτουργούν σαν εμάς.

1.1.2 Βασικά στοιχεία και αρχές των ρομπότ, ταξινόμηση σε είδη

Ένα ρομπότ συγκροτείται από δύο συστήματα, το μηχανικό (στο οποίο περιλαμβάνεται το σύστημα κίνησης) και το ηλεκτρονικό (στο οποίο υπάγεται και η επαναπρογραμματιζόμενη μνήμη του). Υπάρχουν διάφορα κριτήρια διάκρισης και αντίστοιχες κατηγοριοποιήσεις των ρομπότ. Μία από αυτές είναι η διάκρισή τους σε τρεις, επί του παρόντος, "γενιές". Στην πρώτη γενιά κατατάσσονται ρομπότ με περιορισμένη ευελιξία, που διευθύνονται από τον άνθρωπο, όπως, για παράδειγμα, οι απλοί "χειριστές" σχετικά απλά εργαλεία που επιτρέπουν, για παράδειγμα, τη μετακίνηση επικίνδυνων αντικειμένων (π.χ. ραδιενεργών υλικών). Στη δεύτερη γενιά κατατάσσονται τα ρομπότ που είναι εφοδιασμένα με σταθερό πρόγραμμα δράσης και ρομπότ που λαμβάνουν εντολές από κάποιο σύστημα αριθμητικού ελέγχου. Στην τρίτη γενιά κατατάσσονται ρομπότ που είναι εφοδιασμένα:

- με αισθητήριες "πληροφορίες" από το περιβάλλον,
- με διάταξη επεξεργασίας των πληροφοριών και
- με κινητήριο σύστημα εκτέλεσης εργασιών.

Χρήση των ρομπότ

Ευρύτατη χρήση ρομπότ γίνεται σε πάρα πολλούς παραγωγικούς τομείς και κυρίως στη βιομηχανία (βιομηχανική ρομποτική), στην ιατρική, την αεροναυπηγική, την αεροδιαστημική κ.α, γεγονός που έδωσε σημαντική ώθηση στην περαιτέρω ανάπτυξη της βιομηχανίας των ρομπότ, ιδιαίτερα στην Ιαπωνία και τις ΗΠΑ. Οι κυριότερες εφαρμογές των βιομηχανικών ρομπότ, μέχρι σήμερα, ήταν οι ηλεκτροσυγκολλήσεις, οι εφαρμογές σε εργασίες πρεσαρίσματος, οι συναρμολογήσεις, οι βαφές με ψεκασμό και η επεξεργασία επιφανειών σε τροφοδοτήσεις εργαλειομηχανών, σε μορφοποιήσεις πλαστικών σε μήτρες κ.α. Από τα μέσα περίπου της δεκαετίας του 1980 η χρήση των ρομπότ γενικεύτηκε στο πλαίσιο της ανάπτυξης των Ολοκληρωμένων Συστημάτων Παραγωγής (Computer-Integrated Manufacturing), αυτοματοποιημένων και ευέλικτων εργοστασίων, στα οποία οι εργαλειομηχανές

μπορούν να επαναπρογραμματίζονται ταχύτατα για την παραγωγή νέων ή διαφοροποιημένων προϊόντων. Πρωτοποριακά εργοστάσια ως προς την ευρύτατη χρήση ρομπότ θεωρούνται το εργοστάσιο της General Motors στο Hamtramck, το Buick City στο Flint του Michigan, το εργοστάσιο της IBM στο Lexington κ.α. Πέρα, όμως, από τη βιομηχανία ευρύτατη χρήση ρομπότ γίνεται και σε μη μεταποιητικές εφαρμογές, όπως, για παράδειγμα, σε πυρηνικούς σταθμούς, υποθαλάσσιες έρευνες, σε ιατρικές εφαρμογές, στην εξόρυξη πετρελαίου κ.λπ. Ρομπότ επίσης με την ευρεία έννοια μπορούν να θεωρηθούν και τα αυτοματοποιημένα διαστημόπλοια (μη επανδρωμένα), που χρησιμοποιούνται για διαστημικές έρευνες, καθώς και ειδικές κατασκευές όπως, για παράδειγμα, η σοβιετική σεληνάκατος Lunokhod-1, η οποία καθοδηγούνταν με ασύρματο από τη Γη. Τέλος, σε πειραματική και όχι εμπορική βάση έχουν κατασκευαστεί οικιακά ρομπότ που καθαρίζουν το σπίτι, σερβίρουν ποτά ή "παίζουν" με τα παιδιά. Η ανάπτυξη του κλάδου της τεχνητής νοημοσύνης (artificial intelligence) κατά τη δεκαετία του 1980 άνοιξε ευρύτατες προοπτικές εφαρμογής της στη ρομποτική. Όπως είναι γνωστό, η τεχνητή νοημοσύνη αποτελεί ένα από τα σημαντικότερα πεδία έρευνας της πληροφορικής και αφορά την κατασκευή συστημάτων αυτοματισμού εφοδιασμένων με ικανότητα μάθησης, δυνατότητα κατανόησης της φυσικής γλώσσας, ικανότητα αξιολόγησης στοιχείων, λήψης αποφάσεων κ.α. Οι σχετικές έρευνες στον τομέα της ρομποτικής αφορούν την κατασκευή ρομπότ τα οποία πέρα από τις βασικές αισθήσεις, όπως η αφή και η όραση, θα είναι εφοδιασμένα με αντιληπτικές ικανότητες (για παράδειγμα, αντίληψη σχημάτων, μορφών, εικόνων κ.λπ.), με ικανότητα διεξαγωγής λογικών συνειρμών και εξαγωγής συμπερασμάτων, καθώς και με δυνατότητες ανακατανομής δεδομένων ανάλογα με τη χρήση για την οποία ζητούνται καθώς και με ικανότητα αυτό-διόρθωσης. Η ανάπτυξη της προηγμένης τεχνολογίας ρομπότ αναμένεται ότι θα βοηθήσει σημαντικά στην επίλυση προβλημάτων και στην ολοκλήρωση εργασιών σε χώρους που είναι δύσκολα προσπελάσιμοι για τον άνθρωπο. Ωστόσο εκφράζονται και φόβοι για τη δυσκολία ελέγχου των συστημάτων αυτών, καθώς θα είναι εφοδιασμένα με ικανότητα ανάπτυξης σχετικής αυτονομίας κατά τη λειτουργία τους.

Οι 3 νόμοι της ρομποτικής

Οι τρεις νόμοι της ρομποτικής είναι κανόνες στους οποίους υπακούν τα περισσότερα ρομπότ με εγκέφαλο ποζιτρονίου που εμφανίζονται στα έργα επιστημονικής φαντασίας του συγγραφέα Ισαάκ Ασίμωφ, ενώ χρήση τους έχουν κάνει και άλλοι δημιουργοί επιστημονικής φαντασίας. Οι νόμοι αυτοί πρωτοδιατυπώθηκαν από τον Ασίμωφ στο διήγημα "Runaround" (1942) και είναι οι εξής:

1. Το ρομπότ δε θα κάνει κακό σε άνθρωπο, ούτε με την αδράνειά του θα επιτρέψει να βλαφτεί ανθρώπινο όν
2. Το ρομπότ πρέπει να υπακούει τις διαταγές που του δίνουν οι άνθρωποι, εκτός αν αυτές οι διαταγές έρχονται σε αντίθεση με τον πρώτο νόμο

3. Το ρομπότ οφείλει να προστατεύει την ύπαρξή του, εφόσον αυτό δεν συγκρούεται με τον πρώτο και τον δεύτερο νόμο

Σε μεταγενέστερα μυθιστορήματα του Ασίμωφ, οι τρεις νόμοι της ρομποτικής συμπληρώθηκαν από το ρομπότ Ντάνιελ Όλιβοου με τον μηδενικό νόμο της ρομποτικής:

- Το ρομπότ δε θα κάνει κακό στην ανθρωπότητα, ούτε με την αδράνειά του θα επιτρέψει να βλαφτεί η ανθρωπότητα,

οπότε και ο πρώτος νόμος συμπληρώθηκε ανάλογα (παρόμοιες προσαρμογές έγιναν και στον δεύτερο και τρίτο νόμο):

- Το ρομπότ δε θα κάνει κακό σε άνθρωπο, ούτε με την αδράνειά του θα επιτρέψει να βλαφτεί ανθρώπινο όν, εφόσον αυτό δεν αντιτίθεται στο μηδενικό νόμο.

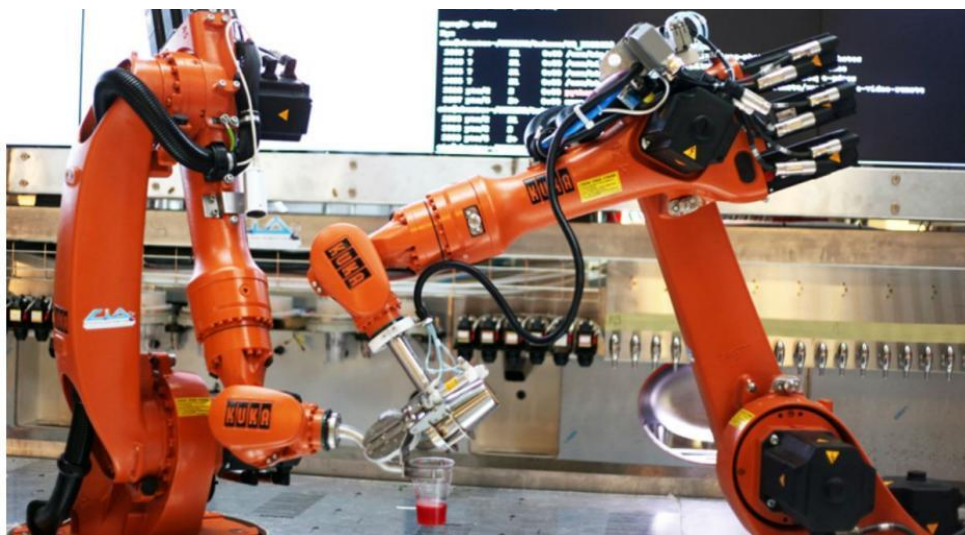
Στους νόμους αυτούς, αλλά και στις παραβιάσεις τους στηρίχθηκαν τα διηγήματα για ρομπότ του Ασίμωφ, αλλά και πολλών άλλων συγγραφέων.

Είδη Ρομπότ

Κατά την πολυετή εξέλιξη της επιστήμης της ρομποτικής προέκυψαν διάφορα είδη ρομποτικών μηχανισμών, οι οποίοι διαφέρουν σημαντικά στην μορφή, αποτελούνται όμως από αντίστοιχα επιμέρους υποσυστήματα. Τέτοια υποσυστήματα είναι το μηχανολογικό υποσύστημα, το υποσύστημα αίσθησης και το σύστημα ελέγχου.

Τα σπουδαιότερα είδη ρομπότ είναι:

- Ρομπότ σταθερής βάσης : τα ρομπότ αυτά αποτελούνται από διαδοχικά στερεά σώματα (σύνδεσμοι) που συνδέονται μέσω αρθρώσεων σχηματίζοντας μια κινηματική αλυσίδα. Η αλυσίδα αυτή έχει το ένα άκρο της (βάση) σταθερά συνδεδεμένο με κάποιο σημείο του περιβάλλοντος χώρου. Η μορφή αυτή ρομπότ είναι η παραδοσιακή μορφή ενός βιομηχανικού ρομποτικού βραχίονα και περιλαμβάνει τον βραχίονα, τον καρπό και το εργαλείο.



Εικόνα 1.3: βιομηχανικός ρομποτικός βραχίονας

- Κινούμενα ρομπότ : ως κινητά ρομπότ χαρακτηρίζονται όλα εκείνα τα ρομπότ που έχουν τη δυνατότητα να μετακινήσουν όλα τα σημεία του μηχανισμού τους. Η δυνατότητα αυτή προσφέρεται από ειδικά συστήματα προώθησης, τα οποία μπορεί να είναι είτε απλά (όπως τροχοί) είτε πολύπλοκα (όπως jet, προπέλες, μηχανικά πόδια). Τα κινούμενα ρομπότ διακρίνονται σε άλλες επιμέρους κατηγορίες ανάλογα με τον βαθμό αυτονομίας τους. Έτσι έχουμε :
 - AGVs: τα AGVs (Automatic Guided Vehicles) έχουν περιορισμένη αυτονομία κίνησης, δεδομένου ότι η τροχιά τους είναι προκαθορισμένη μέσω καλωδίων στο έδαφος ή πομπών στον περιβάλλον χώρο.
 - Αυτόνομα Έντροχα Ρομπότ: τα ρομπότ αυτά λειτουργούν με αρκετά υψηλό βαθμό αυτονομίας. Πιο συγκεκριμένα μπορούν να λειτουργούν χωρίς συνεχή εξωτερική επίβλεψη και είναι ικανά να εκτελούν εργασίες αυτόνομα δεχόμενα μόνο ορισμένες υψηλού επιπέδου εντολές.
 - Βαδίζοντα Ρομπότ: τα ρομπότ αυτά χρησιμοποιούν μηχανικά πόδια για την κίνησή τους και όχι συμβατικούς τροχούς όπως στις προηγούμενες δύο κατηγορίες. Τα κυριότερα πλεονεκτήματα της συγκεκριμένης υλοποίησης είναι η μεγάλη δυνατότητα αποφυγής εμποδίων και η ικανότητα αναρρίχησης σε ανώμαλα εδάφη και μη επίπεδες επιφάνειες. Από τα πιο συνηθισμένα ρομπότ αυτής της κατηγορίας είναι τα δίποδα ενώ δεν αποκλείονται και εφαρμογές με περισσότερα από δύο πόδια πχ. ρομπότ που μοιάζουν και κινούνται όπως οι αράχνες.
 - ROVs: τα ROVs (Remotely Operated Vehicle) ανήκουν στην κατηγορία των επανδρωμένων υποβρύχιων ρομπότ. Όπως δηλώνει και το όνομά τους δεν έχουν μεγάλο βαθμό αυτονομίας

μιας και είναι συνδεδεμένα με το μητρικό πλοίο μέσω καλωδίου, το οποίο καλύπτει τις ανάγκες του ρομπότ σε ενέργεια και επικοινωνίες. Τα ρομπότ αυτού του τύπου έχουν σχήμα κουτιού και κινούνται γενικά σε χαμηλές ταχύτητες.

- AUVs: τα AUVs (Autonomous Underwater Vehicle), αντίθετα με τα ROVs, είναι πλήρως αυτόνομα και συνεπώς δεν έχουν την ανάγκη καλωδίου. Για τις ανάγκες τροφοδοσίας (ενέργεια) χρησιμοποιούνται ειδικές μπαταρίες κάτι που θέτει και περιορισμούς στη λειτουργία των ρομπότ αυτών. Τα AUVs έχουν σχήμα τορπιλών και μπορούν να κινούνται με αρκετά μεγάλες ταχύτητες.
- Εναέρια Ρομπότ: πρόκειται για μη επανδρωμένα ιπτάμενα ρομπότ, όπως ελικόπτερα και αεροπλάνα. Τα ρομπότ αυτά έχουν διαρκώς αυξανόμενες εφαρμογές, όμως εξαιτίας της μειωμένης ακόμα σταθερότητας και ασφάλειας στη συμπεριφορά τους χρησιμοποιούνται για στρατιωτικούς κυρίως σκοπούς.



Εικόνα 1.4: βαδίζοντα ρομπότ, αυτόνομα έντροχα ρομπότ, AGVs, ROVs, εναέρια ρομπότ, AUVs

1.1.3 Βασικά κατασκευαστικά και επιστημονικά στοιχεία

Σύμφωνα με το Robot Institute of America, ως ρομπότ μπορούμε να ορίσουμε ένα μηχανισμό σχεδιασμένο ώστε μέσω προγραμματιζόμενων κινήσεων να μεταφέρει υλικά, τεμάχια, εργαλεία ή ειδικευμένες συσκευές με σκοπό την επιτέλεση ποικιλίας εργασιών. Ένας τέτοιος μηχανισμός περιλαμβάνει συνήθως τις ακόλουθες συνιστώσες:

- Ένα μηχανολογικό υποσύστημα το οποίο ενσωματώνει τη δυνατότητα του ρομπότ για εκτέλεση έργου. Το υποσύστημα αυτό αποτελείται από μηχανισμούς που επιτρέπουν στο ρομπότ να κινείται, όπως αρθρώσεις, συστήματα μετάδοσης κίνησης, επενεργητές-κινητήρες, οδηγούς κλπ.
- Ένα υποσύστημα αίσθησης, μέσω του οποίου το ρομπότ συγκεντρώνει πληροφορίες για την κατάσταση στην οποία βρίσκονται τόσο το ίδιο όσο και το περιβάλλον. Το υποσύστημα αυτό εκτός των άλλων είναι υπεύθυνο για την αποδοχή των εξωτερικών εντολών, την επεξεργασία τους, την μετάφρασή τους σε ηλεκτρική ισχύ που θα δοθεί στους κινητήρες του ρομπότ, καθώς επίσης και την παραγωγή σημάτων εξόδου που θα πληροφορούν για την κατάσταση του συστήματος. Στο υποσύστημα αίσθησης περιλαμβάνονται όργανα μετρήσεως, αισθητήρες, ηλεκτρονικά στοιχεία κλπ.
- Ένα σύστημα ελέγχου, το οποίο συνδυάζει κατάλληλα την αίσθηση με τη δράση, έτσι ώστε το ρομπότ να λειτουργεί αποτελεσματικά και με τον επιθυμητό τρόπο. Ο ελεγκτής του ρομπότ επιβλέπει και συντονίζει ολόκληρο το σύστημα για τη σχεδίαση και υλοποίησή του όπου δεν απαιτείται ο συνδυασμός γνώσεων από άλλες γνωστικές περιοχές όπως είναι ο αυτόματος έλεγχος, η τεχνητή νοημοσύνη, η επιστήμη των υπολογιστών κλπ.

Η εξέλιξη της ρομποτικής σήμερα

Τα τελευταία χρόνια έχουν αναπτυχθεί ρομπότ που προσπαθούν να μιμηθούν ανθρώπινα χαρακτηριστικά όπως το περπάτημα και την όραση. Τα αποτελέσματα είναι ενθαρρυντικά και σίγουρα τις επόμενες δεκαετίες θα δούμε σημαντικά επιτεύγματα στον τομέα της ρομποτικής. Με τη χρήση της ρομποτικής τα ρομπότ έχουν εξελιχθεί, ώστε να δημιουργούνται ανθρωποειδή ρομπότ που να μοιάζουν σε μεγάλο βαθμό με τους ανθρώπους. Οι παράγοντες της ομοιότητας τους είναι το πολύ ρεαλιστικό δέρμα, η προηγμένη υδραυλική τεχνολογία και οι ταχύτεροι υπολογιστές. Τα ρομπότ τέτοιου είδους θα μοιάζουν εξωτερικά μεν με τους ανθρώπους αλλά βέβαια δεν θα έχουν παρόμοιες ψυχικές ικανότητες.



Εικόνα 1.5: ανδροειδές ρομπότ

Η ρομποτική στην ιατρική

Η ρομποτική ωφελεί πολλούς τομείς και μάλιστα την Ιατρική που είναι η πιο ευνοημένη της ρομποτικής. Τα οφέλη της είναι τα εξής:

- **Λιγότερο προσωπικό**
Επειδή τα χειρουργικά ρομπότ μπόρεσαν να αναλάβουν δουλειά στο λειτουργικό δωμάτιο οι μελλοντικές χειρουργικές επεμβάσεις θα απαιτήσουν λιγότερο προσωπικό και θα είναι σε θέση να εκτελέσουν μια αποτελεσματικότερη και ασφαλέστερη χειρουργική επέμβαση.
- **Χειρουργική επέμβαση από απόσταση**
Με την βελτίωση στις τηλεπικοινωνίες και την ταχύτητα της μεταφοράς στοιχείων, η ρομποτική χειρουργική επέμβαση μπορεί ακόμη και να γίνει από απόσταση. Αυτό σημαίνει ότι ο χειρουργός θα μπορεί να εκτελέσει τη λειτουργία ακόμη και αν είναι σε μια άλλη πόλη ή μια άλλη χώρα μακριά από τον ασθενή.
- **Μειωμένο τραύμα και γρηγορότερη υπομονετική αποκατάσταση**
Επειδή η ρομποτική χειρουργική επέμβαση θα επέτρεπε τις διαδικασίες που χρειάζονται μόνο τις μικρές τομές στο σώμα του ασθενή, το ποσοστό αποκατάστασης των ασθενών θα επιταχυνόταν. Αυτό θα μετέφραζε στο λιγότερο πόνο κατά τη διάρκεια και μετά από το χειρουργείο καθώς επίσης και λιγότερους κινδύνους και περιπλοκών για τον ασθενή. Τα υπάρχοντα χειρουργικά συστήματα είναι τα εξής: 1.DA Vinci, 2.Σύστημα ZEUS, 3.ΑΙΣΩΠΟΣ



Εικόνα 1.6: ρομπότ στην Ιατρική

Τα ρομπότ στο διάστημα

Η NASA έστειλε στο διάστημα πολλά ρομπότ με σκοπό να κάνουν έρευνες σχετικά με τους άλλους πλανήτες. Αυτές οι έρευνες ήταν πολύ επικίνδυνες για να επανδρώσουν διαστημόπλοιο και να στείλουν ανθρώπους εκεί και για το λόγο της επικινδυνότητάς του η NASA αναγκάστηκε να προσφύγει σε εναλλακτική λύση όσον αφορά το θέμα αυτό, και η λύση ήταν η χρήση των ρομπότ. Τα ρομπότ ήταν συνδεδεμένα με τη NASA και είχαν πάνω τους κάμερες και ελέγχονταν από τους ανθρώπους της NASA και μελετούσαν τη μορφολογία του εδάφους και του υπεδάφους, εξέταζαν για τυχόν ζωή στον πλανήτη ή ύπαρξη άλλων στοιχείων στην ατμόσφαιρα του πλανήτη αυτού. Επίσης η NASA έχει χρησιμοποιήσει και χρησιμοποιεί είδη ρομποτικό έλεγχο σε τηλεσκόπια με τα οποία παρατηρούν τους

άλλους αστέρες, πλανήτες, γαλαξίες και παντός είδους κινήσεων στο σύμπαν. Ρομπότ επίσης με την ευρεία έννοια μπορούν να θεωρηθούν και τα αυτοματοποιημένα διαστημόπλοια (μη επανδρωμένα), που χρησιμοποιούνται για διαστημικές έρευνες καθώς και ειδικές κατασκευές όπως π.χ. η σοβιετική σεληνάκατος «Λουννοχόντ-1» η οποία καθοδηγούνταν με ασύρματο από τη Γη.



Εικόνα 1.7: ρομπότ στο διάστημα

Η ρομποτική στις βιομηχανίες

Η ρομποτική χρησιμοποιείται στις διάφορες βιομηχανίες και παράγουν ποικίλα προϊόντα. Οι δουλειές για τις οποίες αντικαταστάθηκαν οι άνθρωποι με τους ρομποτικούς βραχίονες ή διάφορα άλλα εξαρτήματα έχουν σχέση με τα ρομπότ. Οι κυριότερες εφαρμογές των βιομηχανικών ρομπότ μέχρι σήμερα ήταν οι ηλεκτροσυγκολλήσεις, οι εφαρμογές σε εργασίες πρεσαρίσματος, οι συναρμολογήσεις, οι βαφές με ψεκασμό και η επεξεργασία επιφανειών σε τροφοδοτήσεις εργαλειομηχανών κ.α. Από τα μέσα της δεκαετίας του 1980 η χρήση των ρομπότ γενικεύτηκε στο πλαίσιο της ανάπτυξης των λεγόμενων ολοκληρωμένων συστημάτων παραγωγής, αυτοματοποιημένων και ευέλικτων εργοστασίων στα οποία οι εργαλειομηχανές μπορούν να επανα-προγραμματίζονται ταχύτατα για την παραγωγή νέων ή διαφοροποιημένων προϊόντων. Πρωτοποριακά εργοστάσια ως προς την ευρύτατη χρήση ρομπότ θεωρούνται το εργοστάσιο της General Motors στο Χάμτραμек, το Buick City στο Φλιντ του Μίσιγκαν, το εργοστάσιο της IBM στο Λέξινγκτον κ.α. Πέρα από τη βιομηχανία ευρύτατη χρήση ρομπότ γίνεται και σε μη μεταποιητικές εφαρμογές.

Η χρήση της ρομποτικής για ψυχαγωγία

Υπάρχουν διάφορες εταιρίες που παράγουν παιχνίδια και άλλα είδη διασκέδασης. Σε πειραματικό και όχι σε εμπορικό στάδιο έχουν κατασκευαστεί οικιακά ρομπότ που καθαρίζουν το σπίτι, σερβίρουν ποτά ή παίζουν με τα παιδιά. Μερικά από αυτά τα ρομπότ που θα μπορούν να παίζουν τα παιδιά είναι τα γνωστά robot dog, τα οποία βλέπουμε στις τηλεοράσεις μας και τους υπολογιστές μας.



Εικόνα 1.8: robot dog

Το έξυπνο σπίτι

Η δημιουργία ενός έξυπνου σπιτιού είναι ένα γνωστό όνειρο από παλιά. Τώρα με την ανάπτυξη της ρομποτικής το όνειρο αυτό γίνεται πραγματικότητα. Σήμερα έχουν δημιουργηθεί πολλές συσκευές που να είναι απλά ένα μέρος του έξυπνου σπιτιού. Συσκευές που παλιότερα φαίνονταν κουραστικές στην χρήση τους τώρα έχουν γίνει απλούστερες.

Ρομποτικοί βραχίονες

Σήμερα τα ρομπότ χρησιμοποιούνται κυρίως για να εκτελέσουν βαριές, ανιαρές, επικίνδυνες και επαναλαμβανόμενες εργασίες κυρίως στις βιομηχανίες. Υπολογίζεται ότι το 90% των ρομπότ είναι εγκατεστημένα σε βιομηχανικούς χώρους, όπως οι βραχίονες που συναρμολογούν και συγκολλούν τον σκελετό ενός αυτοκινήτου στην γραμμή παραγωγής. Στόχος της εφαρμογής των ρομποτικών συστημάτων στη βιομηχανία αποτελούν η υποκατάσταση της χειρωνακτικής εργασίας, η βελτίωση και η σταθεροποίηση της ποιότητας και η αύξηση της παραγωγικότητας. Πρωτοπόρες χώρες στην εγκατάσταση ρομποτικών συστημάτων αποτελούν η Ιαπωνία (352.200 βιομηχανικά ρομπότ στο τέλος του 2004), η Γερμανία (121.500), η Αμερική (121.300), η Ιταλία (53.100), η Γαλλία (28.400) και η Αγγλία (14.600). Συνολικά στην Ευρωπαϊκή Ένωση υπολογίζεται ότι στο τέλος του 2004 υπήρχαν 266.100 εγκατεστημένα ρομπότ στη βιομηχανία.



Εικόνα 1.9: ρομποτικοί βραχίονες

1.1.4 Η εκπαιδευτική ρομποτική- η εξέλιξη της LOGO

Τα τελευταία χρόνια η εκπαιδευτική ρομποτική έχει αξιοποιηθεί, κυρίως στο εξωτερικό αλλά και σε πειραματικά προγράμματα στην Ελλάδα, σε όλες τις βαθμίδες της εκπαίδευσης (από το νηπιαγωγείο ως το πανεπιστήμιο). Η χρήση της προσφέρει ένα περιβάλλον μάθησης που επιτρέπει στους μαθητές να ελέγξουν ένα χειροπιαστό μοντέλο χρησιμοποιώντας ειδικές γλώσσες προγραμματισμού. Αυτό τους δίνει τη δυνατότητα να ασχοληθούν ενεργά με την επίλυση προβλημάτων και να δημιουργήσουν τις δικές τους προγραμματιζόμενες κατασκευές (Resnick και άλλοι 1996). Σημαντικό στοιχείο της εκπαιδευτικής ρομποτικής είναι ότι το περιβάλλον διδασκαλίας της είναι παράλληλα και περιβάλλον παιχνιδιού καθώς και το μεγάλο ενδιαφέρον των παιδιών για τις νέες τεχνολογίες και ειδικότερα τους ηλεκτρονικούς υπολογιστές. Χάρη στα παραπάνω, το αντικείμενο της ρομποτικής μπορεί να έχει μεγάλα ποσοστά αποδοχής από τους μαθητές και με τα κατάλληλα διδακτικά πρωτόκολλα, σημαντικά μαθησιακά αποτελέσματα.

Θεωρητικό υπόβαθρο της εκπαιδευτικής ρομποτικής

Ενώ στους περασμένους αιώνες , το παιχνίδι των παιδιών θεωρούταν κάτι ανούσιο, νεότερες παιδαγωγικές θεωρίες, με κύριους εκφραστές τους Piaget και Papert, αντιμετωπίζουν το παιχνίδι σαν ένα πολύ σημαντικό εργαλείο μάθησης. Η ραγδαία εξέλιξη της τεχνολογίας στο δεύτερο μισό του 20ου αιώνα έδωσε νέα εργαλεία για την εξέλιξη των παιδαγωγικών αυτών θεωριών, αλλά και ανάποδα οι παιδαγωγικές θεωρίες έδωσαν ιδέες για την ανάπτυξη καινούριων τεχνολογικών επιτευγμάτων (Martin και άλλοι 2000). Γέννημα αυτής της αλληλεπίδρασης είναι η εκπαιδευτική ρομποτική.

Η χρήση της ρομποτικής στην εκπαίδευση βασίζεται κυρίως στις θεωρίες του κονστρουκτιβισμού (constructivism) και του κονστρουξιονισμού (constructionism) και υιοθετεί την προσέγγιση της μάθησης με βάση συνθετικές εργασίες (project-based learning).

Προγενέστερη των άλλων δύο θεωρείται η αρχή του κονστρουκτιβισμού. Ετυμολογικά προέρχεται από την λέξη construct (κατασκευή) και αυτό γιατί υποστηρίζει ότι η γνώση δεν μεταδίδεται απλά, αλλά κατασκευάζεται – δομείται (γι' αυτό το λόγο ονομάζεται και εποικοδομητισμός ή οικοδομισμός) μέσα από την

εμπειρία, χρησιμοποιώντας ταυτόχρονα την προϋπάρχουσα γνώση, για να ερμηνευθεί η νέα πληροφορία. Κύριος εκπρόσωπος του εποικοδομητισμού είναι ο Ελβετός βιολόγος και ψυχολόγος Jean Piaget. Σύμφωνα με τον Piaget, τα παιδιά δημιουργούν θεωρίες με βάση την εμπειρία τους που πολλές φορές είναι δογματικές. Οι αντιλήψεις αυτές είναι πολύ δύσκολο να αλλάξουν μόνο και μόνο επειδή κάποιος άλλος (εκπαιδευτικός) τους λέει μια άλλη καλύτερη, αφού δημιουργούν μηχανισμούς άμυνας απέναντι στη νέα θεωρία που αντιπαραβάλλεται με τη δική τους. Αυτό όμως δεν σημαίνει ότι αυτές οι αντιλήψεις δεν αλλάζουν και δεν εξελίσσονται μέσω της επαφής με το περιβάλλον. Η γνώση δεν είναι μια απλή μετάδοση πληροφοριών που ο μαθητής θα απομνημονεύσει και θα ανακτήσει για να την εφαρμόσει όταν χρειαστεί. Η γνώση είναι εμπειρία που αποκτάται μέσω της αλληλεπίδρασης με πρόσωπα και πράγματα (Ackermann 2001, Piaget 1974).

Στις θεωρίες του κονστρουκτιβισμού βασίστηκε ο Seymour Papert για να διατυπώσει την θεωρία του κονστρουξιονισμού (κατασκευαστικός εποικοδομητισμός). Ο Papert δέχεται ότι η γνώση κατασκευάζεται και προχωρώντας ένα βήμα παρακάτω δίνει τα εργαλεία της δόμησης της γνώσης. Η θεωρία του εστιάζει πιο πολύ στον τρόπο μάθησης, παρά στα γνωστικά δυναμικά όπως ο Piaget. Υποστηρίζει ότι οι μαθητευόμενοι οικοδομούν πιο αποτελεσματικά τη γνώση όταν εμπλέκονται ενεργά στη σχεδίαση και κατασκευή (χειρωνακτική και ψηφιακή) πραγματικών αντικειμένων που έχουν νόημα για τους ίδιους είτε αυτά είναι κάστρα από άμμο, είτε κατασκευές LEGO και προγράμματα υπολογιστών (Papert, 1991).

Τα εργαλεία που προτείνει ο Papert είναι κυρίως οι υπολογιστές και μέσω αυτών διάφορα εκπαιδευτικά εργαλεία με κυριότερα αυτά της γλώσσας προγραμματισμού LOGO που ο ίδιος εφεύρε και τα παιχνίδια Lego Mindstorms (Papert 1980).

Τέλος, η ρομποτική στην εκπαίδευση είναι ένα χαρακτηριστικό εργαλείο διδασκαλίας για τη μάθηση με βάση συνθετικές εργασίες (project-based learning, Carbonaro, Rex & Chambers, 2004). Το να σχεδιάσουν οι μαθητές ένα ρομπότ, έστω και με πολύ βασικές λειτουργίες, μπορεί να συμβάλει τα μέγιστα στην ανάπτυξη της δημιουργικότητας τους και της ικανότητας τους να λύνουν προβλήματα και κατ' επέκταση στην ανάπτυξη κριτικής σκέψης (Druin & Hendler, 2000). Στη μάθηση

βασισμένη σε project οι μαθητές δουλεύουν συνεργατικά για να καταλάβουν το πρόβλημα, το οποίο προέρχεται κατά κανόνα από την καθημερινή ζωή τους και εφάπτεται των ενδιαφερόντων τους, με απώτερο σκοπό να δώσουν χειροπιαστές λύσεις (Lui & Hsiao, 2002, Brown, Collins, & Duguid, 1989). Έτσι, μέσω της συνεργατικής μάθησης δίνεται η δυνατότητα να διορθωθούν αθέλητες κοινωνικές και εκπαιδευτικές προκαταλήψεις που ευνοεί ο σχολικός ανταγωνισμός.

Πρακτικές της εκπαιδευτικής ρομποτικής

Η εκπαιδευτική ρομποτική στην πράξη μπορεί να χρησιμοποιηθεί με δύο τρόπους: είτε σαν αντικείμενο μελέτης, είτε σαν εργαλείο μάθησης.

Στην πρώτη περίπτωση η ρομποτική μελετάται σαν ένα ξεχωριστό πεδίο. Περιλαμβάνει εκπαιδευτικές δραστηριότητες που σκοπό έχουν να εμπλέξουν τους μαθητές στην μελέτη και επίλυση προβλημάτων που εστιάζουν στην κατασκευή ρομπότ, στον προγραμματισμό τους και στην τεχνητή νοημοσύνη. Από την άλλη, στα πρωτόκολλα όπου η ρομποτική χρησιμοποιείται σαν εργαλείο μάθησης, η κατασκευή ρομπότ και ο προγραμματισμός δεν είναι αυτοσκοπός αλλά το εργαλείο για τη διδασκαλία άλλων σχολικών αντικειμένων, κυρίως μαθηματικών, φυσικών επιστημών, πληροφορικής, τεχνολογίας αλλά και σε διεπιστημονικές εργασίες. Ωστόσο, αυτός ο διαχωρισμός δεν είναι πάντα εύκολος και σαφής. Ακόμα και όταν η ρομποτική διδάσκεται σαν ανεξάρτητο αντικείμενο, ο μαθητής αποκτά δεξιότητες όπως ικανότητα επίλυσης προβλημάτων, δημιουργικότητα, κριτική σκέψη, συνεργατικότητα κ.α. Επίσης, κατά τη διαδικασία σχεδιασμού και προγραμματισμού ρομπότ, οι μαθητές εμπεδώνουν και έννοιες από τις επιστήμες της μηχανικής, των μαθηματικών και των υπολογιστών (Druin και Hendler, 2000, Arlegui και άλλοι, 2008).

Και στις δύο περιπτώσεις, τα πρωτόκολλα, όπως αναφέρθηκε και στην προηγούμενη ενότητα, βασίζονται σε αυτά των συνθετικών εργασιών (project-based learning). Επομένως, στη πράξη, η μεθοδολογία που ακολουθείται, βασίζεται σε μεγάλο βαθμό στην μεθοδολογία της μάθησης με συνθετικές εργασίες.

Σύμφωνα με τους Han και Bhattacharya η μάθηση είναι βασισμένη σε συνθετικές εργασίες και μπορεί να οργανωθεί σε τρεις φάσεις (Han & Bhattacharya 2001):

1. Φάση σχεδιασμού (Planning phase): Ο μαθητευόμενος διαλέγει το θέμα, εντοπίζει τις πηγές και οργανώνει την συνεργατική δουλειά.
2. Φάση δημιουργίας (Creating or implementing phase): Σε αυτή τη φάση οι μαθητές της ομάδας συνδυάζουν τις ιδέες τους, κατασκευάζουν το project τους και το παρουσιάζουν στην τάξη.
3. Φάση επεξεργασίας (processing phase): Μετά την παρουσίαση στην τάξη των projects, οι ομάδες ανταλλάσσουν ιδέες μεταξύ τους και τους δίνεται έτσι η ευκαιρία για περαιτέρω σκέψη, εμβάθυνση στην διδακτική διαδικασία και βελτίωση των έργων τους. Πιο συγκεκριμένα, σε ότι έχει να κάνει με συνθετικές εργασίες βασισμένες στην ρομποτική, μπορεί να ακολουθηθεί η μεθοδολογία που προτείνεται από την TERECoP (Alimisis 2009) και βασίζεται στο μοντέλο που προτείνει ο Carbonaro (Carbonaro et al. 2004). Σύμφωνα με αυτή τη μεθοδολογία, η πορεία της εργασίας μπορεί να οργανωθεί στα παρακάτω στάδια:

Στάδιο εμπλοκής (engagement).

Στο στάδιο αυτό τίθεται στους μαθητές ένα πρόβλημα ανοιχτού τύπου και τους παρέχεται ένα αρχικό υλικό μελέτης με σκοπό να τους εμπλέξει στην διαδικασία επίλυσης του. Μέσα από το υλικό μελέτης (εφημερίδες, ηλεκτρονικές διευθύνσεις, βίντεο, πρώτες ύλες) οι μαθητές οδηγούνται σε διαδικασίες έκφρασης απόψεων, συζητήσεων, καταγιγισμού ιδεών, με επιθυμητό αποτέλεσμα στο τέλος του σταδίου αυτού να περιγράψουν ένα αρχικό project και τις ερωτήσεις που πρέπει να απαντηθούν για να επιλυθεί αυτό το πρόβλημα.

Στάδιο εξερεύνησης (exploration).

Οι μαθητές χωρίζονται σε ομάδες και εξοικειώνονται με το διαθέσιμο software και hardware για την επίλυση του προβλήματος. Κάνουν υποθέσεις και ελέγχουν το κατά πόσο μπορούν να γίνουν εφικτές στην πραγματικότητα. Τους παρέχονται εκπαιδευτικό υλικό, αντιπροσωπευτικά παραδείγματα, γενικές οδηγίες. Επιθυμητό αποτέλεσμα του σταδίου είναι μια κατασκευή με συγκεκριμένες λειτουργίες και ένα ημερολόγιο δραστηριοτήτων.

Στάδιο διερεύνησης (investigation):

Οι μαθητές επαναπροσδιορίζουν το πρόβλημα και τα ερωτήματα που προέκυψαν στο πρώτο στάδιο με βάση την εμπειρία που απέκτησαν μέσω του σταδίου εξερεύνησης. Από τον επαναπροσδιορισμό μπορεί να προκύψουν εναλλακτικές λύσεις για το πρόβλημα. Βασικό σημείο του σταδίου αυτού είναι ο καθορισμός των καθοδηγητικών ερωτήσεων (driving questions) σε συνεργασία των μαθητών με τον εκπαιδευτικό, οι οποίες θα καθοδηγήσουν τους μαθητές προς την επίλυση του προβλήματος.

Στάδιο δημιουργίας (creation):

Οι μαθητές μοιράζονται και συνδυάζουν τις ιδέες τους, κάθε ομάδα παρουσιάζει τη δουλειά της στην τάξη και στη συνέχεια δημιουργεί την τελική κατασκευή της.

Στάδιο αξιολόγησης (evaluation):

Οι ομάδες παρουσιάζουν τις εργασίες τους σε επίπεδο τάξης οι οποίες αξιολογούνται στο κατά πόσο απαντούν στα ερωτήματα που τέθηκαν στα προηγούμενα στάδια. Έτσι μπαίνουν στην διαδικασία να κάνουν απολογισμό των εργασιών τους, να βρουν τα δυνατά σημεία των κατασκευών κάθε ομάδας και να τα συνδυάσουν σε ένα τελικό προϊόν σε επίπεδο τάξης.

Τα παραπάνω στάδια δεν είναι απαραίτητο να ακολουθηθούν πάντα σε μια γραμμική σειρά. Κάποια στάδια μπορούν να επαναληφθούν, όπως επίσης κάποιο στάδιο μπορεί να περιέχεται μέσα σε ένα άλλο. Για παράδειγμα το στάδιο της δημιουργίας μπορεί να περιέχει ενέργειες διερεύνησης και το αντίθετο. Η πορεία επομένως των εργασιών είναι δυναμική και μπορεί να τροποποιείται με βάση το δυναμικό της τάξης.

Όπως γίνεται αντιληπτό, το αποτέλεσμα της εργασίας δεν είναι αυτοσκοπός. Ο κύριος στόχος είναι να εμπλακούν οι μαθητές σε μια διαδικασία σχεδιασμού δραστηριοτήτων που έχουν σημασία γι' αυτούς. Να μάθουν να σκέφτονται, να αξιοποιούν τα δεδομένα τους (θεωρητικές πηγές, υλικά) που τους δίνονται, να σχεδιάζουν λύσεις, να τις αξιολογούν και να αλλάζουν κατεύθυνση επανασχεδιάζοντας τις αν χρειαστεί, καθώς και να είναι σε θέση να συζητήσουν και να τεκμηριώσουν τις λύσεις που προτείνουν (Maeda 2000). Επομένως, αφού σκοπός είναι να γίνουν οι μαθητές «σχεδιαστές», ο εκπαιδευτικός πρέπει να οργανώσει την διαδικασία της μάθησης προς αυτόν τον σκοπό. Να σχεδιάσει δηλαδή πράγματα (φύλλα εργασίας, πηγές) που θα επιτρέψουν και θα ενθαρρύνουν τους μαθητές να

γίνουν και οι ίδιοι σχεδιαστές για να σχεδιάσουν και να επανασχεδιάσουν την κατασκευή τους (Resnick & Silverman 2005).

Η Ιστορία της Logo

Η Logo είναι γλώσσα προγραμματισμού η οποία αναπτύχθηκε στις αρχές του 1970 από την ομάδα του Seymour Papert (με τη συμβολή της Cynthia Solomon και της Wally Feuerzeig) ως μαθησιακό εργαλείο. Το όνομά της προέρχεται από την ελληνική λέξη ΛΟΓΟΣ. Η Logo αποτελεί διάλεκτο της Lisp, της γλώσσας της τεχνητής νοημοσύνης, και έχει αξιοποιηθεί στην εκπαίδευση περισσότερο από οποιαδήποτε άλλη γλώσσα προγραμματισμού. Βάση της αποτελεί η φιλοσοφία του Κονστρουκτιβισμού όπως αναπτύχθηκε από τον Piaget (1965). Επιπρόσθετα, η Logo συνδυάζει τις προσεγγίσεις του Piaget και του Vygotsky. Παρόλο που συχνά αμφισβητείται η εκπαιδευτική της αποτελεσματικότητα, θεωρείται ιδανικό εργαλείο για να μαθαίνεις κάνοντας (learning by doing) και αναμφίβολα αποτελεί σημαντικό εργαλείο στα χέρια του εκπαιδευτικού για την ανάπτυξη δεξιοτήτων εξερεύνησης, δημιουργικότητας, επίλυσης προβλημάτων, λογικής-αλγοριθμικής σκέψης.

Όλες οι γλώσσες προγραμματισμού αποτελούν εργαλεία για την ανάπτυξη μοντέλων. Η Logo είναι ιδιαίτερα κατάλληλη για την ανάπτυξη μοντέλων από παιδιά. Απ' όλες τις γλώσσες προγραμματισμού η Logo σχεδιάστηκε για να δημιουργεί μοντέλα με πολύ εύκολο τρόπο. Με τη Logo αναπτύσσουμε μοντέλα οραματιζόμενοι ένα «όλον», σπάζοντας το σε τμήματα εύκολα διαχειρίσιμα τα οποία μπορούν να διδαχθούν στον υπολογιστή. Χρησιμοποιούμε τη γλώσσα που γνωρίζει ο υπολογιστής για να τον διδάξουμε νέες λέξεις. Συχνά διδάσκουμε τη χελώνα νέες λέξεις γράφοντας διαδικασίες, οι οποίες είναι μια σειρά εντολών για να κάνει κάτι. Πειραματιζόμαστε με τις εντολές και αξιολογούμε το πρόγραμμά μας για να δούμε αν τα έργα υλοποιήθηκαν όπως περιμέναμε. Ανταποκρινόμαστε στις παρανοήσεις του υπολογιστή καθώς εκσφαλματώνουμε τις εντολές και τις διαδικασίες και μερικές φορές αναδομούμε την προσέγγισή μας.

Μια νέα εταιρεία, η Logo Computer Systems, Inc (LCSI) ιδρύθηκε το 1980. Πολλοί από τους ερευνητές, καθηγητές, προγραμματιστές, και συγγραφείς που συμμετείχαν σε αυτό το εγχείρημα έχουν παίξει σημαντικό ρόλο στην μετέπειτα εξέλιξη της Logo. Ο Brian Silverman είναι Διευθυντής Ερευνών και καθοδήγησε την ανάπτυξη όλων

των προϊόντων LCSI του. Ο Cynthia Solomon, ο οποίος ήταν στην ομάδα που δημιούργησε το αρχικό λογότυπο το 1967, ήταν επικεφαλής στο πρώτο γραφείο LCSI στη Βοστώνη και στη συνέχεια σκηνοθέτησε την Atari Cambridge Research Center. Ο Sharnee Chait έχει επιβλέψει την τεκμηρίωση και την ανάπτυξη του προϊόντος σε LCSI για τα τελευταία 16 χρόνια.

Η LCSI ανέπτυξε την Apple Logo, ακολουθούμενη από εκδόσεις για μια σειρά από άλλους υπολογιστές. Με την εμπορική χρήση της Logo, η διαθεσιμότητα εξαπλώθηκε γρήγορα.

Ένα άλλο σημαντικό γεγονός συνέβη το 1980 -η δημοσίευση του 'Mindstorms' του Seymour Papert. Χιλιάδες εκπαιδευτικοί σε όλο τον κόσμο ενθουσιάστηκαν από το πνευματικό και δημιουργικό δυναμικό της Logo. Ο ενθουσιασμός τους τροφοδότησε την άνθηση της Logo στις αρχές της δεκαετίας του 1980.

Νέες εκδόσεις της Logo υλοποιήθηκαν σε περισσότερες από δώδεκα γλώσσες που ομιλούνται, σε μια ποικιλία μηχανημάτων, με πολλά γραφικά με στυλ video game και δυνατότητες ήχου. Η Logo για υπολογιστές MSX ήταν δημοφιλής στην Ευρώπη, τη Νότια Αμερική και την Ιαπωνία. Οι Atari Logo και Commodore Logo ήταν δημοφιλείς στη Βόρειο Αμερική.

Μέχρι τα μέσα της δεκαετίας του 1980, οι υπολογιστές με δυνατότητες βίντεο παιχνιδιού είχαν βγει εκτός αγοράς και το ίδιο συνέβη στις εκδόσεις της Logo που έφεραν. Οι MSDOS μηχανές κυριάρχησαν όλο και περισσότερο στον κόσμο της εκπαίδευσης μέσω υπολογιστών, εκτός από τις Ηνωμένες Πολιτείες, όπου η Apple ήταν η αγαπημένη των σχολείων. Οι προγραμματιστές της Logo εστίασαν σε αυτά τα μηχανήματα. Παρά το γεγονός ότι νέες εφαρμογές προσέθεσαν χαρακτηριστικά και εκμεταλλεύτηκαν την αυξημένη ταχύτητα και τη μνήμη των καινούργιων υπολογιστών, οι πιο δημοφιλείς εκδόσεις της Logo που ήταν σε χρήση το 1985, ήταν παρόμοιες με αυτές του 1980.

Αυτό το διάστημα υπήρχε και κάποιο ενδιαφέρον για τη χρήση της Logo ως «σοβαρής» γλώσσας προγραμματισμού, ειδικά για το νέο υπολογιστή Macintosh. Η MacLogo από το LCSI προσέθεσε νέα λειτουργικότητα στο περιβάλλον Logo. Η Coral Software, ανέπτυξε μια προσανατολισμένη στο αντικείμενο έκδοση της Logo που ονομάζεται Logo αντικειμένου Object Logo. Περιλάμβανε έναν δημιουργό

κώδικα αντικειμένου (compiler) που επιτρέπει στα προγράμματα να «τρέχουν» με μεγαλύτερη ταχύτητα, και έτσι, stand-alone εφαρμογές θα μπορούσαν να δημιουργηθούν. Αλλά η Logo δεν έγινε δημοφιλής μεταξύ των προγραμματιστών εφαρμογών.

Πλεονεκτήματα της Logo

Για τους μαθητές, κυρίως της υποχρεωτικής εκπαίδευσης, που έρχονται για πρώτη φορά σε επαφή με ένα προγραμματιστικό περιβάλλον, ένα περιβάλλον στο οποίο αναπτύσσουν και καταγράφουν τη σκέψη τους με αυστηρό τρόπο και βλέπουν άμεσα τα αποτελέσματα των ενεργειών τους, η Logo συνεισφέρει στην ανάπτυξη γνωστικών δεξιοτήτων και μπορεί να εξυπηρετήσει τα εξής:

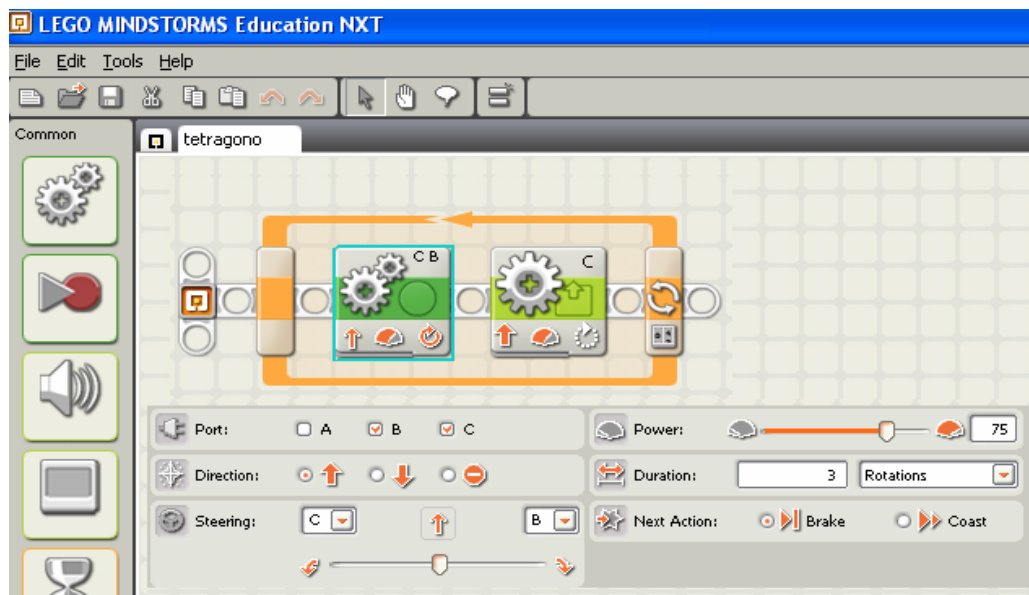
- Τη δυνατότητα για συνεχή, εύκολη και παρατεταμένη ενασχόληση και πειραματισμό με συγκεκριμένες πλευρές ενός προβλήματος και τη διαμόρφωση στρατηγικών, για την αντιμετώπιση του.
- Την «αποποινικοποίηση» και αξιοποίηση της έννοιας του λάθους. Προσπαθώντας οι μαθητές να επιβληθούν στο προγραμματιστικό περιβάλλον, ψάχνοντας τρόπους να κάνουν το πρόγραμμα να δουλέψει, γεύονται την εμπειρία της επιτυχίας και αποτυχίας και τις αντιμετωπίζουν σαν πληροφορία και όχι σαν αμοιβή ή ποινή.
- Παρέχεται η δυνατότητα στους μαθητές, να φτιάξουν κάτι (μια καινούργια εντολή, κανόνα, διαδικασία) και έπειτα να το χρησιμοποιήσουν για την κατασκευή μιας ακόμα πιο σύνθετης και πιο γενικευμένης διαδικασίας.

1.1.5 Συστήματα LEGO και γλώσσες εκπαιδευτικής ρομποτικής

Η ιδέα των προγραμματιζόμενων ρομποτικών κατασκευών ξεκίνησε με τη χελώνα της Logo από τα τέλη της δεκαετίας του '60. Οι χελώνες αυτές συνδέονταν με τον υπολογιστή με ένα καλώδιο και μπορούσαν να κινούνται στο πάτωμα ανάλογα με τις εντολές που λάμβαναν. Σήμερα ποικιλία δομικών στοιχείων και ηλεκτρονικών εξαρτημάτων χρησιμοποιούνται για τη σύνθεση ρομποτικών κατασκευών, ενώ η ανάγκη της αυτονομίας των κατασκευών από τον υπολογιστή οδήγησε στη χρήση προγραμματιζόμενων κύβων με ενσωματωμένους μικροεπεξεργαστές. Το σύστημα LEGO Mindstorms NXT (<http://www.legomindstorms.com>) προσφέρει ένα ευφύες

ελεγχόμενο από υπολογιστή «τούβλο» LEGO, που αποτελεί τον εγκέφαλο του εκπαιδευτικού ρομπότ του LEGO MINDSTORMS, διαδραστικούς κινητήρες, ηχητικούς, υπερηχητικούς και άλλους αισθητήρες, καθώς και μια μεγάλη συλλογή υλικών κατασκευής («τουβλάκια», γρανάζια, τροχαλίες, άξονες κλπ.) που δομούνται γύρω ή πάνω στον μικροεπεξεργαστή. Προσφέρει επίσης ένα απλό γραφικό περιβάλλον προγραμματισμού

(http://mindstorms.lego.com/Overview/NXT_Software.aspx), που δίνει τη δυνατότητα δημιουργίας προγραμματιζόμενων «συμπεριφορών» για τις μηχανικές κατασκευές. Οι προγραμματιζόμενες «συμπεριφορές» μεταβιβάζονται από τον Η.Υ. στη μηχανική κατασκευή μέσω σύνδεσης USB ή Bluetooth μεταξύ του Η.Υ. και του μικροεπεξεργαστή των μηχανικών κατασκευών. Το εκπαιδευτικό λογισμικό LEGO MINDSTORMS Education NXT βασίζεται στη χρήση εικονιδίων και είναι μια εκπαιδευτική έκδοση του επαγγελματικού λογισμικού LabView του National Instruments, λογισμικό που χρησιμοποιούν παγκοσμίως επιστήμονες και μηχανικοί, προκειμένου να σχεδιάσουν, να ελέγξουν και να δοκιμάσουν προϊόντα και συστήματα. Το λογισμικό έχει μια διαισθητική διεπαφή “σύρε και άφησε” (drag and drop) και ένα γραφικό προγραμματιστικό περιβάλλον, το οποίο καθιστά την εφαρμογή προσιτή για έναν αρχάριο, αλλά και εξίσου δυναμική για έναν εξειδικευμένο χρήστη. Οι παλέτες προγραμματισμού προσφέρουν όλα τα blocks προγραμματισμού που απαιτούνται για να δημιουργηθούν τα προγράμματα. Κάθε block προγραμματισμού περιλαμβάνει τις οδηγίες που το NXT μπορεί να ερμηνεύσει. Ένα πρόγραμμα δημιουργείται με συνδυασμό διαφορετικών blocks. Τα διαθέσιμα εικονίδια-blocks περιλαμβάνουν μεταξύ άλλων block κίνησης (κάνουν τα ρομπότ να κινούνται), block αναμονής (κάνουν το ρομπότ να περιμένει για την ενεργοποίηση των αισθητήρων του ή για τη λήξη ενός οριζόμενου χρονικού διαστήματος), block επανάληψης (Loop) (το ρομπότ επαναλαμβάνει την ίδια συμπεριφορά όσες φορές ορίσουμε ή μέχρι να ενεργοποιηθεί κάποιος αισθητήρας), block επιλογής (Switch block) (επιτρέπουν στο ρομπότ να παίρνει τις δικές του αποφάσεις). Η πλήρης παλέτα εικονιδίων περιλαμβάνει blocks δράσης που 4ο 278 Πανελλήνιο Συνέδριο Διδακτική της Πληροφορικής επιτρέπουν τον έλεγχο διάφορων εξωτερικών συσκευών (διαδραστικού κινητήρα, ήχων, λαμπτήρων κ.ά.)



Εικόνα 1.10: ενδεικτικό πρόγραμμα με επανάληψη (loop) για την κίνηση του ρομπότ σε τετράγωνο

Τα blocks ροής επιτρέπουν τη δημιουργία σύνθετων συμπεριφορών. Περιλαμβάνουν τον έλεγχο για την επανάληψη, την αναμονή και τις συνθήκες μεταβλητών, για τη διακοπή συμπεριφοράς ή τον καθορισμό μιας λογικής σειράς σε ένα πρόγραμμα και τη λήψη αποφάσεων για τον προγραμματισμό αντιδράσεων σε καθορισμένες τιμές των αισθητήρων. Τέλος το πρόγραμμα προσφέρει «Τα δικά μου blocks» (My Blocks) με τα οποία ο χρήστης μπορεί να αποθηκεύσει ένα δικό του πρόγραμμα ως ένα μοναδικό block, που μπορεί να το ξαναχρησιμοποιήσει σε άλλο πρόγραμμα. Κάθε block προγραμματισμού έχει έναν πίνακα διαμόρφωσης, μέσω του οποίου μπορούν να γίνονται ρυθμίσεις και επιλέγονται παράμετροι που επιτρέπουν τον έλεγχο της συμπεριφοράς του block (π.χ. αλλάζοντας την ισχύ στο block κίνησης μπορούμε να κάνουμε το ρομπότ να κινηθεί ταχύτερα).

Λογισμικό προγραμματισμού

Το LEGO MINDSTORMS NXT μπορεί να προγραμματιστεί μέσω μιας σειράς γλωσσών προγραμματισμού, οι κυριότερες των οποίων είναι:

- RCX Code (περιέχεται στην παλιότερη έκδοση του Mindstorms)

- LEGO MINDSTORM NXT-G (περιέχεται στην καινούρια έκδοση του Mindstorms)
- ROBOLAB (βασίζεται στο LabView και αναπτύχθηκε στο Tufts University)

Δημοφιλής Γλώσσες τρίτων κατασκευαστών:

- C and C++ under BrickOS (formerly LegOS)
- Java under leJOS or TinyVM
- NQC ("Not Quite C")
- pbFORTH (επεκτάσεις της Forth γλώσσας προγραμματισμού)
- Visual Basic (μέσω του COM+ interface παρεχόμενο με το CD)
- RobotC (νέα γλώσσα συμβατή με την έκδοση NXT)
- Matlab Toolbox

Γιατί να διδάξω με Lego MINDSTORMS στο σχολείο

Η εκπαιδευτική δυναμική των προγραμματιζόμενων ρομποτικών κατασκευών LEGO MINDSTORMS συνίσταται στη δυνατότητα που προσφέρει στους μαθητές, να συνθέσουν μια μηχανική οντότητα (π.χ. ένα μοντέλο αυτοκινήτου) και να την κατευθύνουν με τη βοήθεια ενός απλού και εύχρηστου προγραμματιστικού περιβάλλοντος. Το πακέτο LEGO MINDSTORMS, αν αξιοποιηθεί κατάλληλα, μπορεί να υποστηρίξει τη δημιουργία ενός περιβάλλοντος εποικοδομητικής μάθησης (constructive learning) που θα παρέχει αυθεντικές εκπαιδευτικές δραστηριότητες ενταγμένες σε διαδικασίες επίλυσης ανοιχτών προβλημάτων από τον πραγματικό κόσμο, θα ενθαρρύνει την έκφραση και την προσωπική εμπλοκή στη μαθησιακή διαδικασία και θα υποστηρίξει την κοινωνική αλληλεπίδραση.

Με τα LEGO MINDSTORMS γίνονται πράξη οι ιδέες του S. Papert για «μαστόρεμα της γνώσης»: τα παιδιά θα μπορέσουν να οικοδομήσουν πιο αποτελεσματικά τη γνώση όταν εμπλέκονται ενεργά στη σχεδίαση και κατασκευή (χειρωνακτική και ψηφιακή) πραγματικών αντικειμένων που έχουν νόημα για τους ίδιους είτε αυτά είναι κάστρα από άμμο, είτε κατασκευές LEGO και προγράμματα υπολογιστών.

Η σχεδίαση δραστηριοτήτων με τις ρομποτικές κατασκευές LEGO MINDSTORMS συνδέεται με την εκπλήρωση ενός έργου με στόχο την επίλυση ενός προβλήματος. Σε ένα τέτοιο μαθησιακό περιβάλλον, η μάθηση καθοδηγείται από το προς επίλυση πρόβλημα. Προκειμένου να εμπλακούν οι μαθητές σε δραστηριότητες σχεδίασης και κατασκευής πραγματικών αντικειμένων, δηλαδή ρομποτικών κατασκευών, θα πρέπει να επινοήσουμε δραστηριότητες που θα προτρέπουν τους μαθητές να κατασκευάσουν αλλά συγχρόνως να τους ενθαρρύνουμε και να τους υποστηρίξουμε κατάλληλα ώστε να πειραματιστούν με τις κατασκευές τους. Οι δραστηριότητες αυτές είναι συνήθως διαθεματικές και μπορούν να ενταχθούν στα σχολικά μαθήματα της τεχνολογίας, των

φυσικών επιστημών και της πληροφορικής τόσο στην πρωτοβάθμια όσο και στη δευτεροβάθμια εκπαίδευση.

Πλεονεκτήματα των LEGO MINDSTORMS

- Πολλοί μαθητές εξοικειώνονται με τα γνωστά τουβλάκια της Lego από μικρές ηλικίες.
- Οι μαθητές τα αντιμετωπίζουν περισσότερο ως παιχνίδι, παρά ως εργαλεία μάθησης καθώς η πλειοψηφία τους έχει «παίξει» με αυτά (τουβλάκια).
- Δυνατότητα υλοποίησης ρεαλιστικών σεναρίων (πχ. ένα όχημα που αντιδρά ανάλογα στους φωτεινούς σηματοδότες)
- Το περιβάλλον των LM, είναι ένα περιβάλλον πλούσιο σε υλικά, το οποίο διέπεται από τις Θεωρίες Οικοδόμησης της Γνώσης, σύμφωνα με τις οποίες οι μαθητές δε μαθαίνουν απλώς γεγονότα, εξισώσεις και τεχνικές αλλά μαθαίνουν να σκέπτονται με κριτικό και συστηματικό τρόπο για να λύσουν ένα πρόβλημα.
- Η κατασκευή και ο προγραμματισμός φυσικών μοντέλων τους βοηθά να συνδέσουν ιδέες και πληροφορίες που διδάσκονται θεωρητικά με το φυσικό κόσμο.
- Τα φυσικά μοντέλα είναι ελκυστικά καθώς προσφέρουν άμεση ανατροφοδότηση (feedback) στα παιδιά σχετικά με την αποτελεσματικότητα των προγραμμάτων τους - δοκιμάζουν τις προγραμματιστικές λύσεις που προτείνουν και βλέπουν άμεσα το αποτέλεσμα, παρατηρώντας την προγραμματισμένη συμπεριφορά της ρομποτικής τους κατασκευής.
- Το πακέτο LEGO Mindstorms περιλαμβάνει μία ποικιλία δομικών υλικών που δίνουν τη δυνατότητα ανάπτυξης πολλών διαφορετικών κατασκευών και όχι μιας κατασκευής, με δυνατότητα προσθαφαίρεσης αισθητήρων.
- Ένα ρομπότ NXT προσαρμόζεται εύκολα και μπορεί να επαναχρησιμοποιηθεί στη διδασκαλία και να καλύψει ένα μεγάλο εύρος θεμάτων του αναλυτικού προγράμματος σπουδών.
- Παρέχει αντοχή και η αξιοπιστία στα υλικά.

- Προσφέρει ένα απλό γραφικό περιβάλλον προγραμματισμού (LEGO MINDSTORMS Edu NXT), που δίνει τη δυνατότητα δημιουργίας προγραμματιζόμενων «συμπεριφορών» για τις μηχανικές κατασκευές. Το λογισμικό έχει μια διαισθητική διεπαφή “σύρε και άφησε” (drag and drop) και ένα γραφικό προγραμματιστικό περιβάλλον που βασίζεται στη χρήση εικονιδίων, γεγονός που καθιστά την εφαρμογή προσιτή για έναν αρχάριο, αλλά και εξίσου δυναμική για έναν εξειδικευμένο χρήστη και επομένως είναι κατάλληλο για όλες τις βαθμίδες εκπαίδευσης.
- Κάθε εικονίδιο - εντολής αντιστοιχεί σε μία φυσική οντότητα ή λειτουργία της ή ακόμα και σε ένα σύνολο ενεργειών. Η ανάπτυξη προγραμμάτων με τη χρήση εικονιδίων - εντολών απαλλάσσει τους μαθητές από την εκμάθηση μιας γλώσσας προγραμματισμού και την απομνημόνευση των συντακτικών της κανόνων. Ο προγραμματιστής δε χρειάζεται να θυμάται καμιά εντολή. Πρέπει μόνο να θέτει σε ορθή σειρά τις εικόνες που θα καθορίσουν τον τρόπο κίνησης της κατασκευής του και τον τρόπο αντίδρασης με βάση τα σήματα που λαμβάνουν οι αισθητήρες που ενδεχομένως έχει προσθέσει σ’ αυτή.
- Ο οπτικός προγραμματισμός καθώς και η αναπαραστατικότητα των εικονιδίων - εντολών διευκολύνουν τους μαθητές να κατανοούν εύκολα και γρήγορα τα περισσότερα από τα εικονίδια εντολών.
- Οι μαθητές πολύ γρήγορα δημιουργούν «κώδικα που δουλεύει» δηλαδή δεν χρειάζεται να πάρουν πολλές πληροφορίες για το περιβάλλον ώστε να δημιουργήσουν απλά προγράμματα.
- Χρησιμοποιείται για την εισαγωγή αρχάριων μαθητών στην εκμάθηση του προγραμματισμού.
- Το λογισμικό έχει ενσωματωμένο έναν εύχρηστο ρομποτικό οδηγό ο οποίος παρουσιάζει βήμα - βήμα την κατασκευή και τον προγραμματισμό ρομπότ με διάφορα παραδείγματα, μέσα από εικόνες και videos.

Μειονεκτήματα των LEGO MINDSTORMS

- **Υψηλό κόστος**

Ο εξοπλισμός των εκπαιδευτικών μονάδων με τα LM συνεπάγεται σε αρκετά υψηλό κόστος. Απαιτείται η αγορά ενός επαρκή αριθμού εκπαιδευτικών σετ LM καθώς και κατάλληλες υποδομές για την υποστήριξη εργαστηρίου ρομποτικής με πάγκους εργασίας, Η/Υ, σύνδεση στο Internet και αρκετό ελεύθερο χώρο.

- **Φυσικοί περιορισμοί του υλικού**

Οι κινήσεις ενός ρομποτικού μοντέλου δεν είναι πάντα ακριβείς. Αυτό μπορεί να οφείλεται στο περιβάλλον μέσα στο οποίο λειτουργεί το ρομπότ πχ. ένα ρομπότ που παρεκκλίνει από την προγραμματισμένη πορεία του εξαιτίας κάποιου εμποδίου, λόγω τριβής κ.α. Επίσης, η επαναφορτιζόμενη μπαταρία θα πρέπει να φορτίζεται συχνά κ.λπ.

- **Χρονικοί περιορισμοί**

Οι δραστηριότητες εκπαιδευτικής ρομποτικής απαιτούν περισσότερο διδακτικό χρόνο από αυτόν που συνήθως προβλέπουν τα σχολικά ωρολόγια προγράμματα. Η διδασκαλία με τη χρήση LM είναι χρονοβόρα τόσο για τη διεξαγωγή της διδασκαλίας όσο και για την οργάνωσή της. Ο εκπαιδευτικός σε πρώτη φάση θα πρέπει να αφιερώσει κάποιες ώρες διδασκαλίας για την εξοικείωση των παιδιών με το υλικό και το λογισμικό LM. Επίσης, θα πρέπει να οργανώσει κατάλληλα τη διδασκαλία του με φύλλα εργασίας και άλλο κατάλληλο υποστηρικτικό υλικό καθώς και να προετοιμάσει κατάλληλα το χώρο του εργαστηρίου. Το μεγάλο πλήθος μικρών εξαρτημάτων αυξάνει το χρόνο οργάνωσης και διατήρησης. Τέλος, ο μαθητής πρέπει να έχει στη διάθεσή του το χρόνο που χρειάζεται για να δουλέψει σύμφωνα με τους δικούς του ρυθμούς μάθησης ώστε να επιτευχθεί η μάθησή του.

Για τη διδασκαλία με LM απαιτείται κατάλληλη οργάνωση

- Δημιουργία ομάδων
- Μαθησιακό συμβόλαιο (συναποφασίζουν οι εκπαιδευόμενοι και ο εκπαιδευτικός σχετικά με τους στόχους, τις ομάδες κλπ)
- Οργάνωση των ομάδων
- Ανάθεση ρόλων στις ομάδες πχ. κατασκευαστής, προγραμματιστής
- Υποστηρικτικό υλικό (Φύλλα εργασίας με οδηγίες κλπ)
- Στήριξη των εκπαιδευόμενων (ενθάρρυνση, καθοδήγηση και στήριξη της προσπάθειας των εκπαιδευόμενων)
- Ο εκπαιδευτικός προωθεί το συναγωνισμό (κίνητρο για την ενεργό συμμετοχή των παιδιών) αλλά όχι τον ανταγωνισμό
- Η δημιουργία ενός μαθησιακού περιβάλλοντος εποικοδομητικής μάθησης (constructive learning) που θα παρέχει αυθεντικές εκπαιδευτικές δραστηριότητες ενταγμένες σε διαδικασίες επίλυσης ανοιχτών προβλημάτων από τον πραγματικό κόσμο

Αποτελέσματα από τη χρήση του εκπαιδευτικού προγράμματος

Από έρευνες που έχουν γίνει σε σχολεία όπου χρησιμοποιούσαν τη ρομποτική πλατφόρμα NXT παρατηρήθηκαν τα εξής:

- **Επαφή των παιδιών με τις νέες τεχνολογίες.** Παρουσιάστηκε αύξηση στο ενδιαφέρον των παιδιών να μαθαίνουν και να ανακαλύπτουν πώς λειτουργούν οι μηχανές γύρω τους.
- **Συνεργασία.** Η συνεργασία που αποτελούσε αναπόσπαστο μέρος των δραστηριοτήτων του εκπαιδευτικού υλικού, συνέλαβε στην αύξηση των κινήτρων των παιδιών και οδήγησε στη βελτίωση των επιδόσεών του.
- **Δημιουργικότητα.** Οι άπειρες επιλογές που δίνει το υλικό για την κατασκευή μοντέλων, καθώς και η επίγνωση των παιδιών για την ύπαρξη πέραν από μίας σωστής απάντησης, το καθιστά ιδιαίτερα προκλητικό για την φαντασία τους.

- **Κίνητρα.** Οι εκπαιδευτικοί παρατήρησαν ότι τα παιδιά παρακολουθούσαν πιο ευχάριστα το μάθημα και ανυπομονούσαν να εργαστούν με το υλικό.
- **Ίσες ευκαιρίες.** Το συνηθισμένο πρόβλημα σε μια τάξη είναι η ανομοιογένεια των παιδιών. Το εκπαιδευτικό αυτό πρόγραμμα, ανταποκρίνεται στις ικανότητες τόσο των παιδιών που έχουν ανεπτυγμένες χειριστικές δεξιότητες τα οποία ασχολούνται με την κατασκευή του μοντέλου-ρομπότ, όσο και των παιδιών που έχουν ανεπτυγμένες δεξιότητες ανάλυσης που ασχολούνται με τον προγραμματισμό του μοντέλου.
- **Ερευνητικό ενδιαφέρον.** Η δράση των παιδιών ως επιστημόνων-εφευρετών, οδήγησε στην ανάπτυξη της συνήθειας για υποβολή ερωτήσεων, παρατήρησης, πειραματισμού και εύρεσης πρακτικών λύσεων βάση τεχνολογικών αρχών.

1.1.6 Διαγωνισμός της εκπαιδευτικής ρομποτικής

Ιστορία

Όλα ξεκίνησαν το 1999 ...

Εκείνη την εποχή, ο κ. TaeSoo Park, ο πρόεδρος της κορεατικής εταιρείας "Learning Tool, Inc", επικοινωνήσε με τον κ. Lars Vahl, διευθυντής της LEGO Education Asia Pacific, γεγονός που υποδηλώνει ότι θα μπορούσαν να δημιουργήσουν ένα διαγωνισμό ρομποτικής με τη συμμετοχή φοιτητών από χώρες της Ασίας.

Το έτος 2000, τέσσερις χώρες (Κορέα, Κίνα, Ιαπωνία και Σιγκαπούρη) εντάχθηκαν στο πιλοτικό πρόγραμμα, και ο πρώτος διεθνής διαγωνισμός - με τίτλο Διεθνές Ρομποτική Ολυμπιάδα - έλαβε χώρα στο Ολυμπιακό Πάρκο της Σεούλ, στην Κορέα.



Εικόνα 1.11: world robot Olympiad

Τι είναι το WRO;

Το World Robot Olympiad είναι ένα παγκόσμιο ανταγωνισμό LEGO Robotics. Είναι ένα γεγονός που φέρνει τους νέους ανθρώπους από όλο τον κόσμο μαζί. Η εκδήλωση αυτή συνδυάζει τον κόσμο της επιστήμης, της μηχανικής, της τεχνολογίας και της εκπαίδευσης. Ομάδες νέων έχουν την ευκαιρία να αναπτύξουν τη δημιουργικότητα και τις δεξιότητες επίλυσης προβλημάτων τους μέσα από διάφορες προκλήσεις ρομποτικής. Η Ολυμπιάδα ξεκίνησε το 2004 και παρουσιάστηκε για πρώτη φορά στη Σιγκαπούρη. Επί του παρόντος 35 χώρες με περίπου 16.000 ομάδες που συμμετέχουν σε αυτή την εκδήλωση. Κάθε συμμετέχουσα χώρα έχει το δικό της ανταγωνισμό και οι καλύτερες ομάδες καλούνται να παραστούν και να ανταγωνιστούν στο World Robot Olympiad.

Το World Robot Olympiad ιδρύθηκε το 2004

Το 2003 οι τέσσερις ιδρυτικές χώρες και η LEGO Education ίδρυσαν μια διεθνή WRO επιτροπή, η οποία αποφάσισε να δημιουργήσει μια νέα και μόνιμη ρομποτική οργάνωση, που βασίζεται στην ιδέα ότι οι μαθητές από όλο τον κόσμο θα μπορούν να έχουν την ευκαιρία να συναντηθούν με άλλους μαθητές για να εκπληρώσουν τη νέα αποστολή, που έγραφε:

"Για να φέρει σε επαφή νέους ανθρώπους από όλο τον κόσμο για την ανάπτυξη της δημιουργικότητας, το σχεδιασμό και δεξιοτήτων επίλυσης προβλημάτων τους, μέσα από την αμφισβήτηση και των εκπαιδευτικών διαγωνισμών ρομποτικής και δραστηριοτήτων".



Εικόνα 1.12: Μέλη της 1ης επιτροπής WOR από πάνω αριστερά προς κάτω δεξιά :Ο κ. Tae Soo Park και Dr.Youm (Κορέα), ο Dr. Kanai (Ιαπωνία), ο κ. Lars Vahl (Δανία), ο κ. Clarence Sirisena (Σιγκαπούρη), κ. Ευγένιος Ζανγκ (Κίνα), ο κ. Kobayashi (Ιαπωνία), Dr. Marcello Ang (Sgp.)

Η WRO επιτροπή αποφάσισε το νέο όνομα WORLD ROBOT OLYMPIAD και αναπτύχθηκαν νέα λογότυπα WRO.

Τέλος, το WRO καταστατικό και ένα σύνολο γενικών κανόνων εργάστηκαν για να εξασφαλίσουν ένα υγιές και ασφαλές μέλλον για το WRO. Μία από τις σημαντικότερες αποφάσεις, που εμφανίζεται στο καταστατικό, ήταν ότι η διεθνής εκδήλωση WRO θα μπορούσε να φιλοξενηθεί από μια νέα χώρα κάθε χρόνο και ένας πρόεδρος της WRO θα πρέπει να εκλέγεται από την επιτροπή WRO μία φορά το χρόνο.

Η ενημέρωση του WRO αυξήθηκε γρήγορα και ήδη από το 2004 , 12 χώρες συμμετείχαν στη διεθνή εκδήλωση WRO όπου φιλοξενούνταν στη Σιγκαπούρη. Το 2011 συμμετείχαν σχεδόν 15.000 παιδιά από 34 χώρες.

Προστασία του εμπορικού σήματος WRO το 2010

Για την προστασία του εμπορικού σήματος WRO η WRO επιτροπή αποφάσισε να εγγράφετε το WRO κάτω από το όνομα WORLD ROBOT OLYMPIAD

ASSOCIATION Ltd. στις 29 Μαΐου του 2010, το WRO καταχωρήθηκε επίσημα ως νομική οντότητα με τη Λογιστική και Εταιρική Ρυθμιστική Αρχή (ACRA) στη Σιγκαπούρη. Κατά συνέπεια , η οργάνωση και η διαχείριση του WRO άλλαξε . Η ανώτατη αρχή του WRO είναι τώρα η WRO Advisory Council (AC) με 12 μέλη . Το Δ.Σ. εκλέγει τον Πρόεδρο του WRO, το Γενικό Γραμματέα του WRO καθώς και έξι μέλη του WRO Διοικητικού Συμβουλίου (BOT) . Το bot εκλέγει τον WRO Program Manager για να εφαρμόσουν αποφάσεις που λαμβάνονται από το bot και AC .

Σχετικά με WRO

Το World Robot Olympiad είναι μια εκδήλωση για την επιστήμη , την τεχνολογία και την εκπαίδευση που φέρνει σε επαφή τους νέους ανθρώπους από όλο τον κόσμο για να αναπτύξουν τη δημιουργικότητά τους καθώς και τις δεξιότητες επίλυσης προβλημάτων μέσα από την πρόκληση και τους διαγωνισμούς της εκπαιδευτικής ρομποτικής.

Στο διαγωνισμό αυτό τα σχολεία καλούνται να συμμετάσχουν με μια ομάδα των τριών μαθητών. Οι συμμετέχουσες ομάδες πρέπει να δημιουργήσουν, να σχεδιάσουν και να οικοδομήσουν μοντέλα ρομπότ που φαίνονται ή να συμπεριφέρονται σαν άνθρωποι.

Στόχοι

- Δίνει στους μαθητές μέχρι την ηλικία των 19 ετών την ευκαιρία να διευρύνουν τους ορίζοντές τους μέσα από την εξερεύνηση των ρομπότ και των ρομποτικών συστημάτων στο σχολείο.
- Εισάγει την έννοια της σύγχρονης επιστήμης σε εκπαιδευτικές δραστηριότητες του σχολείου στον τομέα της επιστήμης και της τεχνολογίας .
- Δίνει μια ευκαιρία για την προώθηση της δημιουργικής σκέψης , βελτίωση της επικοινωνίας και των δεξιοτήτων συνεργασίας ενισχύοντας την ικανότητα να αποκτήσουν νέες γνώσεις, σημαντικό για την προοδευτική εκπαίδευση.
- Φέρει σε επαφή νέους ανθρώπους από όλο τον κόσμο για την ανάπτυξη της δημιουργικότητας και δεξιοτήτων επίλυσης προβλημάτων μέσα από την πρόκληση και το διαγωνισμό εκπαιδευτικής ρομποτικής .

- Για να διευρύνει την άποψη της νεολαίας στην εφαρμογή της επιστήμης και της τεχνολογίας , τη βελτίωση της μαθησιακής αποτελεσματικότητά τους και την ενθάρρυνση των νέων να είναι οι μελλοντικοί επιστήμονες, μηχανικοί και εφευρέτες.

Διαγωνισμοί στο WRO

Το World Robot Olympiad διοργανώνει διαγωνισμούς ρομποτικής σε τρεις διαφορετικές κατηγορίες διαγωνισμών

Η τακτική κατηγορία ήταν ο πρώτος διαγωνισμός που εισήγαγε το WRO. Είναι μια πρόκληση που βασίζεται στο διαγωνισμό στον οποίο οι ομάδες κατασκευάζουν ρομπότ που έχουν σχεδιαστεί για την επίλυση μιας δεδομένης πρόκλησης σε ένα πίνακα.

Αργότερα, εισήχθη ο διαγωνισμός ανοιχτής κατηγορίας. Η ανοιχτή κατηγορία είναι ένα θέμα βασιζόμενο στο διαγωνισμό στον οποίο οι ομάδες χρησιμοποιούν τη δημιουργικότητά τους και των δεξιοτήτων τους επίλυσης προβλημάτων στην κατασκευή έξυπνων ρομποτικών λύσεων που συμπληρώνουν το θέμα του διαγωνισμού. Το 2011 το θέμα της ανοιχτής κατηγορίας ήταν "Ρομπότ για τη Βελτίωση της Ζωής".

Το 2010, η WRO εισήγαγε το WRO GEN II Ποδοσφαίρου για πρώτη φορά και είναι πλέον επίσημο στον διαγωνιστικό του WRO. Εδώ οι ομάδες των δύο ρομπότ παίζουν ένα παιχνίδι ποδοσφαίρου με ανταγωνιστικούς όρους.

Τακτική Κατηγορία

Η τακτική κατηγορία είναι μια πρόκληση βασισμένη στον ανταγωνισμό στο οποίο οι μαθητές πρέπει να οικοδομήσουν, κατασκευάσουν και να προγραμματίσουν τα ρομπότ τους για να λύσουν μια πρόκληση. Παραδείγματα θα μπορούσαν να είναι τα ρομπότ που ανεβαίνουν σκάλες και συλλέγουν τις μπάλες πινγκ-πονγκ ή αντικείμενα με διάφορα χρώματα και σχήματα.

Το WRO λειτουργεί με τρεις ηλικιακές κατηγορίες: Δημοτικό, Γυμνάσιο και Λύκειο. Κάθε ηλικιακή κατηγορία μπορεί να αγωνιστεί χρησιμοποιώντας μοναδικές σχεδιασμένες προκλήσεις για να διασφαλιστεί η καταλληλότητα της ηλικίας από άποψη δυσκολίας.

Οι προκλήσεις αναπτύσσονται από τη χώρα υποδοχής του τελικού σε ένα δεδομένο έτος. Οι οδηγίες για το πώς θα κατασκευαστούν οι προκλήσεις γίνονται διαθέσιμες στο διαδίκτυο και διανέμονται στην Εθνική Οργάνωση.

Ανοικτή κατηγορία

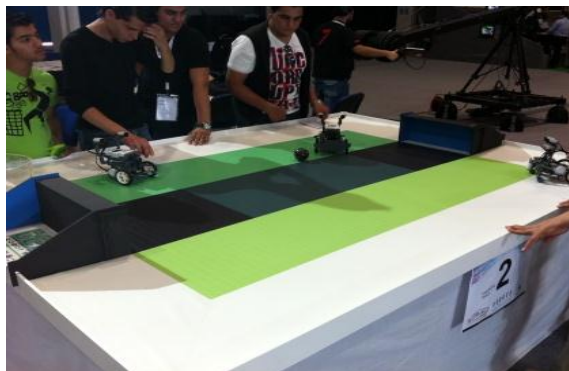
Η ανοικτή κατηγορία είναι ένα θέμα με βάση το ανταγωνισμό. Εδώ οι μαθητές πρέπει να κατασκευάσουν και να εγκαταστήσουν πρόγραμμα που αντιπροσωπεύει το θέμα του διαγωνισμού. Παραδείγματα θεμάτων είναι "Τα Ρομπότ προωθούν τον τουρισμό" (WRO 2010) και "Ρομπότ για τη βελτίωση της ζωής" (WRO 2011). Οι μαθητές ενθαρρύνονται να παρουσιάσουν τα έργα τους με την παραγωγή παρουσιάσεων που εξηγούν το έργο τους. Για να ενθαρρύνει τον ανταγωνισμό, η ανοικτή κατηγορία χωρίζεται επίσης σε τρεις ηλικιακές κατηγορίες: Δημοτικό, Γυμνάσιο και Λύκειο.

WRO GEN II Ποδόσφαιρο

Το WRO GEN II ποδόσφαιρο είναι ένας νέος συναγωνισμός εντός του WRO. Η έννοια του ανταγωνισμού δοκιμάστηκε το 2010 στο τελικό στη Μανίλα και μια πλήρη πιλοτική κλίμακα διοργανώθηκε για το 2011. Το WRO GEN II Football γρήγορα αποδείχθηκε να είναι μια επιτυχία καθώς οι υποστηρικτές των ομάδων παρέχουν μια απίθανη ατμόσφαιρα γύρω από τα γήπεδα.

Το WRO GEN II Football χαρακτηρίζεται από δύο ομάδες των δύο αυτόνομων ρομπότ παίζοντας ένα παιχνίδι των 10 λεπτών. Το WRO λειτουργεί με μόνο μία κατηγορία ηλικίας και όλοι οι μαθητές ηλικίας μεταξύ 10 έως 19 μπορούν να συμμετάσχουν.

Τα υλικά που επιτρέπονται για την κατασκευή των ρομπότ είναι της LEGO ® MINDSTORMS ™ και οι ελεγκτές πρέπει να είναι είτε RCX ή NXT και οι αισθητήρες από HiTechnic.



Εικόνα 1.13: WRO GEN II Football

College Κατηγορία (πilotική)

Η νέα κατηγορία ανταγωνισμού College είναι πιλοτική για το 2013 και επεδείχθη στο WRO 2013 στην Τζακάρτα.

Η συμμετοχή στην WRO 2013 είναι μόνο με πρόσκληση για τον λόγο αυτό οι ενδιαφερόμενοι θα πρέπει να επικοινωνήσουν με τον εθνικό διοργανωτή στη χώρα όπου βρίσκεται για να μάθει αν η κατηγορία College είναι διαθέσιμη στη χώρα του.

Ηλικιακή ομάδα. Η κατηγορία College είναι διαθέσιμη για τους μαθητές από την ηλικία των 17 -21.

Η πρόκληση. Η πρόκληση της κατηγορίας College για το WRO 2013 ονομάζεται "Mars Colony".

Ρομποτικές πλατφόρμες. Η κατηγορία WRO College είναι βασισμένη στο Matrix και στο TETRIX, πλατφόρμες ρομποτικής που χρησιμοποιούν το LabView για MINDSTORMS.

WRO πολιτική πλατφόρμα

Η World Robot Olympiad λειτουργεί με μια ενιαία πολιτική πλατφόρμα. Αυτό σημαίνει ότι το WRO επιτρέπει μόνο τη χρήση του LEGO MINDSTORMS πλατφόρμα ρομποτικής για τρεις βασικές διοργανώσεις. Ο λόγος πίσω από αυτή την πολιτική είναι να διασφαλίσει ίσες ευκαιρίες για όλους τους συμμετέχοντες.

Όλοι οι διαγωνισμοί του WRO στοχεύουν στην πρόκληση των συμμετεχόντων να χρησιμοποιήσουν τη δημιουργικότητά τους για να κατασκευάσουν και να προγραμματίσουν αυτόνομα ρομπότ. Και για το σκοπό αυτό το WRO πιστεύει ότι η LEGO MINDSTORMS πλατφόρμα είναι η πλέον ενδεδειγμένη.

Οργάνωση

Η αρχή που διέπει το WRO είναι η WRO Γνωμοδοτικό Συμβούλιο (AC). Μια μικρότερη ομάδα AC μελών αποτελεί αυτό που ονομάζεται το Διοικητικό Συμβούλιο (BOT). Ο λόγος για αυτή την οργανωτική δομή του WRO είναι ότι από την 1η Ιανουαρίου 2011, η WRO έχει γίνει μια νομική οντότητα, που έχει καταχωρηθεί στη Σιγκαπούρη. Η επίσημη ονομασία του WRO είναι τώρα World Robot Olympiad Association Ε.Π.Ε. Οι καθημερινές δραστηριότητες των WRO διαχειρίζεται η WRO Γενικό Γραμματέα.

WRO Εθνικοί Διοργανωτές

Το World Robot Olympiad είναι παρούσα σε 30 + χώρες σε όλο τον κόσμο. Το WRO ξεκίνησε στην Ασία, αλλά είναι πλέον παρούσα στη Μέση Ανατολή, την Ευρώπη, την Αφρική, τη Νότια Αμερική και την Αυστραλία. Το World Robot Olympiad κάνει δυνατή την ύπαρξή του μόνο με τη βοήθεια των Εθνικών Διοργανωτών από τις χώρες μέλη. Κάθε χρόνο οργανώνουν τοπικές, περιφερειακές και εθνικές διοργανώσεις WRO που το κάνουν διαθέσιμο για έναν αυξανόμενο αριθμό παιδιών κάθε χρόνο.

WRO HELLAS

Ο Οργανισμός Εκπαιδευτικής Ρομποτικής & Επιστήμης W.R.OTM Hellas είναι μία αστική μη κερδοσκοπική εταιρεία με κύριο σκοπό την ανάπτυξη εφαρμογών των φυσικών επιστημών και των νέων τεχνολογιών και ιδιαίτερα μεθόδων και εφαρμογών αυτοματισμών και ρομποτικής στην εκπαίδευση και στην κοινωνία ευρύτερα. Στο πλαίσιο του σκοπού του με δεδομένο ότι επιδιώκουμε την ανοιχτή και απρόσκοπτη ανταλλαγή απόψεων και συνεργασιών με κάθε ενδιαφερόμενο , δίνει βάρος στη προώθηση και ενθάρρυνση όλων των μελών της εκπαιδευτικής κοινότητας αλλά και ιδιωτών /φορέων/ οργανισμών στην ενασχόληση τους με τομείς των νέων τεχνολογιών και της ρομποτικής. Έχοντας ως φιλοδοξία να καταστήσει καταρχήν ενήμερους αλλά κυρίως ενεργούς όλους τους πολίτες της ελληνικής κοινωνίας σε θέματα τεχνολογίας και ρομποτικής, ο WROTM Hellas λειτουργεί με όραμα την τεχνολογική καινοτομία και γι αυτό δίνει έμφαση στους νέους. Αυτό άλλωστε είναι εμφανές και από τη σύνθεση του ,που περιλαμβάνει τους πλέον αναγνωρισμένους επιστήμονες- ακαδημαϊκούς δασκάλους σε παιδαγωγικά και τεχνολογικά θέματα.

ΑΝΑΦΟΡΕΣ

- [1] <http://users.sch.gr/jenyk/index.php/robotics>
- [2] http://courseware.mech.ntua.gr/ml23419/robotics_pdf/intro.pdf
- [3] <http://en.wikipedia.org/wiki/Robotics>
- [4] <http://www.timetoast.com/timelines/history-of-robots--2>
- [5] Papert, S. (1991). Νοητικές θύελλες-Παιδιά, ηλεκτρονικοί υπολογιστές και Δυναμικές ιδέες, Εκδόσεις Οδυσσέας, Αθήνα. Κόμης, Β. (2004). Εισαγωγή στις Εκπαιδευτικές εφαρμογές των Τεχνολογιών της Πληροφορίας και των Επικοινωνιών, Εκδόσεις Νέων Τεχνολογιών Αθήνα.
- [6] Κόμης, Β. (2005). Εισαγωγή στη Διδακτική της Πληροφορικής, Εκδόσεις Κλειδάριθμος.
- [7]http://robotics.ece.auckland.ac.nz/index.php?option=com_content&task=view&id=31
- [8] <http://www.in2life.gr/features/notes/articles/159323/article.aspx>
- [9] http://roboticsinformations.blogspot.gr/2008/03/blog-post_6298.html
- [10] <http://www.paristechreview.com/2013/07/18/history-of-robots/>
- [11] http://en.wikipedia.org/wiki/History_of_robots

- [12] http://en.wikipedia.org/wiki/World_Robot_Olympiad
- [13] <http://www.adec.ac.ae/en/students/athleticeducation/pages/world-robot-olympiad.aspx>
- [14] <http://wrohellas.gr/>
- [15] <http://edurobotics.weebly.com/deltaiotaalphagammaomeganuiotasigmamuomicron943-epsilonkappapialphaiotaepsilonupsilontauiotakappa942sigmaf-rhoomicronmupiomicrontauiotakappa942sigmaf.html>
<http://edurobotics.weebly.com/deltaiotaalphagammaomeganuiotasigmamuomicron943-epsilonkappapialphaiotaepsilonupsilontauiotakappa942sigmaf-rhoomicronmupiomicrontauiotakappa942sigmaf.html>
- [16] <http://www.wroboto.org/media>
- [17] <http://edurobotics.weebly.com/eta-epsilonkappapialphaiotaepsilonupsilontauiotakappa942-pilambdaalphatauphi972rhomualpha-lego-mindstorms-nxt.html>
- [18] http://dide.ilei.sch.gr/keplinet/education/docs/syn_dp2008_LegoNXT.pdf
- [19] http://www.planitikos.gr/2012/12/blog-post_29.html
- [20] http://en.wikipedia.org/wiki/Lego_Mindstorms#Lego_Mindstorms_NXT
- [21] <https://sites.google.com/site/icd11174/>
- [22] http://el.wikipedia.org/wiki/Lego_Mindstorms#.CE.A5.CF.80.CE.BF.CF.83.CF.84.CE.B7.CF.81.CE.B9.CE.B6.CF.8C.CE.BC.CE.B5.CE.BD.CE.B5.CF.82_.CE.93.CE.BB.CF.8E.CF.83.CF.83.CE.B5.CF.82_.CF.80.CF.81.CE.BF.CE.B3.CF.81.CE.B1.CE.BC.CE.BC.CE.B1.CF.84.CE.B9.CF.83.CE.BC.CE.BF.CF.8D
- [23] http://en.wikipedia.org/wiki/Educational_robotics

2.1 ΜΙΚΡΟΕΛΕΓΚΤΕΣ

2.1.1 Ιστορία των μικροεπεξεργαστών-μικροελεγκτών

Η επιθυμία των κατασκευαστικών συστημάτων να δημιουργήσουν συστήματα με περισσότερες δυνατότητες και με μικρότερο μέγεθος οδήγησε στην ανάγκη για ενσωμάτωση όλων των λειτουργιών ενός υπολογιστή σε ένα ή μερικά ολοκληρωμένα κυκλώματα. Η ανάπτυξη της τεχνολογίας που χρησιμοποιήθηκε για την κατασκευή των κυκλωμάτων αυτών είχε σαν αποτέλεσμα την ανάπτυξη του μικροεπεξεργαστή. Οι μικροεπεξεργαστές είναι υπεύθυνοι για την έμπνευση και τη δημιουργία μερικών από τις μεγαλύτερες καινοτομίες στα συστήματα υπολογιστών. Αυτές οι καινοτομίες περιλαμβάνουν τους embedded μικροελεγκτές, τους προσωπικούς υπολογιστές, τους σύγχρονους σταθμούς εργασίας, συσκευές χειρός και κινητές συσκευές (όπως τους επεξεργαστές των κινητών τηλεφώνων), servers (εξυπηρετητές) εφαρμογών και αρχείων, web servers για το internet, υπερυπολογιστές χαμηλού κόστους και ευρείας κλίμακας δίκτυα υπολογιστών. Οι μικροεπεξεργαστές είναι επεξεργαστές συνόλου εντολών (instruction set processors, ISPs). Ένας ISP εκτελεί εντολές ενός προκαθορισμένου συνόλου εντολών. Η λειτουργικότητά του εξαρτάται ανάλογα από το σύνολο εντολών που είναι ικανός να εκτελέσει ο μικροεπεξεργαστής. Σε αυτό το σύνολο των εντολών κωδικοποιούνται όλα τα προγράμματα που τρέχουν σε έναν μικροεπεξεργαστή. Αυτό το προκαθορισμένο σύνολο εντολών ονομάζεται επίσης αρχιτεκτονική συνόλου εντολών (instruction set architecture, ISA). Το ISA χρησιμεύει ως μία διασύνδεση ανάμεσα στο λογισμικό (software) και το υλικό (hardware), δηλαδή ανάμεσα στα προγράμματα και τους επεξεργαστές. Με τον όρο μικροελεγκτή αναφερόμαστε σε έναν τύπο επεξεργαστή ο οποίος μπορεί να λειτουργήσει με ελάχιστα εξωτερικά εξαρτήματα λόγω των πολλών ενσωματωμένων υποσυστημάτων που διαθέτει. Για τον λόγο αυτό θεωρείτε ως παραλλαγή ενός μικροεπεξεργαστή. Ένας μικροελεγκτής είναι ένα ενσωματωμένο τσιπ (ολοκληρωμένο κύκλωμα) που αποτελεί συχνά μέρος ενός συστήματος. Όπως και ένας απλός τυπικός υπολογιστής έτσι και ο μικροελεγκτής περιλαμβάνει CPU, RAM, ROM για αποθήκευση δεδομένων και λογισμικού αντίστοιχα, μνήμη flash για μόνιμη αποθήκευση, θύρες εισόδου/εξόδου, μετατροπέα αναλογικού σε ψηφιακό σήμα και το αντίστροφο και τέλος timers. Καθώς είναι σχεδιασμένος να εκτελεί μόνο μία συγκεκριμένη εργασία για τον έλεγχο ενός απλού συστήματος, είναι πολύ μικρότερος

και απλούστερα σχεδιασμένος ώστε να μπορεί να περιλαμβάνει όλες τις λειτουργίες που απαιτούνται σε ένα μόνο ολοκληρωμένο κύκλωμα. Χρησιμοποιείται ευρέως σε όλα τα ενσωματωμένα συστήματα ελέγχου χαμηλού και μεσαίου κόστους όπως για παράδειγμα αυτά που χρησιμοποιούνται σε αυτοματισμούς, ηλεκτρονικά καταναλωτικά προϊόντα (από ψηφιακές φωτογραφικές μηχανές έως παιχνίδια), ηλεκτρικές συσκευές και κάθε είδους αυτοκινούμενα τροχοφόρα οχήματα. Στις μέρες μας δεν θα ήταν υπερβολή να πούμε ότι η χρήση μικροελεγκτών είναι καθολική για το λόγο ότι κάθε προϊόν το οποίο αλληλεπιδρά με ένα χρήστη περιλαμβάνει ένα μικροελεγκτή, ο οποίος παίζει το ρόλο του «εγκεφάλου» των ηλεκτρονικών κυκλωμάτων.

2.1.2 Η εξέλιξη των μικροεπεξεργαστών

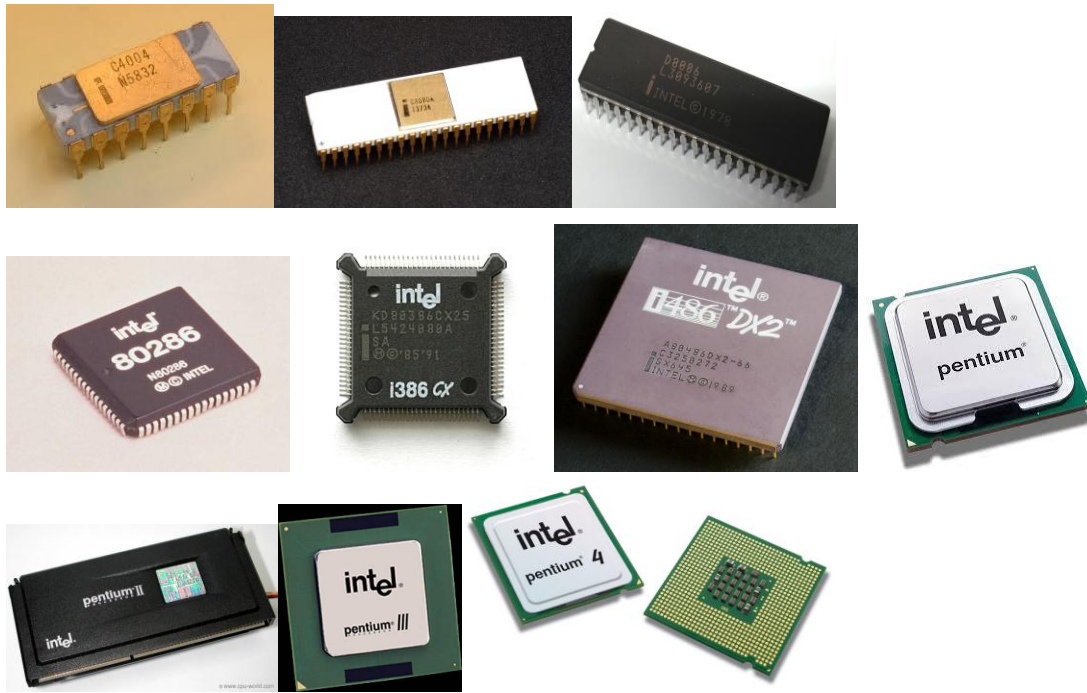
Στις αρχές της δεκαετίας του 70 ο τρόπος κατασκευής των κεντρικών μονάδων επεξεργασίας άλλαξε σημαντικά με την κατασκευή του πρώτου επεξεργαστή από ένα και μόνο ολοκληρωμένο κύκλωμα μεγάλης ολοκλήρωσης. Οι επεξεργαστές αποτελούνταν από δεκάδες πύλες και περίπλοκα κυκλώματα τα οποία με την παρατεταμένη χρήση τους εξέπεμπαν θερμότητα και έπιαναν πολύ χώρο. Καθώς το μέγεθος μειώθηκε οι νέοι επεξεργαστές ονομάστηκαν μικροεπεξεργαστές.

- Το **1971** έχουμε την παρουσίαση του πρώτου μικροεπεξεργαστή Intel 4004 από τον Ted Hoff και τον συνεργάτη του Stan Mazor. Ο Intel 4004 ήταν ένας 4bit επεξεργαστής ο οποίος αποτελούνταν από 2.300 τρανζίστορ με συχνότητα ρολογιού 108KHz, εκτελούσε 60.000 πράξεις το δευτερόλεπτο και μπορούσε να δει 640 bytes μνήμης. Αρχικά εφαρμόστηκε για τη δημιουργία αριθμομηχανών ενώ τον Νοέμβριο του 1971 η Intel ανακοίνωσε τον πρώτο μικροϋπολογιστή. Μέσα στην επόμενη χρονιά εμφανίζεται ο διάδοχός του ο 8008.
- Το **1974** έχουμε την εμφάνιση του 8bit μικροεπεξεργαστή Intel 8080 που ήταν αποτέλεσμα εξέλιξης του 8008. Αποτελούνταν από 6.000 τρανζίστορ με συχνότητα λειτουργίας 2MHz. Την ίδια περίοδο η Motorola παρουσίασε τον 6800 που χρησιμοποιήθηκε σε υπολογιστές, σε όλα τα pinball-παιχνίδια καθώς και σε βιομηχανικές συσκευές ελέγχου. Είχε 4.000 τρανζίστορ, 78 εντολές, σήμα χρονισμού στα 1 ή 2 MHz και 16bit πλάτος διαύλου διευθύνσεων.
- Το **1975** η Zilog φτιάχνει τον Z80 έναν 8bit μικροεπεξεργαστή βασισμένο στον 8080 του οποίου η γλώσσα μηχανής είναι υπερσύνολο αυτής του Intel 8080. Το σήμα χρονισμού του ήταν στα 3.5 MHz με 16 bit πλάτος διευθύνσεων ενώ μπορούσε να δει 64Kbytes μνήμης. Κατά την δεκαετία του

80 είχε την μεγαλύτερη δημοτικότητα καθώς επικεντρώθηκε στο χαμηλό κόστος σε συνδυασμό με τη μικρή συσκευασία, τις χαμηλές απαιτήσεις και τον συνυπολογισμό των στοιχείων κυκλώματος που κανονικά θα έπρεπε να παρασχεθούν σε ένα χωριστό τσιπ.

- Το **1978** εμφανίζονται οι πρώτοι 16bit μικροεπεξεργαστές. Η Intel παρουσιάζει τον 8086/8088 του οποίου η συχνότητα λειτουργίας έχει ανέβει στα 10MHz και απαιτεί 29.000 τρανζίστορ. Αντίστοιχα η Motorola εμφανίζει τον 68000 με συχνότητα λειτουργίας 8MHz και περιέχει 68.000 τρανζίστορ. Ο συνδυασμός της υψηλής ταχύτητας, του μεγάλου χώρου αποθήκευσης (16Mbytes) και του αρκετού χαμηλού κόστους, τον έκανε τον δημοφιλέστερο μικροεπεξεργαστή με αποτέλεσμα να χρησιμοποιηθεί στους υπολογιστές Apple Lisa και Macintosh.
- Το **1982** η Intel δημιούργησε τον 80286 με συχνότητα λειτουργίας αρχικά στα 6 και έπειτα 12.5MHz και 134.000 τρανζίστορ. Περιλάμβανε δίαυλο δεδομένων 16bit, δίαυλο διευθύνσεων 24bit και μπορούσε να δει μέχρι 16Mbytes μνήμης. Είχε δυνατότητα να λειτουργεί στην κατάσταση protected mode (προστατευμένη κατάσταση λειτουργίας). Μετά την επιτυχία της Motorola με τον 68000, οδηγήθηκε στον σχεδιασμό του MC68010 με αποτέλεσμα την προσθήκη υποστήριξης της εικονικής μνήμης.
- Το **1985** εμφανίζονται οι πρώτοι 32bit μικροεπεξεργαστές. Από τη μια ο Intel 80386 όπου περιέχει 275.000 τρανζίστορ και συχνότητα λειτουργίας 33MHz. Η μνήμη που δεχόταν ήταν μέχρι 4 Gbytes. Από την άλλη ο MC86020 της Motorola είχε 200.000 τρανζίστορ και συχνότητα λειτουργίας στα 16MHz. Έγινε ιδιαίτερα δημοφιλής στη microcomputer Unix αγορά ενώ πολλές μικρές επιχειρήσεις παρήγαγαν τα συστήματα desktop.
- Το **1989** κάνει την εμφάνισή του ο 32bit μικροεπεξεργαστής Intel 80486 με 1.200.000 τρανζίστορ και 50MHz συχνότητα λειτουργίας.
- Το **1993** εμφανίζεται ο Intel Pentium της οικογένειας P5 ο οποίος περιέχει 3.100.000 τρανζίστορ και λειτουργούσε στα 60 και 66MHz. Την ίδια περίοδο η Digital παρουσιάζει τον πρώτο 64bit μικροεπεξεργαστή Alpha.
- Το **1997** η Intel εισήγαγε τον μικροεπεξεργαστή Pentium II με τεχνολογία MMX για την υποστήριξη πολυμέσων. Είχε 7.500.000 τρανζίστορ και η συχνότητα λειτουργίας του ήταν στα 300 MHz .
- Το **1999** η Intel ανακοίνωσε τον Pentium III με 9.500.000 τρανζίστορ και η συχνότητα λειτουργίας του βρισκόταν στα 450MHz (και σήμερα έχει φτάσει στα 1.13GHz)
- Το **2000** εμφανίστηκε ο Pentium IV όπου ήταν σχεδιασμένος σύμφωνα με την μικροαρχιτεκτονική NetBurst. Η συχνότητα λειτουργίας του έφτασε στο 1GHz αλλά χωρίς αυτό να παρέχει σημαντική αύξηση της ταχύτητας επεξεργασίας σε σχέση με τον Pentium III.

Από εκεί και μετά έχουμε τους επεξεργαστές στο επίπεδο των πολλαπλών πυρήνων.



Εικόνα 2.1: Intel 4004, 8080, 8086/8088, 80286, 80386, 80486, Pentium, II, III, IV

Η λειτουργία του μικροελεγκτή

Υπάρχουν πολλοί μικροελεγκτές και ακόμα περισσότερα προγράμματα που διατίθενται για μικροελεγκτές, παρόλο αυτά οι περισσότεροι από αυτούς έχουν πολλά πράγματα από κοινού. Έτσι, αν μπορέσουμε να μάθουμε να χειριζόμαστε έναν από αυτούς τότε θα μπορούμε να χειριστούμε και τους άλλους επίσης. Ο μικροελεγκτής είναι μια γρήγορη συσκευή, όχι όσο ο υπολογιστής, έτσι ώστε κάθε εντολή που εκτελείται σε αυτόν να γίνεται με πολύ γρήγορη ταχύτητα. Η λειτουργία του δίνεται παρακάτω:

Όταν ενεργοποιείται η τροφοδοσία, το Μητρώο Λογικού Ελέγχου ενεργοποιεί με τη σειρά του τον ταλαντωτή χαλαζία. Στα πρώτα λίγα χιλιοστά του δευτερολέπτου, ενώ οι πρώτες προετοιμασίες βρίσκονται σε εξέλιξη, οι πυκνωτές παράσιτα φορτίζονται. Όταν η στάθμη της τάσης φτάνει στη μέγιστη τιμή και η συχνότητα του ταλαντωτή χαλαζία γίνεται σταθερή, η διαδικασία της γραφής bits σε ειδικά λειτουργικά μητρώα (SFrs) ξεκινάει. Τα πάντα λαμβάνουν χώρα σύμφωνα με το ρολόι του ταλαντωτή και πάνω από όλα τα ηλεκτρονικά αρχίζουν να δουλεύουν. Όλα αυτά γίνονται σε πολύ λίγα νάνο δευτερόλεπτα. Το PC ή ο μετρητής προγράμματος μηδενίζει τη διεύθυνση

της μνήμης προγράμματος. Στη συνέχεια, η διεύθυνση αποστέλλει τις οδηγίες στον αποκωδικοποιητή ο οποίος αποκωδικοποιεί τις εντολές και έτσι τις εκτελεί. Μετά την εκτέλεση μιας εντολής, η διεύθυνση του απαριθμητή προγράμματος αυξάνεται κατά 1 και ως εκ τούτου στέλνει τη διεύθυνση της επόμενης εντολής στον αποκωδικοποιητή εντολών και εκτελεί τις επόμενες οδηγίες.

2.1.3 Βασικά χαρακτηριστικά που καθιστούν προτιμότερο τον μικροελεγκτή

Το «πακέτο» ενός μικροελεγκτή φέρει κάποια βασικά χαρακτηριστικά που τον καθιστά προτιμότερο για τη χρήση του σε εφαρμογές έναντι της χρήσης των επιμέρους στοιχείων που τον απαρτίζουν ξεχωριστά (επεξεργαστής, μνήμες, συσκευές εισόδου-εξόδου, διεπαφές). Αυτά τα χαρακτηριστικά μπορούν να συνοψισθούν σε:

- **Χαμηλό κόστος.** Είναι ένα από τα βασικότερα χαρακτηριστικά που κάποιος σχεδιαστής λαμβάνει υπόψη. Η συνεχής απελευθέρωση στην αγορά μικροελεγκτών από διάφορες εταιρίες βελτίωσαν την ποιότητα αυτών και μείωσαν τις τιμές λόγω του ανταγωνισμού.
- **Μικρότερο μέγεθος .** Η ολοκλήρωση των βασικών στοιχείων από τα οποία απαρτίζεται μείωσε τις διαστάσεις σε σχέση με τη χρήση των επιμέρους στοιχείων ως σύνολο.
- **Χαμηλή κατανάλωση ισχύος.** Το γεγονός ότι οι μικροελεγκτές λειτουργούν σε συγκριτικά χαμηλές συχνότητες που φτάνουν τα 32KHz, οδηγεί στην κατανάλωση μικρών ποσών ισχύος της τάξης των mW ακόμα και μ W. Επιπλέον έχουν τη δυνατότητα να εισέρχονται σε κατάσταση αναμονής –sleep mode – καταστέλλουν προσωρινά τη λειτουργία της κεντρικής μονάδας επεξεργασίας (ΚΜΕ) και των περιφερειακών , οπότε αυτό μπορεί να γίνει μειώνοντας κατά πολύ την κατανάλωση ισχύος του μικροελεγκτή. Έτσι μπορούν να χρησιμοποιηθούν σε εφαρμογές με αυστηρές απαιτήσεις ως προς αυτήν την παράμετρο.
- **Αυτονομία.** Αυτό επιτυγχάνεται μέσω της ενσωμάτωσης σύνθετων περιφερειακών υποσυστημάτων όπως μνήμες και θύρες επικοινωνίας. Έτσι πολλοί μικροελεγκτές δεν χρειάζονται κανένα άλλο ολοκληρωμένο κύκλωμα για να λειτουργήσουν.

- **Επίτευξη ελέγχου ή μετρήσεων σε πραγματικό χρόνο.** Ενώ οι ηλεκτρονικοί υπολογιστές πρέπει να τρέχουν λειτουργικά συστήματα πραγματικού χρόνου (όπως RT-Linux, QNX κ.ά.) για να το επιτύχουν, οι μικροελεγκτές δεν απαιτούν επιπλέον λογισμικό.

- **Μειωμένες εκπομπές ηλεκτρομαγνητικών παρεμβολών και μειωμένη ευαισθησία σε αντίστοιχες παρεμβολές από άλλες ηλεκτρικές και ηλεκτρονικές συσκευές.** Το πλεονέκτημα αυτό προκύπτει από το μικρότερο αριθμό και μήκος εξωτερικών διασυνδέσεων καθώς και των χαμηλότερων ταχυτήτων λειτουργίας.

- **Περισσότεροι διαθέσιμοι ακροδέκτες για ψηφιακές εισόδους-εξόδους** (για δεδομένο μέγεθος ολοκληρωμένου κυκλώματος) λόγω της μη δέσμευσής τους για τη σύνδεση εξωτερικών περιφερειακών.

- **Η βασική αρχιτεκτονική των μικροελεγκτών δεν διαφέρει από αυτή των κοινών μικροεπεξεργαστών,** αν και στους πρώτους συναντάται συχνά η αρχιτεκτονική μνήμης τύπου Harvard, η οποία χρησιμοποιεί διαφορετικές αρτηρίες σύνδεσης της μνήμης προγράμματος και της μνήμης δεδομένων (π.χ. οι σειρές από την Microchip). Στους κοινούς μικροεπεξεργαστές συνηθίζεται η ενιαία διάταξη μνήμης τύπου φον Νόιμαν

- **Η ενσωμάτωση περιφερειακών σημαίνει ευκολότερη υλοποίηση εφαρμογών** λόγω των απλούστερων διασυνδέσεων. Επιπλέον έχουμε **μεγαλύτερη αξιοπιστία** λόγω των λιγότερων διασυνδέσεων και **μικρό μέγεθος συνολικού υπολογιστικού συστήματος.**

Παρόλο αυτά κάποια από τα μειονεκτήματα του μικροελεγκτή είναι:

- Η μη αλλαγή του προγράμματος για τον λόγο ότι είναι γραμμένο στην ROM
- Η δυσκολία του προγραμματισμού του
- Έχει μεγάλο χρόνο ανάπτυξης. Για να ολοκληρωθεί ένα προϊόν μπορεί να απαιτηθεί από 1 εβδομάδα μέχρι 1 χρόνο.

2.1.4 Διαφορά μεταξύ μικροελεγκτή και μικροεπεξεργαστή

Ο μικροεπεξεργαστής είναι ένα ολοκληρωμένο κύκλωμα (IC) που έχει μόνο τη CPU στο εσωτερικό τους δηλαδή μόνο τις εξουσίες επεξεργασίας, όπως το Pentium I,II,III,IV της Intel κλπ. Αυτοί οι μικροεπεξεργαστές δεν έχουν μνήμη RAM, ROM , και άλλα περιφερειακά στο τσιπ. Για να γίνει λειτουργικός ένας μικροεπεξεργαστής, ο σχεδιαστής του συστήματος πρέπει να τα προσθέσει στο εξωτερικό του. Εφαρμογές του μικροεπεξεργαστή περιλαμβάνουν τα επιτραπέζια PC, οι φορητοί υπολογιστές, τα σημειωματάρια κ.λπ. Αλλά αυτό δεν ισχύει στην περίπτωση με τους μικροελεγκτές. Ο μικροελεγκτής έχει CPU, επιπροσθέτως με ένα σταθερό ποσό της μνήμης RAM, ROM και άλλα περιφερειακά, όλα ενσωματωμένα σε ένα μόνο chip . Μερικές φορές ονομάζεται ως μίνι υπολογιστής ή υπολογιστής με ένα μόνο chip. Σήμερα διαφορετικοί κατασκευαστές παράγουν μικροελεγκτές με ένα ευρύ φάσμα δυνατοτήτων που είναι διαθέσιμοι σε διαφορετικές εκδόσεις. Ορισμένοι κατασκευαστές είναι οι ATMEL, Microchip, Philips, Motorola κ.λπ. Οι μικροελεγκτές είναι σχεδιασμένοι για να εκτελούν συγκεκριμένες διεργασίες. Αυτό σημαίνει εφαρμογές, όπου η σχέση των εισροών και εκροών είναι καθορισμένες. Ανάλογα με την εισροή, πρέπει να γίνει η διεργασία ώστε να έχουμε εκροή. Για παράδειγμα, ηλεκτρολογία, ποντίκια, πλυντήριο ρούχων, φούρνος μικροκυμάτων, αυτοκίνητα , ποδήλατα , τηλέφωνο, κινητά τηλέφωνα , ρολόγια , κλπ. Δεδομένου ότι οι εφαρμογές είναι πολύ συγκεκριμένες , χρειαζόμαστε μικρούς πόρους, όπως μνήμη RAM, ROM, θύρες I / O κλπ και ως εκ τούτου να μπορούν να ενσωματωθούν σε ένα μόνο chip . Αυτό με τη σειρά του μειώνει το μέγεθος και το κόστος. Οι μικροεπεξεργαστές βρίσκουν εφαρμογές όπου τα καθήκοντα είναι αόριστα όπως η ανάπτυξη λογισμικού , παιχνίδια , ιστοσελίδες , επεξεργασία φωτογραφιών , δημιουργία εγγράφων κλπ. Σε τέτοιες περιπτώσεις, η σχέση μεταξύ εισόδου και εξόδου δεν ορίζεται. Χρειάζονται υψηλό ποσό πόρων, όπως RAM , ROM , θύρες I / O κλπ. Η ταχύτητα ρολογιού του μικροεπεξεργαστή είναι αρκετά υψηλή σε σύγκριση με του μικροελεγκτή. Ενώ οι μικροελεγκτές λειτουργούν από μερικά MHz έως 30 με 50 MHz, οι σημερινοί μικροεπεξεργαστές λειτουργούν πάνω από 1GHz , δεδομένου ότι εκτελούν πολύπλοκα καθήκοντα. Η σύγκριση του μικροελεγκτή και του μικροεπεξεργαστή όσον αφορά το κόστος δεν είναι δικαιολογημένη. Αναμφίβολα ένας μικροελεγκτής είναι πολύ φθηνότερος από ένα μικροεπεξεργαστή. Ωστόσο ένας

μικροελεγκτής δεν μπορεί να χρησιμοποιηθεί στη θέση ενός μικροεπεξεργαστή και αντίστοιχα η χρήση ενός μικροεπεξεργαστή δεν συνιστάται στη θέση ενός μικροελεγκτή, καθώς κάνει την εφαρμογή αρκετά δαπανηρή. Ο μικροεπεξεργαστής δεν μπορεί να χρησιμοποιηθεί αυτόνομα. Χρειάζεται άλλα περιφερειακά όπως RAM, ROM, buffer, θύρες I/O κλπ και ως εκ τούτου ένα σύστημα σχεδιασμένο γύρω από έναν μικροεπεξεργαστή, είναι αρκετά δαπανηρό.

2.1.5 Διαδεδομένες κατηγορίες μικροελεγκτών

Λόγω της τάσης ενσωμάτωσης των μικροελεγκτών σε κάθε ηλεκτρική και ηλεκτρονική συσκευή αλλά και του ισχυρότατου ανταγωνισμού, έχουμε καταλήξει στην παραγωγή ανταγωνιστικών μοντέλων μαζικής παραγωγής καθώς και στην παραγωγή μικροελεγκτών για πιο εξειδικευμένες εφαρμογές. Έτσι διακρίνονται οι εξής κυρίως κατηγορίες:

- Μικροελεγκτές (καμιά φορά 4-bit αλλά συνήθως 8-bit) πολύ χαμηλού κόστους, γενικής χρήσης, με πολύ μικρό αριθμό ακροδεκτών (ακόμη και λιγότερους από 8). Για να μη μπορεί να αντιγραφεί εύκολα το εσωτερικό λογισμικό τους σχεδιάζονται με έμφαση στη χαμηλή κατανάλωση ισχύος και την αυτάρκεια. Απουσιάζει η δυνατότητα επέκτασης της μνήμης τους. Μερικά μοντέλα είναι ευρέως γνωστά στους ερασιτέχνες ηλεκτρονικούς, όπως πχ οι περισσότεροι μικροελεγκτές των σειρών PIC (Microchip), AVR (Atmel) και 8051 (Intel, Atmel, Dallas κα).
- Μικροελεγκτές (συνήθως 8-bit αλλά και 16 ή 32-bit) χαμηλού κόστους, γενικής χρήσης, με μέτριο έως σχετικά μεγάλο αριθμό ακροδεκτών. Διαθέτουν μεγάλο αριθμό κοινών περιφερειακών, όπως θύρες UART, I²C, SPI ή CAN, μετατροπείς αναλογικού σε ψηφιακό και ψηφιακού σε αναλογικό. Στους κατασκευαστές της Άπω Ανατολής (Ιαπωνία, Κορέα), συνηθίζεται η ενσωμάτωση ελεγκτών οθόνης υγρών κρυστάλλων και πληκτρολογίου. Μερικές φορές παρέχουν δυνατότητα εξωτερικής επέκτασης της μνήμης τους.
- Μικροελεγκτές (κυρίως 32-bit) μέσου κόστους, γενικής χρήσης, με μεγάλο αριθμό ακροδεκτών. Χαρακτηρίζονται από έμφαση στην ταχύτητα εκτέλεσης εντολών, υψηλή αυτάρκεια περιφερειακών και μεγάλες δυνατότητες εσωτερικής ή εξωτερικής μνήμης προγράμματος (FLASH) και RAM. Στο χώρο αυτό έχουν ισχυρή παρουσία οι αρχιτεκτονικές με υψηλή μεταφερσιμότητα λογισμικού (portability) από τον ένα στον άλλο κατασκευαστή. Πχ μεταξύ των μικροελεγκτών τύπου ARM ή MIPS, το σύνολο των βασικών εντολών που αναγνωρίζει η ALU είναι ακριβώς το ίδιο, μειώνοντας έτσι τις μεγάλες αλλαγές στο λογισμικό, όταν στο μέλλον ο πελάτης υιοθετήσει ένα μικροελεγκτή άλλου κατασκευαστή (αρκεί, φυσικά, να υποστηρίξει κι αυτός το σύνολο εντολών ARM ή MIPS, αντίστοιχα).
- Μικροελεγκτές εξειδικευμένων εφαρμογών, οι οποίοι ενσωματώνουν συνήθως κάποιο εξειδικευμένο πρωτόκολλο επικοινωνίας το οποίο υλοποιείται

πάντοτε σε hardware. Τέτοιοι μικροελεγκτές χρησιμοποιούνται σε τηλεπικοινωνιακές συσκευές όπως τα μόντεμ.

Η μεγάλη μερίδα πωλήσεων των μικροελεγκτών εξακολουθεί να αφορά αυτούς των 8-bit, καθώς είναι η κατηγορία με το χαμηλότερο κόστος και το μικρότερο μέγεθος λογισμικού για το ίδιο αποτέλεσμα, ιδίως επειδή οι σύγχρονες οικογένειες μικροελεγκτών 8-bit έχουν πολύ βελτιωμένες επιδόσεις σε σχέση με το παρελθόν.

2.1.6 Εργαλεία ανάπτυξης και γλώσσες προγραμματισμού

Η επιτυχία καθορίζεται σε μεγάλο βαθμό από τη διαθεσιμότητα και την ευχρηστία των σχετικών εργαλείων ανάπτυξης, όπως μεταφραστές από γλώσσες υψηλού επιπέδου σε γλώσσα κατανοητή από τον μικροελεγκτή (Assembly), προγραμματιστές της εσωτερικής μνήμης και εργαλεία εκσφαλμάτωσης (Debugger). Στους μικροελεγκτές, τα εργαλεία αυτά δεν αποτελούνται ποτέ μόνο από το λογισμικό, καθώς δεν υπάρχει τυποποιημένος τρόπος επικοινωνίας με αυτούς. Στον τομέα των εργαλείων ανάπτυξης, δραστηριοποιούνται όχι μόνο οι ίδιοι οι κατασκευαστές μικροελεγκτών αλλά και εξειδικευμένες εταιρείες.

Η C, η C++ και οι παραλλαγές τους είναι από τις πιο διαδεδομένες γλώσσες προγραμματισμού των μικροελεγκτών. Η Assembly μπορεί να χρησιμοποιείται σε τμήματα του λογισμικού όπου απαιτείται ταχύτητα ή μικρό μέγεθος χρησιμοποιούμενης μνήμης. Όμως οι μεγαλύτερες απαιτήσεις σε λειτουργικότητα και η ευκολία προγραμματισμού της C έναντι της Assembly, σε συνδυασμό με την επάρκεια μνήμης των σύγχρονων μικροελεγκτών, έχουν εκτοπίσει την Assembly από τις περισσότερες εφαρμογές.

2.1.7 Κατασκευαστές μικροελεγκτών

Οι περισσότερες εταιρείες παράγουν μεγάλη γκάμα μικροελεγκτών. Από πολύ μικρούς και φθηνούς για απλές εφαρμογές έως ιδιαίτερα προηγμένους για πολύ απαιτητικές εφαρμογές. Μερικοί από τους γνωστότερους κατασκευαστές μικροελεγκτών είναι οι

- ARM (δεν κατασκευάζει αλλά παραχωρεί δικαιώματα χρήσης του πυρήνα)
- Atmel
- Epson
- Freescale Semiconductor (πρώην Motorola)
- Hitachi
- Maxim (μετά την εξαγορά της Dallas)
- Microchip
- NEC
- Toshiba
- Texas Instruments
- Intel
- Analog Devices

ΑΝΑΦΟΡΕΣ:

[1]<http://el.wikipedia.org/wiki/%CE%9C%CE%B9%CE%BA%CF%81%CE%BF%CE%B5%CE%BB%CE%B5%CE%B3%CE%BA%CF%84%CE%AE%CF%82>

[2] <http://www.newagepublishers.com/samplechapter/001599.pdf>

[3]<http://www.engineersgarage.com/tutorials/difference-between-microprocessor-and-microcontroller>

[4] <http://www.engineersgarage.com/microcontroller>

[5]<http://www.mikroe.com/chapters/view/65/chapter-2-8051-microcontroller-architecture/>

[6] <http://www.micropplanet.gr/>

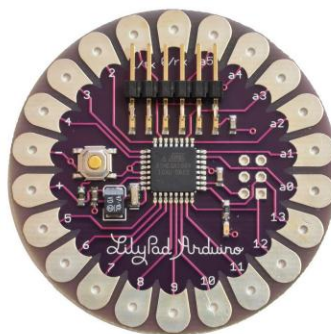
[7] <http://cgi.di.uoa.gr/~std04013/>

2.2 Ο μικροελεγκτής ARDUINO

2.2.1 Ιστορία του Arduino

Το 2005 κάνει την εμφάνισή του το Arduino, όταν ο καθηγητής Massimo Banzi στο ινστιτούτο σχεδιασμού αλληλεπίδρασης στην πόλη της Ivrea, στην Ιταλία, θέλησε να καταστήσει ευκολότερη τη μάθηση των ηλεκτρονικών για τους μαθητές. Ήθελε να δώσει στους μαθητές την ευκαιρία να ανακαλύψουν πράγματα οι ίδιοι, αντί να τα ακούν θεωρητικά χωρίς να κάνουν κάτι. Για τον σκοπό αυτό ζήτησε βοήθεια από τον David Cuatrecasas, έναν μηχανικό από το Πανεπιστήμιο Malmö και μαζί αποφάσισαν να κάνουν ένα μικροελεγκτή που θα ήταν πιο προσιτός ως προς τη χρήση του. Δύο φοιτητές επιλέχθηκαν να γράψουν το λογισμικό για τη συσκευή. Ο Ηλεκτρολόγος μηχανικός Gianluca Martino, κλήθηκε να κάνει μια αρχική παρτίδα των 200 μικροελεγκτών. Το όνομα Arduino δόθηκε από ένα ιστορικό χαρακτήρα, τον Arduino της Ivrea. Το πρώτο Arduino που φτιάχτηκε ονομάστηκε "Serial Arduino» και περιελάμβανε μια ATmega8 με άμεση σύνδεση RS-232 με το μικροελεγκτή και όλα τα συστατικά του. Στη συνέχεια σχεδιάστηκε η έκδοση 2.0 και μια μονόπλευρη έκδοχή σαφέστερη για τους χομπίστες. Οι εκδόσεις που ακολούθησαν ήταν όλες

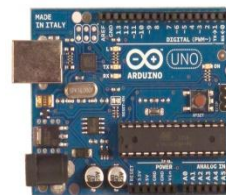
FTDI USB μετατροπέα. Μετά το USB v1.0 and v2.0, κυκλοφόρησε το Arduino Extreme το οποίο αύξησε την ποσότητα των επιφανειακών εξαρτημάτων. Το Arduino Nuova Generazione μεταβαίνει σε έναν απλούστερο μετατροπέα USB και μετατρέπεται από το ATmega8 σε ATmega168. Οι βελτιώσεις συνεχίστηκαν με το Diecimila, το Duemilanove και το Uno αντικαθιστώντας το μετατροπέα FTDI ATmega8U2, ο οποίος περιέχει έναν ελεγκτή USB και αναβαθμίζει τον μικροελεγκτή σε ATmega328. Η τελευταία έκδοση του βασικού Arduino Leonardo, κινείται εύκολα από εναλλάξιμη υποδοχή μικροελεγκτή σε μια επιφάνεια που έχει τοποθετηθεί συγκόλληση και απαιτεί να αλλαχθεί, καθώς εξαλείφει τον μετατροπέα πλήρως αναβαθμίζοντάς το σε ένα ATmega32U4, το οποίο περιέχει ένα ελεγκτή USB απλουστεύοντας έτσι τον σχεδιασμό σε μεγάλο βαθμό. (Massimo 2008.). Έχουν φτιαχτεί επίσης εξειδικευμένες εκδόσεις. Οι μικρότερες εκδόσεις, όπως Mini και Nano επιτρέπουν μικρότερες εγκαταστάσεις. Έχουμε μεγαλύτερες εκδόσεις με το ATmega1260 και το ATmega2560 για μεγαλύτερη επεκτασιμότητα. Μερικές εκδόσεις είναι επίσης διαθέσιμες σε μορφή Arduino BT για συνδεσιμότητα με Bluetooth και σε μορφή Arduino Ethernet για σύνδεση με LAN το οποίο έχει επίσης υποστήριξη για Power over Ethernet, εάν μια add-on μονάδα έχει εγκατασταθεί στο πλακίδιο. Το Arduino LilyPad είναι η σαφής υπόδειξη που έχει σχεδιαστεί για ευελιξία. Μπορούν να συνδεθούν 6 προστασίες στις πινέζες που είναι διαθέσιμες στο Arduino και περιλαμβάνει προστασία όπως Wi-Fi, Ethernet, Wireless, Motor και Proto για τη προτυποποίηση. Η Wireless ασπίδα έρχεται είτε με υποδοχή SD ή χώρο πρωτοτύπων και έχει κεφαλίδες για τη σύνδεση των μονάδων που χρησιμοποιούν το XBee αποτύπωμα. Το Motor επιτρέπει τον έλεγχο των 2 κινητήρων συνεχούς ρεύματος. Ο καθένας οπουδήποτε μπορεί να κάνει ασπίδες για αυτούς, εάν μπορούν να παράγουν τυπωμένα κυκλώματα. (Massimo 2008.)



Εικόνα 2.2: Arduino LilyPad

2.2.2 Τι είναι το Arduino

το Arduino είναι μια «ανοικτού κώδικα» πλατφόρμα «πρωτοτυποποίησης» ηλεκτρονικών. Βασίζεται στο ευέλικτο και εύκολο στη χρήση hardware και software που προορίζεται για οποιονδήποτε έχει λίγη προγραμματιστική εμπειρία, στοιχειώδεις γνώσεις ηλεκτρονικών και ενδιαφέρεται να δημιουργήσει διαδραστικά αντικείμενα ή περιβάλλοντα. Ουσιαστικά, πρόκειται για ένα ηλεκτρονικό κύκλωμα που βασίζεται στον μικροελεγκτή ATmega της Atmel και του οποίου όλα τα σχέδια, καθώς και το software που χρειάζεται για την λειτουργία του, διανέμονται ελεύθερα και δωρεάν ώστε να μπορεί να κατασκευαστεί από τον καθένα . Αφού κατασκευαστεί, μπορεί να συμπεριφερθεί σαν ένας μικροσκοπικός υπολογιστής, αφού ο χρήστης μπορεί να συνδέσει επάνω του πολλαπλές μονάδες εισόδου/εξόδου και να προγραμματίσει τον μικροελεγκτή να δέχεται δεδομένα από τις μονάδες εισόδου, να τα επεξεργάζεται και να στέλνει κατάλληλες εντολές στις μονάδες εξόδου.



2.2.3 Η γλώσσα προγραμματισμού που χρησιμοποιείται

Η γλώσσα προγραμματισμού που χρησιμοποιεί είναι η Wiring, η οποία είναι αρκετά εύκολη στη σύνταξη και διατίθεται σε πλατφόρμες Linux, MAC και Windows με άδεια χρήσης GPL. Αυτό όμως που κάνει το Arduino ακόμα πιο σημαντικό είναι ότι όλο το κύκλωμα της πλακέτας διατίθεται με άδεια χρήσης Creative Commons, πράγμα που σημαίνει ότι ο καθένας μπορεί να κατασκευάσει την δική του πλακέτα όπως αυτός επιθυμεί. Μάλιστα κάποιος θα μπορούσε να ισχυριστεί - και θα ήταν ένας αρκετά πετυχημένος παραλληλισμός - ότι λειτουργικά το Arduino μοιάζει πολύ με το NXT Brick των Lego Mindstorms NXT. Άλλωστε το Arduino διαπρέπει στις εφαρμογές της ρομποτικής. Το Arduino βέβαια, δεν είναι ούτε ο μοναδικός, ούτε και ο καλύτερος δυνατός τρόπος για την δημιουργία μιας οποιασδήποτε διαδραστικής ηλεκτρονικής συσκευής.

2.2.4 Πλεονεκτήματα του ARDUINO

Το κύριο πλεονέκτημα του Arduino είναι η τεράστια κοινότητα που το υποστηρίζει και η οποία έχει δημιουργήσει, επεκτείνει και συντηρήσει μια ανάλογου μεγέθους online

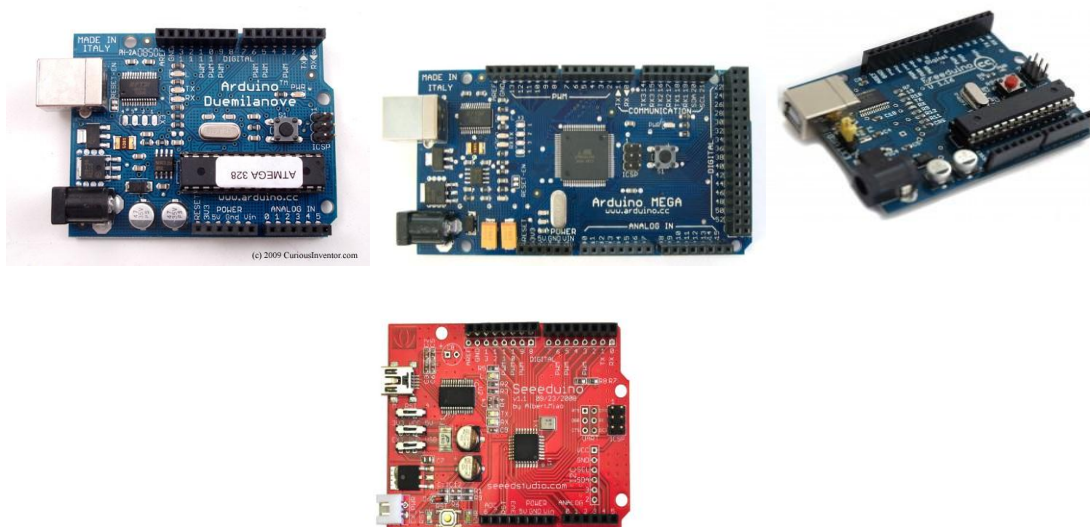
γνωστική βάση. Έτσι, ενώ ένας έμπειρος ηλεκτρονικός μπορεί να προτιμήσει μια διαφορετική πλατφόρμα ή εξαρτήματα ανάλογα με την εφαρμογή που έχει στον νου του, το Arduino, με την εκτενές τεκμηρίωση, καταφέρνει να κερδίσει όλους αυτούς των οποίων οι γνώσεις στα ηλεκτρονικά περιορίζονται στα όσα λίγα έμαθαν στο σχολείο με ένα ξεκάθαρο προγραμματιστικό περιβάλλον. Ακριβώς επειδή απευθύνεται κυρίως σε αρχάριους των ηλεκτρονικών και επειδή, παρά τις αναλυτικότερες οδηγίες που υπάρχουν, δεν έχουν όλοι τις γνώσεις και τα μέσα να κατασκευάσουν μια ηλεκτρονική πλακέτα, κυκλοφορούν έτοιμες, προκατασκευασμένες πλακέτες Arduino στο διαδίκτυο σε προσιτές τιμές. Επίσης οι περισσότεροι προμηθευτές με λίγα χρήματα παραπάνω μπορούν να διαθέτουν το Arduino Starter Kit, το οποίο, εκτός από το ίδιο το Arduino, περιέχει διάφορα άλλα εξαρτήματα και εργαλεία που μπορεί να χρειαστούν για τις εφαρμογές (όπως το απαραίτητο καλώδιο USB για την σύνδεση με τον υπολογιστή, καλώδια, LED, διακόπτες, ποτενσιόμετρα, αντιστάσεις, διόδους, τρανζίστορ κ.λπ.) Επιπλέον βλέπουμε ότι τρέχει σε διάφορα Λειτουργικά Συστήματα. Οι μηχανικοί λογισμικού, ανέπτυξαν το περιβάλλον προγραμματισμού του Arduino για Windows, Macintosh OSX και για λειτουργικά συστήματα Linux. Τα περισσότερα συστήματα ανάπτυξης Μικροελεγκτών περιορίζονται στα Windows.

2.2.5 Δυνατότητες του ARDUINO

Αν και μικροσκοπικό (7x5 cm) οι δυνατότητες που προσφέρει είναι πάρα πολλές. Μπορούμε να το χρησιμοποιήσουμε σε εφαρμογές ρομποτικής και γενικότερα σε αυτοματισμούς καταφέροντας έτσι πάρα πολλά όπως: την κίνηση servo, stepper και DC κινητήρων, τη λήψη πληροφοριών από διάφορους αισθητήρες (θερμοκρασίας, υγρασίας, υπερύθρων κ.α), την αμφίδρομη σειριακή επικοινωνία μεταξύ Arduino και PC χρησιμοποιώντας γλώσσες προγραμματισμού (όπως Java και python), όπως επίσης την αναπαραγωγή και αντίληψη ήχων. Η πλακέτα Arduino μέχρι αυτή τη στιγμή διατίθεται σε 12 βασικές παραλλαγές οι οποίες αναφέρονται σε διαφορετικές χρήσεις η κάθε μια, ανάλογα με τις ανάγκες της εφαρμογής μας.

2.2.6 Εκδόσεις ARDUINO

Το Arduino κυκλοφορεί σε διαφορετικές εκδόσεις, επίσημες και ανεπίσημες. Από τις επίσημες εκδόσεις (Duemilanove, Diecimila, Nano, Mega, Bluetooth, LilyPad, Mini, Mini USB, Pro, Pro Mini, Serial και Serial SS) συνιστάται κυρίως η αγορά του Arduino Duemilanove ή τουλάχιστον των Diecimila ή Mega επειδή διαθέτουν υποδοχή USB και είναι συμβατές με τα shield. Για τους ίδιους λόγους, από τις ανεπίσημες εκδόσεις (Freeduino, Boarduino, Sanguino, Seeduino, BBB, RBBB κ.α.) συνιστάται μόνο το Freeduino v1.16 και το Seeduino.



Εικόνα 2.3: Arduino Duemilanove, Mega, Freeduino v1.16, Seeeduino

2.2.7 Εφαρμογές Arduino

Όπως αναφερθήκαμε και προηγουμένως το Arduino είναι ένας ελεγκτής που έχει την ικανότητα να επικοινωνεί μέσω σειριακής θύρας με άλλες συσκευές δίνοντάς του τη δυνατότητα να αλληλεπιδράσει σε συνεργασία διαφόρων προγραμμάτων του υπολογιστή. Σύμφωνα με τις απαιτήσεις του προγραμματιστή οι γλώσσες προγραμματισμού στα οποία έχουν αναπτυχθεί τα προγράμματα ποικίλουν. Ο πιο διαδεδομένος τρόπος για την ανάπτυξη προγραμμάτων που έχουν την ικανότητα να αλληλεπιδρούν με το Arduino είναι η γλώσσα προγραμματισμού Processing. Μερικές εφαρμογές που αναπτύχθηκαν με τον ελεγκτή Arduino αλλά και με τη βοήθεια άλλων προγραμμάτων θα αναλυθούν στη συνέχεια.

Εφαρμογές του Arduino στην μουσική

Laser harp

Το πιο γνωστό project με την χρήση του Arduino είναι το laser harp (laser άρπα). Πιθανόν να παρουσιάστηκε για πρώτη φορά από τον Bernard Szajner το 1981, αλλά έγινε δημοφιλές από τον Jean Michel Jarre όπου και το χρησιμοποιούσε στις συναυλίες του. Το 2009 ο Stephen Hobley επηρεασμένος από μία συναυλία του Jarre αποφάσισε να κατασκευάσει τη δική του laser άρπα. Το laser harp είναι ένα μουσικό όργανο που αλληλεπιδράει με το φως. Στην πραγματικότητα αποτελείται από μία συσκευή η οποία παράγει μία σειρά κάθετων φωτεινών γραμμών (laser) που ξεκινούν από το δάπεδο. Ο μουσικός διακόπτοντας τις δέσμες φωτός παράγει μία ποικιλία μουσικών ήχων. Ρόλο για την παραγωγή του ήχου δε παίζει μόνο η διακοπή της δέσμης αλλά και το ύψος που θα

τοποθετηθεί το εμπόδιο από το δάπεδο. Η άρπα δεν παράγει από μόνη της τους ήχους, απαιτείται να συνδεθεί με ένα νέας τεχνολογίας synthesizer ώστε να λαμβάνει από το Arduino σειριακά τα MIDI (Musical Instrument Digital Interface) δεδομένα που παράγει.



Εικόνα 2.4: Laser harp

Soundmachine

Το Σεπτέμβριο στην έκθεση μηχανοκίνητων IAA 2011 της Φρανκφούρτης, η σχεδιαστική ομάδα The Product σε ένα ειδικό τμήμα που αφορούσε νέες τεχνολογίες οι οποίες αλληλεπιδρούν με τον άνθρωπο παρουσίασε την Soundmachine. Η εξωτερική εμφάνιση της κατασκευής όπως φαίνεται και στην εικόνα 18 μοιάζει αρκετά με ένα κλασικό πικάπ, η λειτουργία της όμως είναι εντελώς διαφορεική. Η βελόνα του πικάπ έχει αντικατασταθεί με ένα led το οποίο φωτίζει την περιστρεφόμενη πλατφόρμα και έναν αισθητήρα φωτός που ανιχνεύει το αντανακλώμενο φως. Η κυκλική περιστρεφόμενη πλατφόρμα η οποία στην πραγματικότητα έχει αντικαταστήσει το βινύλιο αποτελείται από διάφορες χρωματιστές ακτίνες. Οι ακτίνες με τη σειρά τους ανάλογα από το χρώμα που έχουν, αντανακλούν το φως στον αισθητήρα ο οποίος μετατρέπει το φως σε ηλεκτρικά σήματα και τα στέλνει στο Arduino. Ο μικροελεγκτής είναι υπεύθυνος για την μετατροπή των σημάτων στην κατάλληλη μορφή και στη συνέχεια για την αποστολή τους στον υπολογιστή. Από το σημείο αυτό και μετά την εργασία αναλαμβάνει μία ειδική εφαρμογή που έχει αναπτυχθεί σε processing όπου ανάλογα το φως που αντανακλάται από την πλατφόρμα, ο υπολογιστής παίζει τα ανάλογα μουσικά loops.



Εικόνα 2.5: Soundmachine

Εφαρμογές του Arduino στον μοντελισμό

Μία άλλη διαδεδομένη εφαρμογή του Arduino κυρίως ανάμεσα στους φίλους του μοντελισμού είναι η κατασκευή τηλεκατευθυνόμενου οχήματος. Για τη κατασκευή ενός τέτοιου οχήματος απαιτείται ένας μικροελεγκτής που θα είναι υπεύθυνος για τον έλεγχο του οχήματος (Arduino), επίσης απαραίτητο είναι και ένα σύστημα για την ασύρματη επικοινωνία του Arduino με την συσκευή που θα το τηλεκατευθύνει (όπως για παράδειγμα η shield x-bee). Ο δουλειά του μικροελεγκτή είναι να ελέγχει τους απαραίτητους κινητήρες και σερβοκινητήρες οι οποίοι τροφοδοτούνται από μία πηγή ρεύματος (συνήθως μπαταρίες) για την κατεύθυνση του σκάφους. Η τηλεκατεύθυνση μπορεί να προσαρμοστεί σε οποιαδήποτε μορφή σκάφους (αυτοκίνητο, πλοίο, αεροπλάνο). Εκτός όμως του βασικού εξοπλισμού, οι μοντελιστές έχουν την δυνατότητα να προσθέσουν περαιτέρω συσκευές ανάλογες των απαιτήσεων τους. Όπως για παράδειγμα είναι η shield GPS που ενημερώνει το Arduino με τις συντεταγμένες του σημείου που βρίσκεται. Με τον τρόπο αυτό το όχημα γνωρίζοντας τον προορισμό του βάση των συντεταγμένων που του δόθηκαν, θα μπορεί να κινηθεί προς αυτόν χωρίς να χρειάζεται κάποιος να το κατευθύνει.



Εικόνα 2.6: τηλεκατευθυνόμενο όχημα

Εφαρμογές του Arduino για την διακόσμηση

Led cube

Μία ενδιαφέρουσα και σχετικά απλή εφαρμογή είναι η κατασκευή κύβου με led. Το led cube σχηματίζεται από led τα οποία είναι διατεταγμένα σε θέσεις ίσης απόστασης μεταξύ τους. Συνήθως ο κύβος είναι 4x4x4, δηλαδή αποτελείται από τέσσερις στρώσεις, και η κάθε μία με την σειρά της αποτελείται από τέσσερις γραμμές και τέσσερις στήλες. Τα led αναβοσβήνουν σύμφωνα με το Arduino είτε τυχαία είτε με τέτοιο τρόπο ώστε να σχηματίσουν ένα σχήμα. Με την χρήση περισσότερων led είναι δυνατή η αναπαράσταση τρισδιάστατων γραμμάτων ή ακόμη και εικόνων. Τέτοιου είδους εφαρμογές συνήθως χρησιμοποιούνται για διακοσμητικούς ή διαφημιστικούς σκοπούς.



Εικόνα 2.7: Led Cube

Εφαρμογές του Arduino στο σπίτι

Το Arduino επίσης μπορεί να χρησιμοποιηθεί για να διευκολύνει τον τρόπο διαβιώσεις των ανθρώπων μέσα στο σπίτι. Αυτό επιτυγχάνεται με διάφορες κατασκευές

Arduino security alarm

Ο ελεγκτής έχει την δυνατότητα να συνδυαστεί με μία ποικιλία αισθητήρων με σκοπό να λειτουργεί ως ένας συναγερμός σπιτιού. Έτσι ανιχνεύοντας για παράδειγμα την κίνηση, ελέγχοντας συγκεκριμένες πόρτες και παράθυρα ακόμη και την

θερμοκρασία σε περίπτωση πυρκαγιάς μπορεί να γίνει ένα αξιόπιστο σύστημα συναγερμού, ενώ με ένα κατάλληλο πρόγραμμα στον υπολογιστή έχει την δυνατότητα ακόμη και να ενημερώνει απομακρυσμένα τους υπεύθυνους του κτιρίου.

RGB combination door lock

Μία άλλη γνωστή κατασκευή είναι η ηλεκτρονική κλειδαριά για πόρτες. Σε αντίθεση με τις κοινές ηλεκτρονικές κλειδαριές που απαιτείται να πληκτρολογηθεί ένας κωδικός ο οποίος αποτελείται από αριθμούς, το Arduino δίνει την δυνατότητα της κατασκευής μιας κλειδαριάς όπου ο κωδικός αποτελείται από χρώματα. Όπως φαίνεται στην εικόνα 2.8 (δίπλα), πάνω στο πληκτρολόγιο αντί να αναγράφονται οι αριθμοί, το κάθε ένα πλήκτρο φωτίζεται εσωτερικά με ξεχωριστά χρώματα. Έτσι αντί να πληκτρολογείτε ένας αριθμός για κωδικό από το κάθε άτομο που θέλει να ανοίξει την πόρτα, θα πρέπει να πληκτρολογηθούν τα φωτισμένα με χρώματα πλήκτρα με την σωστή σειρά.

Εφαρμογές Arduino σε παιχνίδια

Όπως είναι φυσικό οι καινοτομίες του Arduino δεν πέρασαν απαρατήρητες και από τους ανθρώπους που ασχολούνται με την κατασκευή πρωτότυπων παιχνιδιών. Η επικοινωνία του ελεγκτή με τον υπολογιστή έδωσε την δυνατότητα για την εύκολη κατασκευή παιχνιδιών που αλληλεπιδρούν με τον άνθρωπο.

Marble labyrinth ελεγχόμενο με τη χρήση the WiiFit

Το marble labyrinth είναι ένα είδος λαβυρίνθου που περιέχει μία μεταλλική μπίλια. Σκοπός του παιχνιδιού είναι η μπίλια να διασχίσει μία συγκεκριμένη διαδρομή του λαβύρινθου για να φτάσει στον τερματισμό. Μέσα στους διαδρόμους υπάρχουν διάφορες τρύπες που θα πρέπει να τις προσπεράσει χωρίς να πέσει μέσα ώστε να μην χρειαστεί να ξαναρχίσει την διαδρομή από την αρχή. Για να μετακινηθεί η μπίλια, πρέπει ο παίχτης να αλλάζει συνεχώς το ύψος των πλευρών του λαβύρινθου από το



δάπεδο, έτσι λόγω της βαρύτητας η μπίλια θα αναγκαστεί να κινηθεί προς την κατεύθυνση της πιο χαμηλής πλευράς του λαβύρινθου. Αντί ο χρήστης να χρειάζεται να σηκώσει ολόκληρη την πλατφόρμα του λαβύρινθου για να ελέγξει την

κίνηση της μπίλιας, σταθεροποίησαν τον λαβύρινθο πάνω σε μία άλλη πλατφόρμα η οποία με την βοήθεια δύο σερβοκινητήρων μπορεί να αλλάζει το ύψος των πλευρών του λαβύρινθου και να αναγκάσει την μπίλια να κινηθεί. Ο παίχτης για να κατευθύνει την μπίλια πατάει πάνω σε ένα WiiFit, μία επιδαπέδια πλατφόρμα της παιχνιδομηχανής wii της NINTENDO. Η πλατφόρμα WiiFit λειτουργεί όπως ένα χειριστήριο, δηλαδή ανάλογα με το σημείο που ο παίχτης θα ρίξει το βάρος του, αυτή θα ερμηνεύσει την κίνηση σε μία κατεύθυνση (μπροστά, δεξιά, πίσω, αριστερά). Το WiiFit με την σειρά του στέλνει σειριακά τα δεδομένα στο Arduino το οποίο ελέγχει τους σερβοκινητήρες και έτσι αλλάζει στάση ο λαβύρινθος. Η ιδέα για την κατασκευή του παιχνιδιού ανήκε στον Jestin Stoffel. Το WiiFit μπορεί να αντικατασταθεί με διαφορετικά τύπου χειριστήρια χωρίς ιδιαίτερη δυσκολία.

Εφαρμογές του Arduino στις τέχνες (ζωγραφική)

Senseless drawing bot

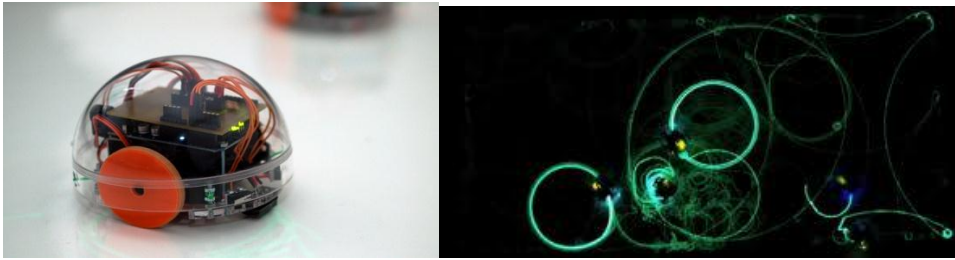
Το senseless drawing bot είναι μία κατασκευή η οποία ζωγραφίζει τυχαίες καμπύλες στους τοίχους. Όπως φαίνεται και στην εικόνα 22 η συσκευή αυτή είναι ένα μηχανοκίνητο όχημα το οποίο κινείται παράλληλα σε έναν τοίχο. Επάνω στο όχημα έχει τοποθετηθεί ένα διπλό εκκρεμές και στην άκρη του ένα σπρέι χρώματος. Κατά την κίνηση του οχήματος το εκκρεμές αρχίζει να ταλαντώνεται δεξιά και αριστερά ενώ παράλληλα ελέγχεται από έναν αισθητήρα για το εάν ξεπερνάει ένα κατώτατο και ένα ανώτατο όριο. Έτσι για παράδειγμα στην περίπτωση που η ταλάντωση του εκκρεμές είναι μικρότερη από την επιθυμητή τότε το όχημα αρχίζει και κινείται ρυθμικά αριστερά και δεξιά ώστε να αυξήσει την ταλάντωση του εκκρεμές. Αντίθετα όταν η ταλάντωση είναι μεγαλύτερη από την επιτρεπτή τότε το όχημα σταματάει. Όταν το εκκρεμές έχει την σωστή ταλάντωση, τότε με ένα σύστημα αυτοματισμού το σπρέι αρχίζει να ζωγραφίζει χρωματιστές καμπύλες σε όλο το μήκος του τοίχου. Αντικαθιστώντας το σπρέι με διάφορα χρώματα μπορεί να δημιουργηθεί ένα πρωτότυπο έργο τέχνης. Η κατασκευή αυτή προσομοιώνει την δημιουργία ενός γκράφιτι σε τοίχο απελευθερώνοντας τον δυναμισμό του σχεδιασμού του γκράφιτι καταργώντας παράλληλα τον ανθρώπινο παράγοντα.



Εικόνα 2.9: Senseless drawing bot

LumiBots

Έναν αξιοθαύμαστο και διαφορετικό τρόπο σχεδίασης σκέφτηκε και κατασκεύασε ο δημιουργός του LumiBots. Ο Mey Lean Kronemann πάνω σε μία φωσφορίζουσα επιφάνεια έβαλε να κινούνται 9 αυτόνομα robot τα οποία στο κάτω μέρος τους εκπέμπουν προς την φωσφορίζουσα επιφάνεια υπεριώδες φως. Έτσι καθώς τα ρομπότ κινούνται στο σκοτάδι, αφήνουν πίσω τους να φωσφορίζουν τα σημεία απ' όπου πέρασαν. Το κάθε ένα LumiBots (ρομπότ) σχεδιάστηκε με σκοπό τα υλικά κατασκευής του να έχουν όσον το δυνατόν φθηνότερο κοστολόγιο. Αποτελείται από ένα ελεγκτή Arduino, δύο αισθητήρες φωτός, δύο διακόπτες οι οποίοι ανιχνεύουν την επαφή του ρομπότ με τα τοιχώματα ή με ένα άλλο ρομπότ και τέλος ένα led που εκπέμπει UV φως και κάνει την επιφάνεια στην οποία κινείται να φωσφορίζει. Οι κινήσεις τους δεν είναι ούτε προγραμματισμένες ούτε προβλέψιμες, απλά βασίζονται σε δύο απλούς κανόνες. Ο πρώτος είναι να ακολουθούν μία φωσφορίζουσα γραμμή που υπάρχει στην επιφάνεια πάνω στην οποία κινούνται (όσο πιο φωτεινή η γραμμή τόσο καλύτερα). Ο δεύτερος κανόνας αναγκάζει το όχημα να αλλάζει πορεία όταν αυτό έρχεται σε επαφή με ένα άλλο όχημα ή με τα τοιχώματα που υπάρχουν με σκοπό να περιορίσει την κίνηση του LumiBot μέσα στην φωσφορίζουσα επιφάνεια. Όταν το LumiBot δεν βρίσκει να ακολουθήσει φωσφορίζουσες γραμμές τότε αυτό από μόνο του κινείται κυκλικά σύμφωνα με μία ακτίνα που του έχει δοθεί. Όπως φαίνεται και στην παρακάτω εικόνα τα σχήματα που δημιουργούνται είναι εκπληκτικά.



Εικόνα 2.10: LumiBots

Εφαρμογές του Arduino στην ρομποτική

Η μεγάλη ποικιλία των συμβατών αισθητήρων για Arduino που διατίθενται στο εμπόριο κάνουν τον σχεδιασμό ρομποτικών συστημάτων πολύ ενδιαφέρον. Ενώ παράλληλα η ασύρματη επικοινωνία του Arduino με έναν υπολογιστή μπορεί να αυξήσει τις δυνατότητες των ρομπότ σε πολύ μεγάλο βαθμό.

Mind-control

Μία από τις πιο εντυπωσιακές ρομποτικές κατασκευές είναι και αυτή η οποία παρουσιάζεται στο βιβλίο «Make a Mind-Controlled Arduino Robot» της oreilly. Στο βιβλίο αυτό περιγράφεται πως κατασκευάζεται ένα ρομπότ το οποίο θα ανταποκρίνεται σύμφωνα με την επιθυμία του ανθρώπου χωρίς να χρειαστεί να πατήσει κάποιο πλήκτρο. Πιο συγκεκριμένα εξηγείται ο τρόπος για την κατασκευή ενός ρομπότ το οποίο θα παίζει ήχους, θα αναβοσβήνει led και θα αντιδρά στα σήματα που θα δέχεται από το mind wave. Το mind wave είναι μία συσκευή η οποία τοποθετείται στο κεφάλι και ανιχνεύει την ηλεκτρική δραστηριότητα του ανθρώπινου εγκεφάλου. Η συσκευή αυτή μπορεί να ερμηνεύσει συγκεκριμένες ενέργειες και να τις μετατρέψει σε εντολές οι οποίες στην συνέχεια μπορούν να μεταβιβαστούν σειριακά στο Arduino. Ο τρόπος λειτουργίας του mind wave είναι ακριβώς ο ίδιος με αυτόν του εγκεφαλογράφηματος.



2.3 ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ ΜΙΚΡΟΕΛΕΓΚΤΗ ARDUINO

2.3.1 HARDWARE

Μια πλακέτα Arduino αποτελείται από ένα μικροελεγκτή Atmel AVR 8-bit, με συμπληρωματικά στοιχεία για τη διευκόλυνση του χρήστη στον προγραμματισμό και την ενσωμάτωση σε άλλα κυκλώματα. Μια σημαντική πτυχή του Arduino είναι ο συνήθης τρόπος που εκτίθενται οι σύνδεσμοι, επιτρέποντας στην πλακέτα CPU να συνδεθεί με μία ποικιλία εναλλάξιμων add-on modules γνωστές ως ασπίδες. Μερικές ασπίδες επικοινωνούν με την πλακέτα Arduino πάνω ακριβώς από διάφορες καρφίτσες, αλλά πολλές ασπίδες είναι μεμονωμένα διευθετημένες μέσω ενός I²C Serial Bus, επιτρέποντας σε πολλές ασπίδες να στοιβάζονται και να χρησιμοποιούνται παράλληλα. Το αυθεντικό Arduino έχει χρησιμοποιήσει τη σειρά megaAVR των τσιπ, και συγκεκριμένα το ATmega8, ATmega168, ATmega328, ATmega1280 και ATmega2560. Πολλοί από τους επεξεργαστές έχουν χρησιμοποιηθεί από το Arduino συμβατά. Οι περισσότεροι πίνακες περιλαμβάνουν 5 volt γραμμικού ρυθμιστή και 16MHz ταλαντωτή κρυστάλλου (ή κεραμικό αντηχείο σε ορισμένες παραλλαγές), αν και ορισμένα σχέδια, όπως το LilyPad τρέχουν στα 8 MHz και απονέμουν με τον ενσωματωμένο ρυθμιστή τάσης λόγω της ειδικής μορφής-παράγοντα περιορισμού. Ο μικροελεγκτής Arduino είναι επίσης προγραμματισμένος με έναν φορτωτή εκκίνησης που απλοποιεί την μεταφορά των προγραμμάτων για την on-chip μνήμη flash, σε σύγκριση με άλλες συσκευές που συνήθως χρειάζονται έναν εξωτερικό προγραμματιστή.

Σε εννοιολογικό επίπεδο, όταν χρησιμοποιούν τη στοίβα λογισμικού Arduino, όλες οι κάρτες έχουν προγραμματιστεί πάνω από μια RS-232 σειριακή σύνδεση, αλλά ο τρόπος που αυτό υλοποιείται ποικίλλει ανάλογα με την έκδοση εξοπλισμού. Μια σειρά από πλακέτες Arduino περιέχουν ένα απλό κύκλωμα επιλογέα για τη μετατροπή μεταξύ επιπέδου RS-232 και επιπέδου TTL σημάτων. (Οι τρέχουσες πλακέτες Arduino έχουν προγραμματιστεί μέσω USB, με τη χρήση USB σε σειριακό τσιπ προσαρμογέα όπως το FTDI FT232). Μερικές παραλλαγές, όπως το Arduino

Mini και το ανεπίσημο Boarduino, χρησιμοποιούν ένα αποσπώμενο USB σε πλακέτα σειριακού μετασχηματιστή ή καλώδιο, Bluetooth ή άλλες μεθόδους.

Η πλακέτα Arduino εκθέτει περισσότερες πινέζες I/O του μικροελεγκτή για χρήση από άλλα κυκλώματα. Η Diecimila, Duemilanove, και την τρέχουσα Uno προσφέρει 14 ψηφιακές καρφίτσες I / O, έξι από τις οποίες μπορεί να παράγει με διαμόρφωση εύρους παλμού σήματα, και έξι αναλογικές εισόδους. Αυτοί οι πείροι είναι στην κορυφή του πίνακα, μέσω θηλυκό 0,1 ιντσών (2,5 mm) επικεφαλίδες. Αρκετές plug-in ασπίδες εφαρμογής είναι επίσης εμπορικά διαθέσιμες.

Το Arduino Nano, και το Arduino-συμβατό με τη Bare Bones πλακέτα και τη Boarduino πλακέτα μπορούν να παρέχουν αρσενικές πινέζες κεφαλίδα στο κάτω μέρος του πίνακα και να συνδεθεί σε solderless breadboards.

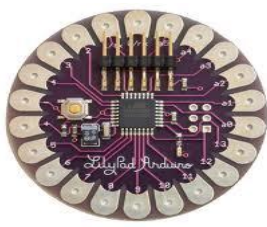
Υπάρχουν πάρα πολλοί Arduino-συμβατές και Arduino που προέρχονται από πίνακες. Μερικά είναι λειτουργικά ισοδύναμα με ένα Arduino και μπορεί να χρησιμοποιηθούν εναλλακτικά. Πολλοί είναι το βασικό Arduino με την προσθήκη κοινός οδηγούς εξόδου, συχνά για χρήση σε εκπαίδευση-επίπεδο για να απλοποιήσει την κατασκευή των μικρών ρομπότ. Άλλοι είναι ηλεκτρικά ισοδύναμοι αλλά να αλλάζουν τη μορφή παράγοντας, μερικές φορές επιτρέπουν τη συνέχιση της χρήσης του Shields, μερικές φορές όχι. Μερικές παραλλαγές χρησιμοποιούν τελείως διαφορετικούς επεξεργαστές, με διαφορετικά επίπεδα συμβατότητας.

2.3.2 Μοντέλα Arduino

Τα μοντέλα Arduino που κυκλοφορούν στην αγορά, παρουσιάζονται στον πίνακα . Το Arduino UNO επιλέχθηκε μιας και καλύπτει όλες τις ανάγκες της εργασίας καθώς τα pins που διαθέτει επαρκούν για όλες τις λειτουργίες.



Arduino UNO



Arduino LilyPad



Arduino Ethernet



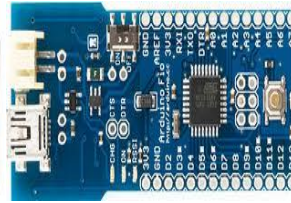
Arduino Leonard



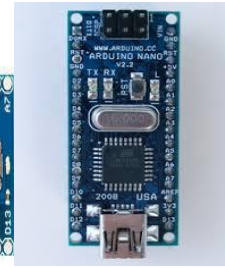
Arduino Mega 2560



Arduino Mega ADK



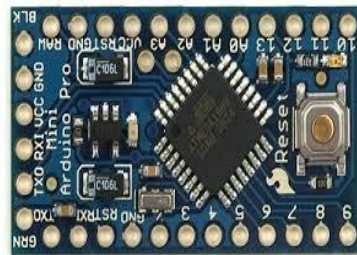
Arduino Fio



Arduino Nano



Arduino Pro



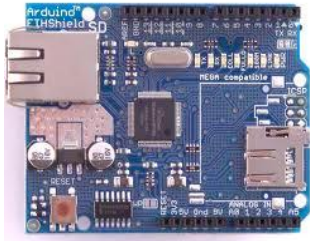
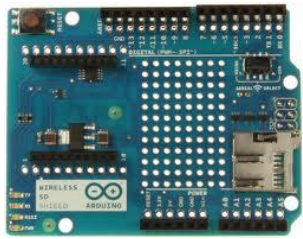
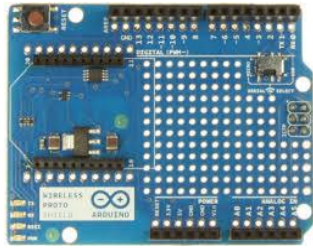
Arduino Pro Mini

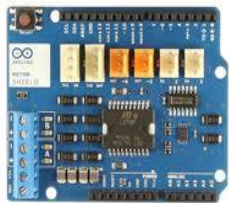
Πίνακας 1: Μοντέλα Arduino

2.3.3 Arduino Shields

Shields είναι τα εξαρτήματα που συνδέονται απευθείας με όλα τα pin του Arduino. Μερικά από αυτά φαίνονται στον πίνακα

<p>Arduino Wi-Fi Shield</p>		<p>Ασύρματη σύνδεση του Arduino στο διαδίκτυο</p>
-----------------------------	---	---

<p>Arduino Ethernet Shield</p>		<p>Σύνδεση του Arduino στο διαδίκτυο με RJ45 καλώδιο</p>
<p>Wireless SD Shield</p>		<p>Ασύρματη επικοινωνία με ασύρματη μονάδα με πλακέτα Arduino. Η μονάδα μπορεί να επικοινωνήσει έως και 100 πόδια σε εσωτερικούς χώρους ή ως 300 πόδια σε εξωτερικούς χώρους. Η μονάδα περιλαμβάνει μια θύρα υποδοχής SD</p>
<p>Wireless Proto Shield</p>		<p>Επιτρέπει στο Arduino να επικοινωνεί ασύρματα με μια ασύρματη μονάδα. Η μονάδα μπορεί να επικοινωνήσει έως και 100 πόδια σε εσωτερικούς χώρους ή σε εξωτερικούς χώρους ως 300 πόδια. Η μονάδα δεν περιλαμβάνει θύρα υποδοχής SD</p>

Arduino Motor Shield		Επιτρέπει την οδήγηση δύο DC κινητήρων από την ίδια συσκευή, ελέγχοντας την ταχύτητα και την κατεύθυνση του καθενός ξεχωριστά.
----------------------	---	--

Πίνακας 2: Arduino Shields, λεπτομέρειες

2.3.4 Τα χαρακτηριστικά εν συντομία:

Ο πίνακας αναλύει τα χαρακτηριστικά του μοντέλου Arduino UNO

Μικροελεγκτής	ATmega328
Τάση λειτουργίας	5V
Τάση εισόδου	7-12 V
Όρια τάσης	6-20V
Ψηφιακοί ακροδέκτες I/O	14 (εκ των οποίων 6 παρέχουν PWM έξοδο)
Ψηφιακοί ακροδέκτες εισόδου	6
DC ρεύμα ανά I/O ακροδέκτη	40 mA
DC ρεύμα για 3,3V ακροδέκτη	50 mA
Μνήμη Flash	32KB
SRAM	2KB
EEPROM	1KB
Ταχύτητα ρολογιού	16MHz

Πίνακας 3: χαρακτηριστικά εν συντομία

- 2Kb μνήμης SRAM είναι η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν τα προγράμματα για να αποθηκεύουν μεταβλητές, πίνακες κ.λπ. κατά το runtime. Όπως και σε έναν υπολογιστή, αυτή η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο Arduino σταματήσει ή αν γίνει reset.
- 1Kb μνήμης EEPROM η οποία μπορεί να χρησιμοποιηθεί για εγγραφή/ανάγνωση δεδομένων (χωρίς data type) ανά byte από τα προγράμματα κατά το runtime. Σε αντίθεση με την SRAM, η EEPROM δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset οπότε είναι το ανάλογο του σκληρού δίσκου.
- 32Kb μνήμης Flash, από τα οποία τα 2Kb χρησιμοποιούνται από το firmware του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware αυτό που στην ορολογία του Arduino ονομάζεται bootloader είναι αναγκαίο για την εγκατάσταση των προγραμμάτων που επιθυμούμε στον μικροελεγκτή μέσω της θύρας USB, χωρίς δηλαδή να χρειάζεται εξωτερικός hardware programmer. Τα υπόλοιπα 30Kb της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα μεταγλωττιστούν στον υπολογιστή μας. Η μνήμη Flash, όπως και η EEPROM δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset. Επίσης, ενώ η μνήμη Flash υπό κανονικές συνθήκες δεν προορίζεται για χρήση runtime μέσα από τα προγράμματα, λόγω της μικρής συνολικής μνήμης που είναι διαθέσιμη σε αυτά (2Kb SRAM + 1Kb EEPROM), έχει σχεδιαστεί μια βιβλιοθήκη που επιτρέπει την χρήση όσου χώρου περισσεύει (30Kb μείον το μέγεθος του προγράμματός μας σε μεταγλωττισμένη μορφή).

▪ Οι ακροδέκτες του Arduino

Καταρχήν το Arduino διαθέτει σειριακό interface. Ο μικροελεγκτής ATmega υποστηρίζει σειριακή επικοινωνία, την οποία το Arduino προωθεί μέσα από έναν ελεγκτή Serial-over-USB ώστε να συνδέεται με τον υπολογιστή μέσω USB. Η σύνδεση αυτή χρησιμοποιείται για την μεταφορά των προγραμμάτων που σχεδιάζονται από τον υπολογιστή στο Arduino αλλά και για αμφίδρομη επικοινωνία του Arduino με τον υπολογιστή μέσα από το πρόγραμμα την ώρα που εκτελείται.

Στην πάνω πλευρά του Arduino βρίσκονται 14 θηλυκά pin, αριθμημένα από 0 ως 13, που μπορούν να λειτουργήσουν ως ψηφιακές είσοδοι και έξοδοι. Λειτουργούν στα 5V και καθένα μπορεί να παρέχει ή να δεχτεί το πολύ 40mA. Ως ψηφιακή έξοδος, ένα από αυτά τα pin μπορεί να τεθεί από το πρόγραμμά μας σε κατάσταση HIGH ή LOW, οπότε το Arduino θα ξέρει αν πρέπει να διοχετεύσει ή όχι ρεύμα στο συγκεκριμένο pin. Με αυτόν τον τρόπο μπορούμε για παράδειγμα να ανάψουμε και να σβήσουμε ένα LED που έχουμε συνδέσει στο συγκεκριμένο pin. Αν πάλι ρυθμίσουμε ένα από αυτά τα pin ως ψηφιακή είσοδο μέσα από το πρόγραμμα, μπορείτε με την κατάλληλη εντολή να διαβάσετε την κατάστασή του (HIGH ή LOW) ανάλογα με το αν η εξωτερική συσκευή που έχουμε συνδέσει σε αυτό το pin διοχετεύει ή όχι ρεύμα στο pin (με αυτόν τον τρόπο «διαβάζετε» η κατάσταση ενός διακόπτη).

Μερικά από αυτά τα 14 pin, εκτός από ψηφιακές είσοδοι/έξοδοι έχουν και δεύτερη λειτουργία. Συγκεκριμένα:

- Τα pin 0 και 1 λειτουργούν ως RX και TX της σειριακής όταν το πρόγραμμά μας ενεργοποιεί την σειριακή θύρα. Έτσι, όταν το πρόγραμμά μας στέλνει δεδομένα στην σειριακή, αυτά προωθούνται και στην θύρα USB μέσω του ελεγκτή Serial-Over-USB αλλά και στο pin 0, για να τα διαβάσει ενδεχομένως μια άλλη συσκευή (π.χ. ένα δεύτερο Arduino στο δικό του pin 1). Αυτό φυσικά σημαίνει ότι αν στο πρόγραμμά μας ενεργοποιήσουμε το σειριακό interface, χάνουμε 2 ψηφιακές εισόδους/εξόδους.
- Τα pin 2 και 3 λειτουργούν και ως εξωτερικά interrupt (interrupt 0 και 1 αντίστοιχα). Με άλλα λόγια, μπορούμε να τα ρυθμίσουμε μέσα από το πρόγραμμά μας ώστε να λειτουργούν αποκλειστικά ως ψηφιακές είσοδοι στις οποίες όταν συμβαίνουν συγκεκριμένες αλλαγές, η κανονική ροή του προγράμματος σταματάει άμεσα και εκτελείται μια συγκεκριμένη συνάρτηση. Τα εξωτερικά interrupt είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας.
- Τα pin 3, 5, 6, 9, 10 και 11 μπορούν να λειτουργήσουν και ως ψευδοαναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation), δηλαδή το ίδιο σύστημα που διαθέτουν οι μητρικές των υπολογιστών για να ελέγχουν τις ταχύτητες των ανεμιστήρων. Έτσι, μπορούμε να συνδέσουμε για παράδειγμα ένα LED σε κάποιο από αυτά τα pin και να ελέγξουμε πλήρως την φωτεινότητά του με ανάλυση 8bit (256 καταστάσεις από 0-

σβηστό ως 255-πλήρως αναμμένο) αντί να έχουμε απλά την δυνατότητα αναμμένο-σβηστό που παρέχουν οι υπόλοιπες ψηφιακές έξοδοι. Είναι σημαντικό να καταλάβουμε ότι το PWM δεν είναι πραγματικά αναλογικό σύστημα και ότι θέτοντας στην έξοδο την τιμή 127, δεν σημαίνει ότι η έξοδος θα δίνει 2.5V αντί της κανονικής τιμής των 5V, αλλά ότι θα δίνει ένα παλμό που θα εναλλάσσεται με μεγάλη συχνότητα και για ίσους χρόνους μεταξύ των τιμών 0 και 5V.

Στην κάτω πλευρά του Arduino, με τη σήμανση ANALOG IN, θα βρούμε μια ακόμη σειρά από 6 pin, αριθμημένα από το 0 ως το 5. Το καθένα από αυτά λειτουργεί ως αναλογική είσοδος κάνοντας χρήση του ADC (Analog to Digital Converter) που είναι ενσωματωμένο στον μικροελεγκτή. Για παράδειγμα, μπορούμε να τροφοδοτήσουμε ένα από αυτά με μια τάση την οποία μπορούμε να κυμάνουμε με ένα ποτενσιόμετρο από 0V ως μια τάση αναφοράς Vref η οποία, αν δεν κάνουμε κάποια αλλαγή είναι προ-ρυθμισμένη στα 5V. Τότε, μέσα από το πρόγραμμά μας μπορούμε να «διαβάσουμε» την τιμή του pin ως ένα ακέραιο αριθμό ανάλυσης 10-bit, από 0 (όταν η τάση στο pin είναι 0V) μέχρι 1023 (όταν η τάση στο pin είναι 5V). Η τάση αναφοράς μπορεί να ρυθμιστεί με μια εντολή στο 1.1V, ή σε όποια τάση επιθυμούμε(μεταξύ 2 και 5V) τροφοδοτώντας εξωτερικά με αυτή την τάση το pin με την σήμανση AREF που βρίσκεται στην απέναντι πλευρά της πλακέτας. Έτσι, αν τροφοδοτήσουμε το pin AREF με 3.3V και στην συνέχεια δοκιμάσετε να διαβάσουμε κάποιο pin αναλογικής εισόδου στο οποίο εφαρμόζετε τάση 1.65V, το Arduino θα μας επιστρέψει την τιμή 512. Τέλος, καθένα από τα 6 αυτά pin, με κατάλληλη εντολή μέσα από το πρόγραμμα μπορεί να μετατραπεί σε ψηφιακό pin εισόδου/εξόδου όπως τα 14 που βρίσκονται στην απέναντι πλευρά και τα οποία περιγράφηκαν πριν. Σε αυτή την περίπτωση τα pin μετονομάζονται από 0~5 σε 14~19 αντίστοιχα.

▪ Τροφοδοσία

Το αναπτυξιακό Arduino Duemilanove τροφοδοτείται είτε από εξωτερική τροφοδοσία είτε απευθείας από την θύρα USB. Η επιλογή της πηγής γίνεται αυτόματα από το αναπτυξιακό. Ως εξωτερική τροφοδοσία ορίζεται είτε μια μπαταρία, είτε ένας μετασχηματιστής των 9Volt από 220V. Η μπαταρία μπορεί να συνδεθεί στις υποδοχές του Arduino Vin και GND όπου τοποθετούνται ο θετικός πόλος και ο αρνητικός αντίστοιχα. Από την άλλη αν τροφοδοτήσουμε με μετασχηματιστή απλά τοποθετούμε το βύσμα στην υποδοχή που υπάρχει με τον θετικό πόλο στο κέντρο.

Η πλακέτα μπορεί να λειτουργήσει με εξωτερική πηγή από 6 έως 20 Volts. Αν ωστόσο τροφοδοτηθεί με λιγότερα από 7 Volt τα pin εξόδου 5Volt δεν θα

καταφέρουν να εξάγουν τάση 5 Volts. Αν από την άλλη δώσουμε πάνω από 12 Volts θα υπερθερμανθεί ο σταθεροποιητής τάσης στην πλακέτα και ενδεχομένως να καταστραφεί. Συνεπώς μια ιδανική τάση είναι τα 9 Volts.

Οι ακροδέκτες τροφοδοσίας είναι οι εξής:

- **VIN.** Ακροδέκτης για μη σταθεροποιημένη τάση. Συνήθως εδώ συνδέεται μια εξωτερική πηγή τροφοδοσίας.
- **5V.** Ακροδέκτης σταθεροποιημένης τάσης 5Volt. Χρησιμοποιείται για την τροφοδοσία του μικροελεγκτή ή άλλων ηλεκτρονικών στοιχείων.
- **3.3V** Το ολοκληρωμένο FTDI που βρίσκεται στην πλακέτα του Arduino παράγει τάση των 3.3V με μέγιστο ρεύμα 50mA.
- **GND.** Ακροδέκτες Γείωσης

Μνήμη

Ο μικροεπεξεργαστής ATmega328 έχει τρεις ομάδες μνήμης. Διαθέτει flash memory, στην οποία αποθηκεύονται τα Arduino sketch, SRAM (static random access memory), στην οποία δημιουργείται το sketch και χρησιμοποιεί τις μεταβλητές όταν τρέχει, και EEPROM, η οποία χρησιμοποιείται από τους προγραμματιστές για την αποθήκευση μακροχρόνιων πληροφοριών. 2KB μνήμης **SRAM**: Η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν τα προγράμματα για να αποθηκεύουν μεταβλητές, πίνακες κ.λπ. Η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο Arduino σταματήσει ή πατηθεί το κουμπί επανεκκίνησης. 1KB μνήμης

EEPROM: Μπορεί να χρησιμοποιηθεί για εγγραφή ή ανάγνωση δεδομένων από τα προγράμματα. Σε αντίθεση με την SRAM, δε χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή επανεκκίνησης.

32KB μνήμης Flash: 2 KB χρησιμοποιούνται από το firmware του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware είναι αναγκαίο για την εγκατάσταση προγραμμάτων στο μικροελεγκτή μέσω της θύρας USB. Τα υπόλοιπα 30KB της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash, δε χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή επανεκκίνησης.

▪ Ενσωματωμένα κουμπιά και LED

Πάνω στην πλακέτα του Arduino υπάρχει ένας διακόπτης micro-switch και 4 μικροσκοπικά LED επιφανειακής στήριξης. Η λειτουργία του διακόπτη (που έχει την σήμανση RESET) και του ενός LED με την σήμανση POWER είναι μάλλον προφανής. Τα δύο LED με τις σημάνσεις TX και RX, χρησιμοποιούνται ως ένδειξη λειτουργίας του σειριακού interface, καθώς ανάβουν όταν το Arduino στέλνει ή λαμβάνει

(αντίστοιχα) δεδομένα μέσω USB. Σημειώνουμε ότι τα LED αυτά ελέγχονται από τον ελεγκτή Serial-over-USB και συνεπώς δεν λειτουργούν όταν η σειριακή επικοινωνία γίνεται αποκλειστικά μέσω των ψηφιακών pin 0 και 1. Τέλος, υπάρχει το LED με την σήμανση L. Η βασική δοκιμή λειτουργίας του Arduino είναι να του αναθέσουμε να αναβοσβήνει ένα LED. Για να μπορούμε να το κάνουμε αυτό από την πρώτη στιγμή, χωρίς να συνδέσουμε τίποτα πάνω στο Arduino, οι κατασκευαστές του σκέφτηκαν να ενσωματώσουν ένα LED στην πλακέτα, όπου το σύνδεσαν στο ψηφιακό pin 13. Έτσι, ακόμα και αν δεν έχουμε συνδέσει τίποτα πάνω στο φυσικό pin 13, αναθέτοντάς του την τιμή HIGH μέσα από το πρόγραμμά μας, θα ανάψει αυτό το ενσωματωμένο LED.

2.3.5 SOFTWARE

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) του Arduino είναι μια εφαρμογή γραμμένη σε Java, που λειτουργεί σε πολλές πλατφόρμες, και προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing και το σχέδιο Wiring. Έχει σχεδιαστεί για να εισαγάγει τον προγραμματισμό στους καλλιτέχνες και τους νέους που δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως είναι η επισήμανση σύνταξης, ο συνδυασμός αγκύλων και η αυτόματη εσοχή, καθώς επίσης είναι σε θέση να μεταγλωττίζει και να φορτώνει προγράμματα στην πλακέτα με ένα μόνο κλικ. Συνήθως δεν υπάρχει καμία ανάγκη να επεξεργαστούμε `make files` ή να τρέξουμε προγράμματα σε ένα περιβάλλον γραμμής εντολών. Με τον όρο «σκίτσο» αναφερόμαστε σε ένα πρόγραμμα ή κώδικα που γράφτηκε για Arduino.

Τα Arduino προγράμματα είναι γραμμένα σε C ή C++. Το Arduino IDE έρχεται με μια βιβλιοθήκη λογισμικού που ονομάζεται "Wiring" από το αρχικό σχέδιο Wiring, γεγονός που καθιστά πολλές κοινές λειτουργίες εισόδου / εξόδου πολύ πιο εύκολες. Οι χρήστες πρέπει μόνο να ορίσουν δύο λειτουργίες για να κάνουν ένα πρόγραμμα κυκλικής εκτέλεσης:

`setup ()`: μια συνάρτηση εκτελείται μία φορά κατά την έναρξη ενός προγράμματος η οποία μπορεί να αρχικοποιεί τις ρυθμίσεις.

`loop ()`: μια συνάρτηση η οποία καλείται συνέχεια μέχρι η πλακέτα να απενεργοποιηθεί.

Ένα τυπικό πρώτο πρόγραμμα για έναν μικροελεγκτή είναι απλά όταν αναβοσβήνει ένα LED. Στο περιβάλλον Arduino, ο χρήστης μπορεί να γράψει ένα πρόγραμμα σαν αυτό:

```
#define LED_PIN 13

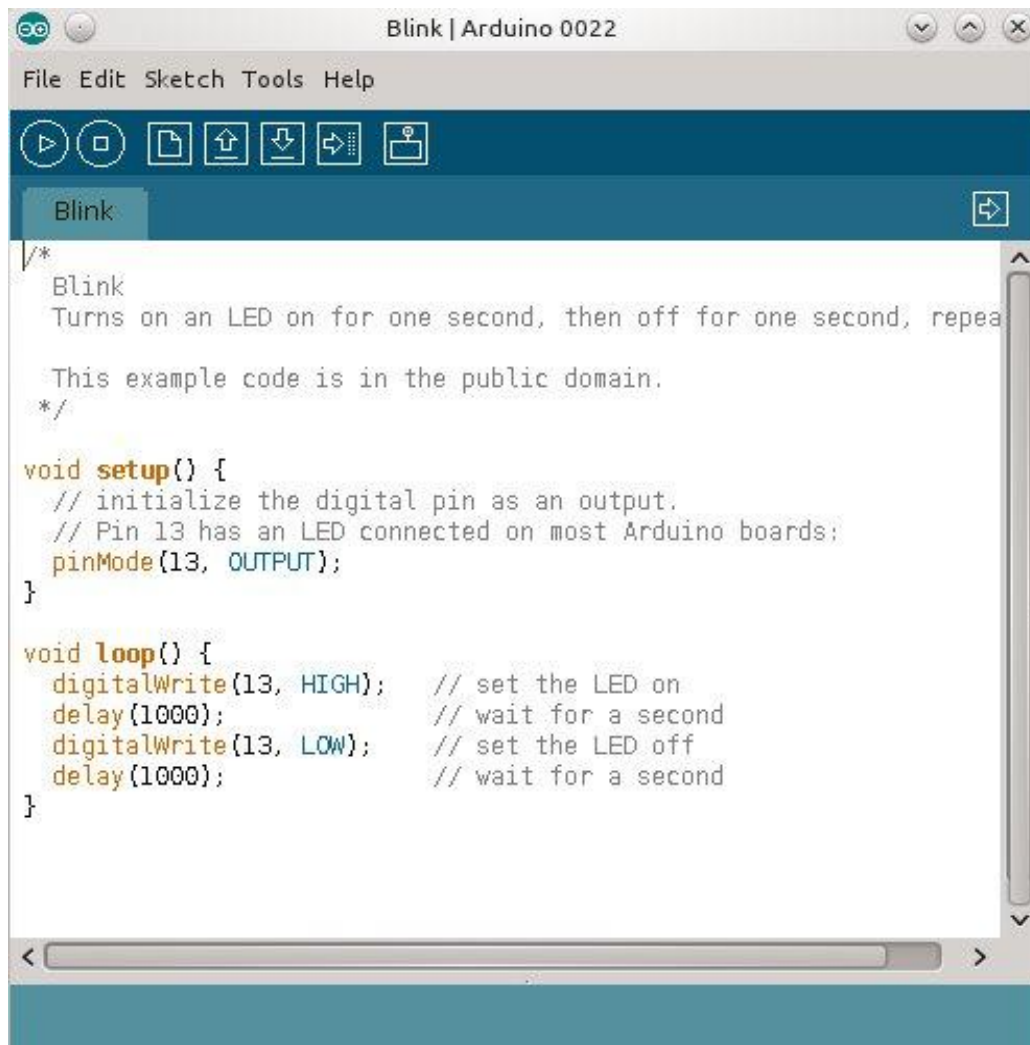
void setup () {
  pinMode (LED_PIN, OUTPUT); // enable pin 13 for digital output
}

void loop () {
  digitalWrite (LED_PIN, HIGH); // turn on the LED
  delay (1000); // wait one second (1000 milliseconds)
  digitalWrite (LED_PIN, LOW); // turn off the LED
  delay (1000); // wait one second
}
```

Είναι ένα χαρακτηριστικό των περισσότερων πλακετών Arduino ότι έχουν ένα LED και μια αντίσταση φορτίου που συνδέονται μεταξύ του pin 13 και του εδάφους. Ο προηγούμενος κωδικός δεν θα μπορέσει να αναγνωριστεί από έναν κανονικό μεταγλωττιστή C++ ως έγκυρο πρόγραμμα, έτσι ώστε όταν ο χρήστης κάνει κλικ στο κουμπί "Upload to I/O board" στο IDE, ένα αντίγραφο του κώδικα θα γραφτεί σε ένα προσωρινό αρχείο με ένα επιπλέον include στην κορυφή και μια πολύ απλή συνάρτηση main() στο κάτω μέρος, για να φτιάξει ένα έγκυρο C++ πρόγραμμα. Το IDE του Arduino χρησιμοποιεί το GNU toolchain και AVR libc για την μεταγλώττιση των προγραμμάτων και το avrdude για την φόρτωση προγραμμάτων στην πλακέτα. Δεδομένου ότι η πλατφόρμα Arduino χρησιμοποιεί Atmel μικροελεγκτές, το περιβάλλον ανάπτυξης της Atmel, AVR Studio ή η νεότερη έκδοση του Atmel Studio, μπορούν επίσης να χρησιμοποιηθούν για την ανάπτυξη λογισμικού για το Arduino.








2.3.6 Το περιβάλλον ανάπτυξης

Το περιβάλλον ανάπτυξης (IDE) του Arduino είναι μία πολυπλατφορμική εφαρμογή γραμμένη σε Java και βασίζεται στο περιβάλλον της γλώσσας προγραμματισμού Processing.



Εικόνα 2.12: Το περιβάλλον ανάπτυξης του Arduino

Βασικές λειτουργίες του IDE:

	Έλεγχος του κώδικα για λάθη.
	Τερματισμός της σειριακής κονσόλας.
	Δημιουργία νέου έργου (sketch)
	Παρουσίαση μενού με όλα τα αποθηκευμένα έργα. Πατώντας σε ένα από αυτά ανοίγει για επεξεργασία.
	Αποθήκευση του έργου.
	Μεταγλώτιση του κώδικα και ανέβασμά του στο Arduino.
	Εμφάνιση της σειριακής κονσόλας. Αποστολή και λήψη δεδομένων που στάλθηκαν μέσω της σειριακής θύρας.

Εικόνα 2.13: Λειτουργίες του IDE

▪ Ρυθμίσεις του περιβάλλοντος ανάπτυξης

Οι βασικές ρυθμίσεις που πρέπει να κάνουμε από την στιγμή που ενώσουμε το Arduino στο σύστημά μας είναι:

1. Από το μενού Tools -> Board επιλέγουμε την πλακέτα που έχουμε
2. Από το μενού Tools -> Serial Port επιλέγουμε την σειριακή θύρα ή τη θύρα USB που έχουμε συνδεδεμένο το Arduino.

Οι ρυθμίσεις που αφορούν το μέγεθος του κειμένου, τον φάκελο αποθήκευσης και τη χρήση εξωτερικού κειμενογράφου βρίσκονται στη καρτέλα Preferences (File -> Preferences). Αν θέλουμε να πραγματοποιήσουμε περισσότερες ρυθμίσεις μπορούμε να κάνουμε αλλαγές το αρχείο preferences.txt (πχ στο Λειτουργικό Σύστημα του Linux βρίσκεται στον φάκελο του χρήστη ~/.Arduino/preferences.txt).

▪ Δομή προγράμματος

Η δομή ενός τυπικού προγράμματος του Arduino είναι η εξής:

// δηλώσεις μεταβλητών

void setup() {

// αρχικοποιήσεις

}

```
void loop() {  
  
  //...  
  
}
```

Υπάρχουν δυο βασικές συναρτήσεις σε ένα τυπικό πρόγραμμα.

Η συνάρτηση **setup()** όπου εκτελείται στην αρχή του προγράμματος και για μία μόνο φορά. Χρησιμοποιείται για τις αρχικοποιήσεις των μεταβλητών, τις δηλώσεις των pin (αν θα είναι είσοδος ή έξοδος) και τις αρχικοποιήσεις των βιβλιοθηκών.

Η συνάρτηση **loop()** κάνει όπου ο κώδικας που γράφεται μέσα στη συνάρτηση αυτή επαναλαμβάνεται συνεχώς δίνοντας την δυνατότητα στο πρόγραμμά μας να αλλάζει τιμές και το Arduino να ανταποκρίνεται ανάλογα.

▪ Μεταβλητές

Μεταβλητή στη γλώσσα προγραμματισμού ονομάζουμε ένα γλωσσικό αντικείμενο που μπορεί να λάβει διάφορες τιμές, μία κάθε φορά. Οι τιμές μιας μεταβλητής περιορίζονται συνήθως σε ένα τύπο δεδομένων.

Οι βασικοί τύποι δεδομένων στο Arduino είναι:

1. **byte**: αποθηκεύει μια αριθμητική τιμή 8-bit χωρίς δεκαδικά ψηφία, παίρνουν τιμές από 0 μέχρι 255.
2. **int**: ακραίοι, παίρνουν τιμές από -32,768 μέχρι 32767.
3. **long**: μεγάλου μεγέθους ακραίοι, παίρνουν τιμές από -2,147,483,648 μέχρι 2,147,483,647
4. **float**: πραγματικοί αριθμοί, παίρνουν τιμές από 3.4×10^{-38} μέχρι 3.4×10^{38}

Τις μεταβλητές μπορούμε να τις δηλώσουμε στην αρχή του προγράμματός μας:
int myvariable;

Μπορούμε επίσης να δώσουμε αρχική τιμή στη μεταβλητή ταυτόχρονα με τη δήλωσή της: **int myvariable = 47;**

▪ Σταθερές

Οι σταθερές είναι αντικείμενα τα οποία παίρνουν μόνο μία τιμή, και δηλώνονται μαζί με τις μεταβλητές: **#define ledPin 13**

▪ Πίνακες – Arrays

Πίνακα ονομάζουμε τη διάταξη δεδομένων μιας ή περισσότερων διαστάσεων η οποία είναι συγκεκριμένου τύπου δεδομένων. Για παράδειγμα αν έχουμε ένα πίνακα ακραίων 5 θέσεων τον οποίο ονομάζουμε **myarray** τον δηλώνουμε όπως βλέπουμε παρακάτω: **int myarray[5];**

Για να δώσουμε τιμή στο τέταρτο στοιχείο του πίνακα **myarray** γραφτούμε: **myarray[3] = 12;**

επίσης μπορούμε να γεμίσουμε τον πίνακα ταυτόχρονα με την δήλωση του:

int myarray[] = {12, 45, 32, 61, 55};

▪ Αριθμητικοί τελεστές

Οι αριθμητικοί τελεστές καλύπτουν τις βασικές πράξεις: πρόσθεση, αφαίρεση, πολλαπλασιασμό, διαίρεση (+, -, *, /). Για παράδειγμα μπορούμε να κάνουμε την πρόσθεση δύο ακέραιων και το αποτέλεσμα να εκχωρηθεί σε μία μεταβλητή:

sum = 458 + 954;

▪ Τελεστές σύγκρισης

Με τους τελεστές σύγκρισης μπορούμε να ελέγξουμε αν μία συγκεκριμένη συνθήκη μεταξύ μεταβλητών ή σταθερών είναι “Αληθής”. Ποιο συγκεκριμένα υπάρχουν οι παρακάτω τελεστές σύγκρισης:

x == y το x είναι ίσο με το y

x != y το x είναι άνισο του y

x < y το x είναι μικρότερο με το y

x > y το x είναι μεγαλύτερο με το y

x <= y το x είναι μικρότερο ή ίσο με το y

x >= y το x είναι μεγαλύτερο ή ίσο με το y

▪ Λογικοί τελεστές

Με τους λογικούς τελεστές μπορούμε να συγκρίνουμε δύο ή περισσότερες εκφράσεις, δίνοντας αποτέλεσμα “Αληθής” ή “Ψευδής”. Υπάρχουν τρεις λογικοί τελεστές:

Λογικό **KAI &&** - επιστρέφει “Αληθής” αν όλες οι εκφράσεις είναι “Αληθείς”

Λογικό **Ή ||** - επιστρέφει “Αληθής” αν μία από τις εκφράσεις είναι “Αληθείς”

Λογικό **ΟΧΙ !** - επιστρέφει “Αληθής” αν η έκφραση είναι “Ψευδής”

Παράδειγμα:

```
if(x > 0 && x < 5){
```

```
//κώδικας
```

```
}
```

Στο παραπάνω κομμάτι κώδικα γίνεται έλεγχος αν το x είναι μεγαλύτερο από το 0 ΚΑΙ μικρότερο από 5 τότε εκτελείται ο κώδικας που βρίσκεται μέσα στις αγκύλες. Με λίγα λόγια η πρόταση **If()** ελέγχει αν η συνθήκη μέσα στις παρενθέσεις είναι “Αληθής”.

Ένα άλλο παράδειγμα:

```
if(!x > 0){
```

```
//κώδικας
```

```
}
```

Εδώ γίνεται έλεγχος αν το x είναι μεγαλύτερο από 0, αν αυτή η συνθήκη ΔΕΝ ισχύει τότε έχουμε το αποτέλεσμα “Αληθής” και εκτελείται ο κώδικας μέσα στις αγκύλες.

▪ Ψηφιακή έξοδος

Το Arduino Diecimila αποτελείται από δεκατρία ψηφιακά pin, τα οποία μπορούμε να τα χρησιμοποιήσουμε το κάθε ένα ξεχωριστά, είτε για είσοδο είτε για έξοδο. Μπορούμε να τα προγραμματίσουμε να συμπεριφέρονται όπως εμείς θέλουμε, αρκεί να κάνουμε τις σωστές δηλώσεις στο κώδικα που θα φορτώσουμε στη πλακέτα. Η έξοδος του κάθε pin μπορεί να προγραμματιστεί να δίνει τιμές HIGH ή LOW. Λέγοντας HIGH ενόνοουμε το δυαδικό '1' και έχουμε τάση εξόδου 5V DC, ενώ το LOW είναι το δυαδικό '0' και έχει τάση εξόδου 0V DC (ground).

2.3.7 ARDUINO EXAMPLE

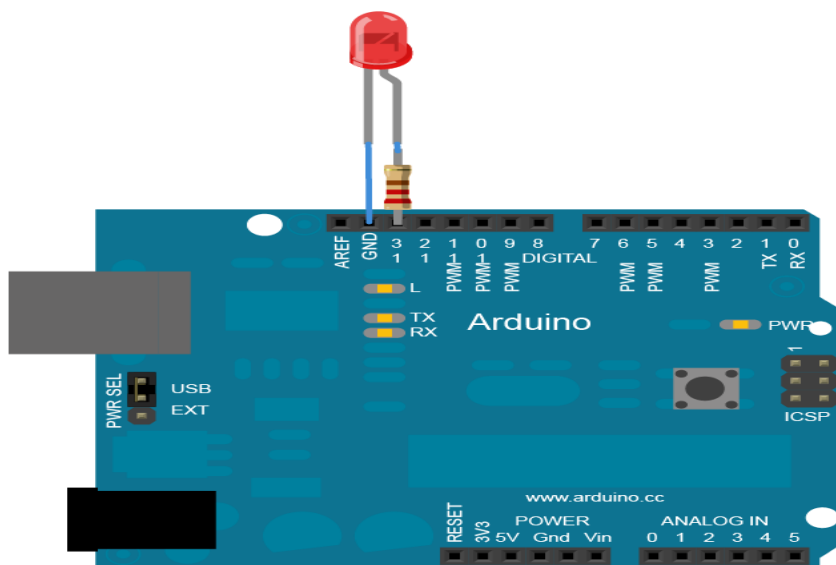
Παράδειγμα 1

Blink

Στις περισσότερες γλώσσες προγραμματισμού, το πρώτο πρόγραμμα που γράφετε τυπώνει το "hello world" στην οθόνη. Από μια πλακέτα Arduino που δεν έχει οθόνη, θα αναβοσβήνει ένα LED αντί αυτού.

Οι πίνακες έχουν σχεδιαστεί ώστε να είναι εύκολο να αναβοσβήνει ένα LED που χρησιμοποιεί την ψηφιακή pin 13. Μερικοί (όπως ο Diecimila και LilyPad) έχουν το LED ενσωματωμένο στην πλακέτα ενώ σε άλλες (όπως το Mini και BT), υπάρχει 1 KB αντίσταση στον πείρο, που μας επιτρέπει να συνδέσουμε ένα LED άμεσα. Για να συνδέσουμε ένα LED σε ένα άλλο ψηφιακό pin, θα πρέπει να χρησιμοποιήσουμε μια εξωτερική αντίσταση.

Τα LED έχουν πολικότητα, πράγμα που σημαίνει ότι θα ανάψει μόνο αν προσανατολίσει σωστά τα πόδια. Το μακρύ πόδι είναι τυπικά θετική και πρέπει να συνδεθεί με τον ακροδέκτη 13. Το κοντό πόδι συνδέεται με GND (η λάμπα του LED επίσης τυπικά θα έχει μια επίπεδη ακμή σε αυτή την πλευρά).



Εικόνα 2.14: Turn an LED on and off

CODE

```
int ledPin = 13; // LED συνδέεται στο ψηφιακό pin 13
```



```

void setup()

{

  pinMode(ledPin, OUTPUT); // ορισμός ψηφιακού pin ως έξοδος

}

void loop()

{

  digitalWrite(ledPin, HIGH); // όρισε το LED ανοιχτό

  delay(1000); // περίμενε για ένα λεπτό

  digitalWrite(ledPin, LOW); // όρισε το LED κλειστό

  delay(1000); // περίμενε ένα λεπτό

}

```

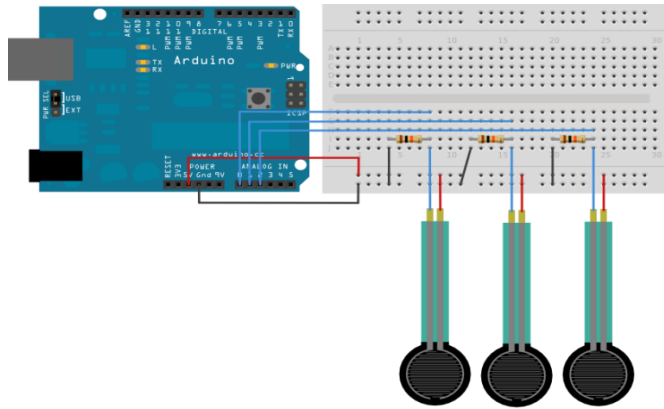
Παράδειγμα 2

Virtual Color Mixer

Επένδυση τεχνικής για την αποστολή πολλαπλών τιμών από την πλακέτα Arduino στον υπολογιστή. Σ εαυτή την περίπτωση οι αναγνώσεις από τρία ποτενσιόμετρα χρησιμοποιούνται για να ορίσουν το κόκκινο, το πράσινο και το μπλε συστατικό του χρωματιστού φόντου στο επεξεργασμένο σκίτσο.

ΚΥΚΛΩΜΑ

Τα ποτενσιόμετρα συνδέονται στα πηνία αναλογικών εισόδων 0,1 και 2



Εικόνα 2.15: virtual color mixer

CODE

```
int redPin = 0;

int greenPin = 1;

int bluePin = 2;

void setup()
{
  Serial. begin(9600);
}

void loop ()
{
  Serial. Print ("R");
  Serial.println (analogRead (redPin));

  Serial. Print ("G");
  Serial.println(analogRead(greenPin));

  Serial. print ("B");
  Serial.println(analogRead(bluePin));

  delay(100);
}
```

```
}
```

ΕΠΕΞΕΡΓΑΣΙΑ ΚΩΔΙΚΑ

```
/**
```

```
 * Color Mixer
```

```
*/
```

```
import processing. serial.*;
```

```
String buff = "";
```

```
int rval = 0, gval = 0, bval = 0;
```

```
int NEWLINE = 10;
```

```
Serial port;
```

```
void setup()
```

```
{
```

```
size(200, 200);
```

```
// Εκτυπώστε μια λίστα COM1 σε περίπτωση που δεν λειτουργεί έξω
```

```
println("Available serial ports:");
```

```
println(Serial.list());
```

```
//port = new Serial(this, "COM1", 9600);
```

```
// Χρησιμοποιείτε την πρώτη διαθέσιμη θύρα
```

```
port = new Serial(this, Serial.list()[0], 9600);
```

```
}
```

```
void draw()
```

```
{
```

```
while (port.available() > 0) {
```

```
serialEvent(port.read());
```

```
}
```

```
background(rval, gval, bval);
```

```

}

void serialEvent(int serial)

{
    // Όταν η μεταβλητή "serial" δεν είναι ίση με την τιμή για μια νέα γραμμή,
    προσθέστε την τιμή για τη μεταβλητή "buff". Εάν η τιμή "serial" είναι ίση με την τιμή
    για μια νέα γραμμή, σώστε την τιμή του buffer στη μεταβλητή "val".

    if(serial != NEWLINE) {

        buff += char(serial);

    } else {

        // ο πρώτος χαρακτήρας μας λέει ποιο χρώμα είναι αυτή η τιμή

        char c = buff.charAt(0);

        // μετακίνησέ το από το string

        buff = buff.substring(1);

        // Απορρίψτε τη μεταφορά μετ'επιστροφής στο τέλος του buffer

        buff = buff.substring(0, buff.length()-1);

        // Αναλύσει το String σε ακέραιο αριθμό

        if (c == 'R')

            rval = Integer.parseInt(buff);

        else if (c == 'G')

            gval = Integer.parseInt(buff);

        else if (c == 'B')

            bval = Integer.parseInt(buff);

        // Καταργήστε την τιμή του "buff"

        buff = "";

    }

}

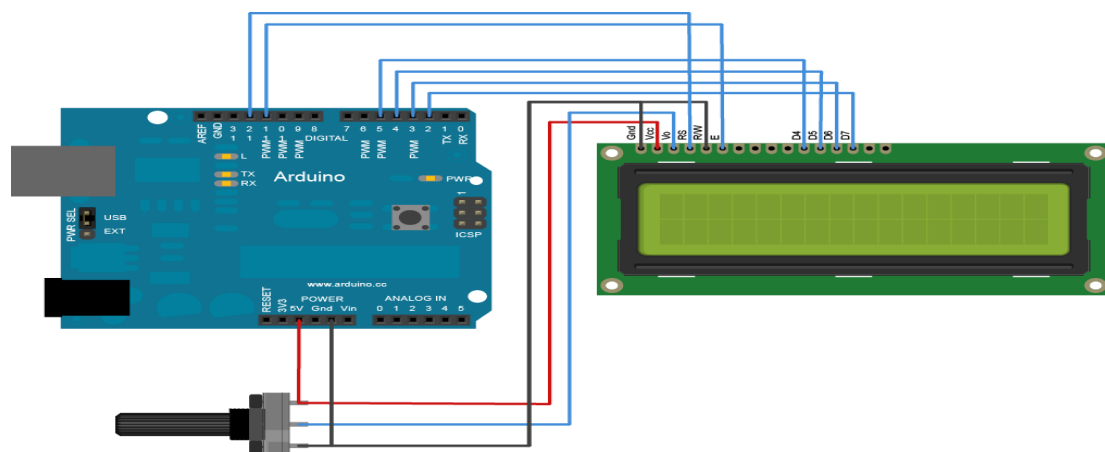
```

Παράδειγμα 3

LCD Display - 8 bits

Αυτό το παράδειγμα δείχνει την πιο βασική δράση που μπορεί να γίνει με μια οθόνη LCD: να δείξει ένα μήνυμα καλωσορίσματος. Στην περίπτωση μας έχουμε μια οθόνη LCD με οπίσθιο φωτισμό και τον έλεγχο της αντίθεσης. Ως εκ τούτου, θα χρησιμοποιήσουμε ένα ποτενσιόμετρο για τη ρύθμιση της αντίθεσης.

Οι οθόνες LCD είναι τις περισσότερες φορές καθοδηγούμενες χρησιμοποιώντας ένα βιομηχανικό πρότυπο που θεσπίστηκε από την Hitachi. Σύμφωνα με αυτό υπάρχει μια ομάδα των ακίδων είναι αφοσιωμένα στην αποστολή των δεδομένων και τις θέσεις των εν λόγω δεδομένων στην οθόνη, ο χρήστης μπορεί να επιλέξει να χρησιμοποιήσει 4 ή 8 καρφίτσες για να στείλει δεδομένα.



Εικόνα 2.16: LCD display

CODE

```
/* LCD Hola
```

* Αυτό είναι το πρώτο παράδειγμα για το πώς να χρησιμοποιήσουμε μια οθόνη LCD με μεταφορά δεδομένων πάνω από 8 bits. το παράδειγμα χρησιμοποιεί όλες τις ψηφιακές ακίδες στην πλακέτα Arduino, αλλά μπορεί εύκολα να εμφανίσετε τα δεδομένα στην οθόνη

*

* Υπάρχουν οι εξής πινέζες που πρέπει να εξεταστούν:

* - DI, RW, DB0..DB7, Enable (11 in total)

* το pinout για οθόνες LCD είναι στάνταρ και υπάρχει αφθονία της τεκμηρίωσης που πρέπει να βρεθεί στο διαδίκτυο

*/

```
int DI = 12;
```

```
int RW = 11;
```

```
int DB[] = {3, 4, 5, 6, 7, 8, 9, 10};
```

```
int Enable = 2;
```

```
void LcdCommandWrite(int value) {
```

```
    // καταγραφή όλων των ακίδων
```

```
    int i = 0;
```

```
    for (i=DB[0]; i <= DI; i++) {
```

```
        digitalWrite(i,value & 01);
```

```
        value >>= 1;
```

```
    }
```

```
    digitalWrite(Enable,LOW);
```

```
    delayMicroseconds(1);
```

```
    // στέλνουμε έναν παλμό για ενεργοποίηση
```

```
    digitalWrite(Enable,HIGH);
```

```
    delayMicroseconds(1); // παύση 1 ms
```

```
    digitalWrite(Enable,LOW);
```

```
    delayMicroseconds(1); // παύση 1 ms
```

```
}
```

```
void LcdDataWrite(int value) {
```

```
    // καταγραφή όλων των ακίδων
```

```

int i = 0;

digitalWrite(DI, HIGH);

digitalWrite(RW, LOW);

for (i=DB[0]; i <= DB[7]; i++) {

digitalWrite(i,value & 01);

value >>= 1;

}

digitalWrite(Enable,LOW);

delayMicroseconds(1);

// στέλνουμε έναν παλμό για ενεργοποίηση

digitalWrite(Enable,HIGH);

delayMicroseconds(1);

digitalWrite(Enable,LOW);

delayMicroseconds(1); // παύση 1 ms

}

void setup (void) {

int i = 0;

for (i=Enable; i <= DI; i++) {

pinMode(i,OUTPUT);

}

delay(100);

// προετοιμαστεί lcd μετά από μια σύντομη παύση

// απαιτείται από τον ελεγκτή οθόνες LCD

LcdCommandWrite(0x30); // function set:

// 8-bit interface, 1 display lines, 5x7 font

delay(64);

```

```

LcdCommandWrite(0x30); // function set:

// 8-bit interface, 1 display lines, 5x7 font
delay(50);

LcdCommandWrite(0x30); // κανονισμός λειτουργίας:

// 8-bit interface, 1 display lines, 5x7 font
delay(20);

LcdCommandWrite(0x06); // λειτουργία ρύθμισης εισόδου:

// αύξηση αυτόματα, ότι δεν υπάρχει αλλαγή της οθόνης
delay(20); LcdCommandWrite(0x0E); // οθόνη ελέγχου:

// ενεργοποιήσετε οθόνη, δρομέα επάνω, δεν αναβοσβήνει
delay(20);

LcdCommandWrite(0x01); // ευκρινής οθόνη, θέτει τον δρομέα στο μηδέν
delay(100);

LcdCommandWrite(0x80); // οθόνη ελέγχου:

// ενεργοποιήσετε οθόνη, δρομέα επάνω, δεν αναβοσβήνει delay(20);
}

void loop (void) {

LcdCommandWrite (0x02); // ρύθμιση της θέσης του δρομέα στο μηδέν

Delay (10);

// γραφή του μηνύματος καλωσορίσματος

LcdDataWrite ('H');

LcdDataWrite ('o');

LcdDataWrite ('l');

LcdDataWrite ('a');

LcdDataWrite (' ');

LcdDataWrite ('C');

```



```
LcdDataWrite ('a');  
LcdDataWrite ('r');  
LcdDataWrite ('a');  
LcdDataWrite ('c');  
LcdDataWrite ('o');  
LcdDataWrite ('l');  
LcdDataWrite ('a');  
Delay (500);  
}
```

Παράδειγμα 4

Knock

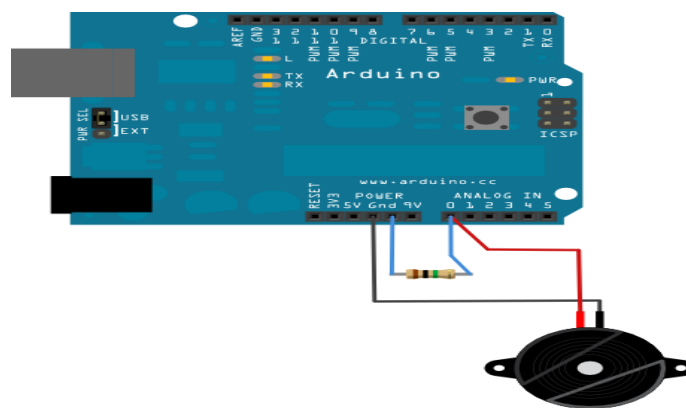
Χρησιμοποιούμε ένα στοιχείο Piezo για την ανίχνευση ήχου, αυτό που θα μας επιτρέψει να το χρησιμοποιήσουμε ως ένα αισθητήρα κτύπων. Λαμβάνουμε τα πλεονεκτήματα της ικανότητας του επεξεργαστή να διαβάσει αναλογικά σήματα μέσω του ADC - αναλογικό σε ψηφιακό μετατροπέα. Οι μετατροπείς διαβάζουν την αξία της τάσης και τη μετατρέπουν σε μια τιμή που κωδικοποιείται ψηφιακά. Στην περίπτωση της πλακέτας Arduino, έχουμε μετατρέψει την τάση σε αξία στο εύρος 0 .. 1024. Το 0 αντιπροσωπεύει τα 0volts, ενώ 1.024 αντιπροσωπεύουν 5volts στην είσοδο ενός από τις έξι αναλογικές καρφίτσες.

Το Piezo δεν είναι τίποτα παρά μια ηλεκτρονική συσκευή που μπορεί τόσο να χρησιμοποιηθεί για να παίζει ήχους όσο και να εντοπίσει τους τόνους. Στο παράδειγμά μας, συνδέουμε το Piezo στην αναλογική είσοδο αριθμό pin 0, που υποστηρίζει τη λειτουργία της ανάγνωσης μια τιμή μεταξύ 0 και 5volts, και όχι μόνο ένα απλό HIGH ή LOW.

Το άλλο πράγμα που πρέπει να θυμόμαστε είναι ότι τα Piezos έχουν πολικότητα, οι εμπορικές συσκευές έχουν συνήθως ένα κόκκινο και ένα μαύρο καλώδια υποδεικνύοντας πώς να το συνδέσουμε στη πλακέτα. Συνδέουμε το μαύρο στη

γείωση και το κόκκινο στην είσοδο. Έχουμε, επίσης, να συνδέσουμε μια αντίσταση στο εύρος των Mega ohms παράλληλα με το στοιχείο Piezo, στο παράδειγμα το έχουμε συνδεδεμένο απευθείας στις θηλυκές υποδοχές. Μερικές φορές είναι δυνατό να αποκτήσουμε στοιχεία Piezo χωρίς πλαστικό περίβλημα, τότε θα δείχνουν όπως ένας μεταλλικός δίσκος και είναι πιο εύκολο να χρησιμοποιηθούν ως αισθητήρες εισόδου.

Το παράδειγμα κώδικα θα συλλάβει το χτύπημα και αν είναι ισχυρότερη από ένα ορισμένο όριο, θα στείλει το string "Knock!" πίσω στον υπολογιστή μέσω της σειριακής θύρας.



Εικόνα 2.17: κύκλωμα χτύπου

CODE

```
/* Knock Sensor
```

```
*Το πρόγραμμα χρησιμοποιεί ένα στοιχείο Piezo σαν έναν αισθητήρα κτύπων.
```

```
* Έχουμε να ακούσουμε ουσιαστικά σε ένα αναλογικό pin και να εντοπίσουμε αν το  
σήμα πηγαίνει πάνω από ένα ορισμένο όριο. Γράφει "Knock" στη σειριακή θύρα, αν  
περνά το όριο, και εναλλάσσει το LED για το pin 13.
```

```
*/
```

```
int ledPin = 13; // Το led συνδέεται με τον έλεγχο pin 13
```

```
int knockSensor = 0; // ο αισθητήρας knock θα πρέπει να συνδεθεί στο αναλογικό pin  
0
```

```

byte val = 0; // μεταβλητή για να αποθηκεύσουμε την τιμή που διαβάζεται από τον
πείρο του αισθητήρα

int statePin = LOW; // μεταβλητή που χρησιμοποιείται για την αποθήκευση της
τελευταίας κατάστασης LED, για να αλλάξετε το φως

int THRESHOLD = 100; // τιμή κατωφλίου για να αποφασίσει πότε ο ανιχνεύσιμος
ήχος είναι ένα χτύπημα ή όχι

void setup() {

  pinMode(ledPin, OUTPUT); // κηρύσσει την ledPin όπως ΕΞΟΔΟΣ

  Serial.begin(9600); // χρησιμοποιήσετε τη σειριακή θύρα

}

void loop() {

  val = analogRead(knockSensor); // διαβάστε τον αισθητήρα και αποθηκεύστε το στη
μεταβλητή "val"

  if (val >= THRESHOLD) {

    statePin = !statePin; // εναλλαγή της κατάστασης του ledPin digitalWrite(ledPin,
statePin); // LED ενεργοποιημένο ή απενεργοποιημένο

    Serial.println("Knock!"); // στέλνει το string "Knock!" πίσω στον υπολογιστή, που
ακολουθείται από αλλαγή γραμμής

    delay(10); // μικρή καθυστέρηση για να αποφευχθεί η υπερφόρτωση της σειριακής
θύρας

  }

}

```

ΑΝΑΦΟΡΕΣ:

[1] http://www.ubuntucy.org/wiki/index.php/%CE%93%CE%BD%CF%89%CF%81%CE%AF%CF%83%CF%84%CE%B5_%CF%84%CE%BF_Arduino

[2] <http://www.howtogeek.com/99041/the-making-of-arduino-geek-history/>

- [3] <http://www.benarent.co.uk/revolve/video/the-history-of-arduino/>
- [4] <http://www.akouseto.gr/eisagogi-sto-arduino>
- [5] <http://arduino.cc/en/Guide/Introduction>
- [6] <http://www.arduino.cc/>
- [7] <http://www.kenleung.ca/portfolio/arduino-a-brief-history-3/>
- [8] <http://www.fieconference.org/fie96/papers/196.pdf>
- [9] <http://www.ijee.ie/articles/Vol13-4/ijee984.pdf>
- [10] <http://www.microplanet.gr/tutorials/microcontrollers/arduino>
- [11] <http://www.grobot.gr/index.php/2008-04-19-13-16-38/197-arduino-30432>
- [12] <http://en.wikipedia.org/wiki/Arduino>
- [13] <http://deltahacker.gr/2009/08/01/arduino-intro/>
- [14] <http://hacknmod.com/hack/top-40-arduino-projects-of-the-web/>
- [15] <http://www.arduino.gr.com/>

2.4 Ο μικροελεγκτής PIC

2.4.1 Η ιστορία του PIC

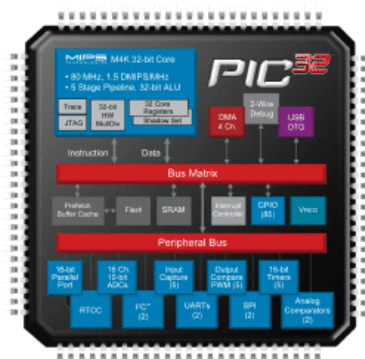
Πριν από χρόνια, το υπουργείο εθνικής άμυνας των Ηνωμένων Πολιτειών ζήτησε από τα Πανεπιστήμια του Harvard και το Princeton να σχεδιάσουν ένα τύπο υπολογιστή με σκοπό την σχεδίαση τροχιάς βλημάτων του πυροβολικού. Το Harvard παρουσίασε μια αρχιτεκτονική στα πλαίσια της οποίας το τμήμα μνήμης χωρίζεται σε δύο μέρη: το ένα χρησιμοποιείται για την αποθήκευση του προγράμματος και το άλλο για προσωρινή αποθήκευση μεταβλητών (τύπου μνήμης τυχαίας προσπέλασης ή RAM). Το πλεονέκτημα της μεθόδου αυτής ήταν ότι η ανάκληση των εντολών του προγράμματος από τη μνήμη ελέγχου μπορεί να πραγματοποιηθεί ταυτόχρονα με τη διαδικασία πρόσβασης στη μνήμη RAM. Επιταχύνοντας με τον τρόπο αυτό την εκτέλεση των εντολών. Η αρχιτεκτονική που παρουσίασε το Princeton και που

ανταγωνιζόταν την προηγούμενη, χρησιμοποιούσε την ίδια μνήμη ελέγχου και ως μνήμη RAM. Η δεύτερη αυτή αρχιτεκτονική είναι γνωστή ως “αρχιτεκτονική Von Neumann”. Η ιδέα του διαχωρισμού του συστήματος μνήμης σε μνήμη ελέγχου και μνήμη RAM, που επικρατούσε στην αρχιτεκτονική του Harvard, έδινε ένα πολύ πιο γρήγορο σύστημα από αυτό του Von Neumann αλλά πολύ πιο ακριβό. Έτσι η αρχιτεκτονική που προτάθηκε από τον Von Neumann του Princeton κέρδισε στον παραπάνω διαγωνισμό αλλά κέρδισε και σημαντικό έδαφος αφού αποτέλεσε τη βασική αρχιτεκτονική για τους περισσότερους τύπους υπολογιστών. Πολλά χρόνια αργότερα, μετά από αυτόν τον διαγωνισμό, κάποιοι σχεδιαστές μηχανικοί της εταιρίας General Instruments, θέλοντας να σχεδιάσουν ένα δικό τους τύπο μικροελεγκτή, έφεραν ξανά στο φως την αρχιτεκτονική του Harvard. Κρατώντας την μνήμη του συστήματος χωρισμένη σε δύο μέρη, το ένα για μνήμη ελέγχου και το άλλο για μνήμη RAM, ο σχεδιασμός του μικροελεγκτή πλεονεκτεί με την έννοια της απλότητας στην υλοποίηση αλλά και την εξαιρετικά γρήγορη λειτουργία (παράγοντες πολύ σημαντικοί για τους μικροελεγκτές). Στην πραγματικότητα, οι διατάξεις στις οποίες χρησιμοποιείται η αρχιτεκτονική Harvard είναι ελαφρά φθηνότερες από τις αντίστοιχες της κλασικής αρχιτεκτονικής Von Neumann γιατί στις πρώτες δεν είναι απαραίτητο να σχεδιάζονται μαζί η μνήμη ROM(EPROM) με τη μνήμη RAM. Χρησιμοποιώντας την αρχιτεκτονική Harvard, η σειρά των μικροελεγκτών PIC χαρακτηρίζεται από τη μεγάλη ταχύτητα λειτουργίας και από το χαμηλό κόστος σε σχέση με άλλα ανταγωνιστικά προϊόντα ανάλογου τύπου. Τα παραπάνω συνέβαιναν στα μέσα, περίπου, της δεκαετίας του 1970. Σήμερα η Microchip διαθέτει μια πλήρη σειρά μικροελεγκτών PIC τεχνολογίας CMOS με μια πληθώρα χαρακτηριστικών που τους καθιστά κατάλληλους για πάρα πολλές εφαρμογές. Από την άποψη της ποικιλίας, σε διαφορετικούς τύπους, της σειράς των μικροελεγκτών PIC, η Microchip χώρισε τις διατάξεις αυτές σε τρεις ομάδες: χαμηλής, μεσαίας και υψηλής τάξης. Με βάση αυτό το σχήμα, όλα τα μέρη της σειράς έχουν σχεδιαστεί ώστε να είναι, με τη λογική της μετάβασης από χαμηλότερη σε υψηλότερη τάξη, συμβατά ως προς το κώδικα του λογισμικού (οι εκδόσεις των PIC υψηλότερης τάξης μπορούν να «τρέξουν» προγράμματα που γράφηκαν για εκδόσεις χαμηλότερης τάξης), όπως επίσης διαθέτουν εσωτερικούς καταχωρητές δεσμευμένους στις ίδιες διευθύνσεις μνήμης, ώστε να απλοποιείται η μεταβίβαση προγραμμάτων μεταξύ διαφορετικών εκδόσεων της σειράς. Η αρχική PIC χτίστηκε

για να χρησιμοποιηθεί με νέο 16-bit CPU GI, το CP1600. Ενώ γενικά είναι μια καλή CPU, το CP1600 είχε φτωχή απόδοση I / O, και το 8-bit PIC αναπτύχθηκε το 1975 για να βελτιώσει την απόδοση του συνολικού συστήματος με την εκφόρτωση I / O εργασίες από την CPU. Το PIC χρησιμοποιείται με απλό μικροκώδικα αποθηκευμένο στη μνήμη ROM για την εκτέλεση των καθηκόντων του, αν και ο όρος δεν χρησιμοποιήθηκε κατά τη χρονική στιγμή, που μοιράζεται αρκετά κοινά χαρακτηριστικά με RISC σχέδια. Το 1985 η General Instruments διαχώρισε την μικροηλεκτρονική διαίρεσή τους, και η νέα ιδιοκτησία ακύρωσε σχεδόν τα πάντα - το οποίο αυτή τη φορά ήταν ως επί το πλείστον ξεπερασμένο. Το PIC, ωστόσο, αναβαθμίστηκε με το EPROM για να παράγει ένα προγραμματιζόμενο ελεγκτή καναλιού, και σήμερα μια τεράστια ποικιλία από μικροελεγκτές Pic είναι διαθέσιμοι με ποικιλία περιφερειακών πλακέτας (σειριακές μανάδες επικοινωνίας, UARTs, κινητήρας πυρήνα ελέγχου, κ.λπ.) και στη μνήμη του προγράμματος από 512 λέξεις σε 64k λέξεις και περισσότερο (μια "λέξη" είναι μία εντολή της γλώσσας assembly, που κυμαίνεται από 12, 14 ή 16 bits ανάλογα με τη συγκεκριμένη οικογένεια macro PIC). Η Microchip Technology δεν χρησιμοποιεί το PIC ως έχει. Στην πραγματικότητα το εμπορικό σήμα είναι PIC micro. Γενικά θεωρείται ότι ο PIC σημαίνει Peripheral Controller Interface, αν και το αρχικό ακρωνύμιο General Instruments για τις αρχικές PIC1640 και PIC1650 συσκευές ήταν "Programmable Controller Interface". Το ακρώνυμο γρήγορα αντικαταστάθηκε με το "Programmable Intelligent Computer". Η Microchip 16C84 (PIC16x84), που θεσπίστηκε το 1993, ήταν η πρώτη CPU με ενσωματωμένη μνήμη EEPROM. Αυτή η ηλεκτρικά διαγράψιμη μνήμη κατέστησε να κοστίζει λιγότερο από ότι οι επεξεργαστές που απαιτούν χαλαζία "erased window" για τη διαγραφή του EPROM. Ο PIC είναι μια οικογένεια της αρχιτεκτονικής μικροελεγκτών του Harvard κατασκευασμένος από την Microchip Technology, που προέρχεται από την PIC1640 που αναπτύχθηκε αρχικά από τη General Instrument's Microelectronics Division. Οι PICs είναι δημοφιλείς στους προγραμματιστές και τους χομπίστες λόγω του χαμηλού κόστους τους, την ευρεία διαθεσιμότητα, τη μεγάλη βάση χρηστών, την εκτενή συλλογή από σημειώσεις εφαρμογών, η διαθεσιμότητα των χαμηλού κόστους ή τα δωρεάν εργαλεία ανάπτυξης, και την ικανότητα σειριακού προγραμματισμού (και επανα-προγραμματισμό με μνήμη flash). Η Microchip πρόσφατα ανακοίνωσε την αποστολή των 6 εκατομμυρίων επεξεργαστών PIC.

2.4.2 Τι είναι ο PIC

Οι μικροελεγκτές και οι άνθρωποι επικοινωνούν μέσω της γλώσσας προγραμματισμού που ονομάζεται Assembly. Πιο συγκεκριμένα, η γλώσσα Assembly είναι μια μεταβατική λύση. Για να μπορεί ο μικροελεγκτής να καταλάβει ένα πρόγραμμα γραμμένο σε συμβολική γλώσσα, θα πρέπει να συγκεντρώνεται σε μια γλώσσα από μηδενικά και μονάδες. Η γλώσσα Assembly και συναρμολόγησης δεν έχουν την ίδια έννοια. Η πρώτη αφορά το σύνολο των κανόνων που χρησιμοποιούνται για τη σύνταξη του προγράμματος για το μικροελεγκτή, ενώ το δεύτερο αναφέρεται σε ένα πρόγραμμα σε έναν προσωπικό υπολογιστή που χρησιμοποιείται για να μεταφράσει την γλώσσα Assembly σε καταστάσεις από μηδενικά και μονάδες. Ένα εκτελέσιμο πρόγραμμα ονομάζεται επίσης Κώδικας Μηχανής. Το "Πρόγραμμα" είναι ένα αρχείο δεδομένων που αποθηκεύεται στο σκληρό δίσκο του υπολογιστή (ή στη μνήμη του μικροελεγκτή, αν φορτωθεί) και γράφεται σύμφωνα με τους κανόνες της Assemble ή κάποιας άλλης γλώσσας προγραμματισμού. Η γλώσσα Assembly είναι κατανοητή για τους ανθρώπους, επειδή αποτελείται από σημαντικές λέξεις και από σύμβολα του αλφαβήτου. Ας πάρουμε για παράδειγμα, την εντολή " RETURN ", η οποία είναι, ότι υποδηλώνει το όνομά της, η οποία χρησιμοποιείται για να επιστρέψει ο μικροελεγκτής από μια υπορουτίνα. Σε κώδικα μηχανής, η ίδια εντολή αντιπροσωπεύεται από μια 14 -bit συστοιχία από μηδενικά και μονάδες κατανοητές από τον μικροελεγκτή . Όλες οι εντολές στη γλώσσα Assembly καταρτίζονται στην αντίστοιχη σειρά από μηδενικά και μονάδες. Ένα αρχείο δεδομένων που χρησιμοποιείται για την αποθήκευση καταρτιζόμενων προγραμμάτων ονομάζεται «εκτελεστικό αρχείο», δηλαδή « αρχείο δεδομένων HEX» . Το όνομα προέρχεται από την δεκαεξαδική παρουσίαση ενός αρχείου δεδομένων και έχει κατάληξη " Hex", όπως για παράδειγμα, " probe.hex ". Μετά τη δημιουργία , το αρχείο δεδομένων φορτώνεται στον μικροελεγκτή χρησιμοποιώντας έναν programmer. Προγράμματα της γλώσσας Assembly μπορούν να γραφτούν σε οποιοδήποτε πρόγραμμα επεξεργασίας κειμένου (editor) σε θέση να δημιουργούνται αρχεία δεδομένων ASCII σε ένα σκληρό δίσκο ή σε ένα εξειδικευμένο περιβάλλον εργασίας, όπως το MPLAB που θα περιγραφεί αργότερα.



Εικόνα 2.18: μικροελεγκτής PIC

2.4.3 Γλώσσα προγραμματισμού που χρησιμοποιείται

Ένα πρόγραμμα γραμμένο σε γλώσσα assembly αποτελείται από διάφορα στοιχεία που ερμηνεύονται διαφορετικά κατά την κατάρτιση του προγράμματος σε ένα εκτελέσιμο αρχείο δεδομένων. Η χρήση αυτών των στοιχείων απαιτεί αυστηρούς κανόνες και είναι απαραίτητο να δοθεί ιδιαίτερη προσοχή σε αυτά κατά τη διάρκεια της γραφής του προγράμματος, ώστε να αποφευχθούν λάθη. Όπως αναφέρθηκε, είναι απαραίτητο να τηρούν ορισμένους ειδικούς κανόνες προκειμένου να καταστεί δυνατή η διαδικασία κατάρτισης σε εκτελέσιμο κωδικό HEX να τρέξει χωρίς σφάλματα. Οι υποχρεωτικοί κανόνες εξηγούν πώς οι ακολουθίες εκφράσεων τοποθετούνται μαζί για να σχηματίσουν τις δηλώσεις που συνθέτουν ένα πρόγραμμα σε γλώσσα assembly που ονομάζεται σύνταξη. Υπάρχουν μόνο μερικά από αυτά:

Κάθε γραμμή του προγράμματος μπορεί να αποτελείται από το πολύ 255 χαρακτήρες

Κάθε γραμμή του προγράμματος που πρόκειται να καταρτιστούν πρέπει να ξεκινήσει με ένα σύμβολο, ετικέτα, οδηγία

Κείμενο μετά το σήμα "?" Σε μια γραμμή πρόγραμμα αποτελεί ένα σχόλιο που αγνοείται από τον assembler (δεν καταρτίζονται)

Και όλα τα στοιχεία του προγράμματος μιας γραμμής (ετικέτες, οδηγίες, κλπ.) πρέπει να διαχωρίζονται τουλάχιστον από έναν χαρακτήρα χώρου. Για χάρη της καλύτερης σαφήνειας, χρησιμοποιείται συνήθως ένα κουμπί TAB, έτσι ώστε να είναι εύκολο να οριοθετήσει στήλες με τις οδηγίες, τις ετικέτες, κλπ. σε ένα πρόγραμμα.

2.4.4 Πλεονεκτήματα του PIC

- Πρόκειται για έναν σχεδιασμό RISC (reduced instruction set computer)
- Έχουμε να θυμόμαστε μόνο 37 οδηγίες
- Έχει υψηλή ταχύτητα ρολογιού
- Είναι χαμηλού κόστους
- Ο κώδικας είναι εξαιρετικά αποτελεσματικός, επιτρέποντας στον PIC να τρέχει τυπικά με λιγότερη μνήμη προγράμματος από τους μεγαλύτερους ανταγωνιστές
- Χτισμένο σε ταλαντωτή με δυνατότητα επιλογής ταχυτήτων
- Η διαθεσιμότητα των επεξεργαστών στο πακέτο DIP καθιστούν εύκολο το χειρισμό από χομπίστες.

2.4.5 Εκδόσεις PIC

I. 12Cxxx οικογένεια

- ✓ 12/14 bit των εσωτερικών λειτουργιών
- ✓ 33/35 οδηγίες
- ✓ 0,4 sec οδηγίες κυκλικού χρόνου (ελάχιστος χρόνος για την εκτέλεση εντολών)
- ✓ PIC12F675 υψηλή απόδοση σε Flash μνήμη
- ✓ PIC12F675 έχει 1k του κωδικού χώρου (πρόγραμμα μνήμης), 64 bytes της μνήμης RAM και 128 bytes της EEPROM και τρέχει ως 20MHz ταχύτητα ρολογιού

II. 16C5xx οικογένεια

- ✓ 12 bit εσωτερικών λειτουργιών
- ✓ 32 οδηγίες
- ✓ 0,2μs (200ns) οδηγίες κυκλικού χρόνου

III. 16CFxx οικογένεια

- ✓ 14 bit εσωτερικών λειτουργιών
- ✓ 35 οδηγίες
- ✓ 0,2μs οδηγίες κυκλικού χρόνου

Π.χ. 16F877A

- Υψηλή απόδοση
- Μνήμη Flash
- 8k x14 κωδικού χρόνου
- 368 bytes RAM
- 256 bytes της EEPROM
- Σε έναν μόνο κύκλο (0,2sec) έχουμε οδηγίες για όλα εκτός από την εντολή διακλάδωσης (ρολόι 20MHz)

- Η διακλάδωση διαρκεί 2 κύκλους
- IV. 17C5xx οικογένεια**
 - ✓ 16 bit εσωτερικών λειτουργιών
 - ✓ 58 οδηγίες
 - ✓ 0,12μs χρόνος για οδηγίες κυκλικού χρόνου
- V. 17Cxx οικογένεια**
 - ✓ 16 bit ενσωματωμένων εσωτερικών λειτουργιών
 - ✓ 77 οδηγίες
 - ✓ 0,1μs χρόνος για ελάχιστη εκτέλεση των εντολών χρόνου
- VI. 18Fxx οικογένεια**
 - ✓ Νέα και προηγμένα χαρακτηριστικά – δύο φορές ο χώρος μνήμης του προγράμματος και 4 φορές μεγαλύτερη μνήμη RAM, δύο φορές την ταχύτητα-

Π.χ. 18F452

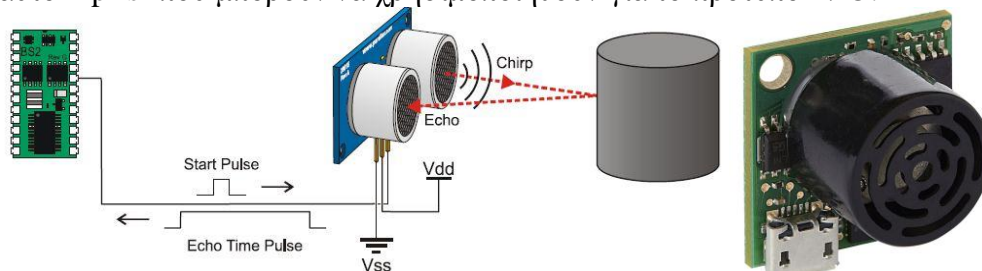
- Υψηλή απόδοση και μνήμη Flash
- 16 kB κώδικα χώρου, 1536 bytes μνήμης RAM
- 256 bytes EEPROM
- Λειτουργεί έως και 40MHz ταχύτητα ρολογιού
- VII. PIC32 οικογένεια**
 - ✓ 32 bit εσωτερικών λειτουργιών
 - ✓ Υψηλότερη DMIPS ανά MHz λειτουργία ρολογιού
 - ✓ Αποτελεσματική εσωτερική αρχιτεκτονική διαύλου
 - ✓ Προηγμένες λειτουργίες διδασκαλίας και DMA κανάλια ελεγκτή
 - ✓ Μνήμη Flash 32Kb και 512kB
 - ✓ Ενσωματωμένα με CAN, USB και Ethernet
 - ✓ Λειτουργίες ρολογιού 80MHz (εσωτερική)
 - ✓ Απόδοση 1,56 DMIPS ανά MHz
 - ✓ Γρήγορη εναλλαγή πλαισίου (δεδομένων από τη CPU και την κατάσταση μητρώων, αποθηκεύονται αυτόματα στην κλήση ή διακόπτονται και η αποκατάσταση γίνεται αυτόματα)
 - ✓ 8 κανάλια ελεγκτή DMA

2.4.6 Εφαρμογές του PIC

i. A PIC Sonar (Ultrasonic) Range Finding Project

Στο project αυτό ο μικροελεγκτής PIC με βάση σόναρ διαδίδει ένα σύντομο παλμό της συχνότητας θορύβου, αδύνατο να ακουστεί από τα ανθρώπινα αυτιά, δηλαδή υπερηχητικό ήχο ή υπέρηχο. Αργότερα ο μικροελεγκτής παρατηρεί την ηχώ της διάδοσης του θορύβου. Η σύνδεση από τη διάδοση

του θορύβου στη λήψη του ήχου, εκτιμάται από την απόσταση του αντικειμένου. Στο project αυτό της εμβέλειας σόναρ γίνεται χρήση 5 προτύπων τρανζίστορ για τη λήψη και τη διάδοση των υπερηχητικών ήχων και σύγκρισης για την τοποθέτηση του ορίου αναγνώρισης επιπέδου ήχου - έτσι δεν υπάρχουν μοναδικά συστατικά εκτός από το μικροελεγκτή. Ο υπερηχητικός μετατροπέας ήχου είναι συνηθισμένος στα 40 kHz ταξινόμησης. Σημείωση-ο εσωτερικός ταλαντωτής του μικροελεγκτή PIC χρησιμοποιεί και αυτό 2 pins που μπορούν να χρησιμοποιηθούν για το πρότυπο I / O.



Εικόνα 2.19: A PIC Sonar (Ultrasonic) Range Finding

ii. PIC Based BRAM (Beginner's Robot Autonomous Mobile)

Το έργο αυτό παρουσιάζει πώς να αναπτύξουμε μια BRAM. Αυτό προορίζεται να κατασκευαστεί εύκολα φέρνοντας σε χρήση κάποια από τα συστατικά που μπορούν να ανακαλυφθούν εύκολα στο σπίτι. Το κλειδί ελέγχου για αυτό το ρομποτικό έργο είναι ένα μικροτσίπ (PIC16F690). 2 παλιά CDs χρησιμοποιούνται για την ανάπτυξη του πλαισίου για το ρομποτικό σύστημα. Ο προσανατολιζόμενος κινητήρας DC, ο τροχίσκος, η ισχύς της μπαταρίας και ο προφυλακτήρας κλειδί του ρομπότ πιάνονται στο κατώτερο στρώμα, ενώ το ανώτερο στρώμα αποτελείται από τον πίνακα αισθητήρων του ρομπότ, το μικροτσίπ PIC16F690 & τις οδηγίες του κινητήρα.

Παρακάτω δίνεται το υλικό κατασκευής του Bram:

2 CD ή DVD για το σασί

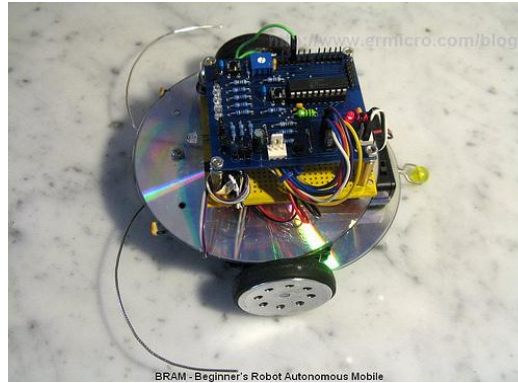
2 προσανατολιζόμενοι κινητήρες DC με τον τροχό ή μπορεί να χρησιμοποιηθούν αλλαγές στο σερβοκινητήρα

Ένα 3 από 1,5 volt AA κιβώτιο της μπαταρίας με κουμπιά ON-OFF

1 πλαστικό σφαιρίδιο και 1 συνδετήρας για τον τροχό

2 micro κλειδιά και 2 συνδετήρες για τον αισθητήρα προφυλακτήρων

Βίδες, τυπωμένο κύκλωμα, ξηροί καρποί, κάτοχοι, ταινία διπλής όψης για να αγκαλιάσει όλα αυτά τα συστατικά μαζί.



Εικόνα 2.20: PIC Based BRAM (Beginner's Robot Autonomous Mobile)

iii. Versatile Central Heating Program Controller using PIC16F628A

Αυτός ο ευέλικτος ελεγκτής κεντρικού συστήματος θέρμανσης έχει ως στόχο να γίνει χρήση ενός λέβητα. Τα 2 ρελέ ελέγχουν το ζεστό νερό και την παροχή θερμότητας. Περιέχει μια πρόσοψη ελέγχου με οθόνη LCD των 16×2 . Δίνει επίσης μια διαδοχική ένωση που επιτρέπει να εργαζόμαστε από απόσταση μέσω της βοήθειας υπολογιστή. Τα ρελέ ελέγχου του προγραμματιστή & του λέβητα θέρμανσης κρατούνται σε διαφορετικές μονάδες απλά για να εντοπίσουμε τα ρελέ κοντά στο λέβητα, ενώ ο προγραμματιστής μπορεί να τοποθετηθεί οπουδήποτε στην κατοικία που χρησιμοποιεί χαμηλή τάση πίσω στο συστατικό ρελέ. Επιπλέον, μπορούμε επίσης να αναπτύξουμε μια σειριακή σύνδεση διεπαφή γειτονική με τον προγραμματιστή όπου σε αυτή την περίπτωση απαιτούνται μόνο 4 καλώδια για την ενέργεια και τα ρελέ ελέγχου.

χαρακτηριστικά :

Αυτο -ρύθμιση για κεντρική θέρμανση και boiler .

Δέκα ευέλικτα προγράμματα .

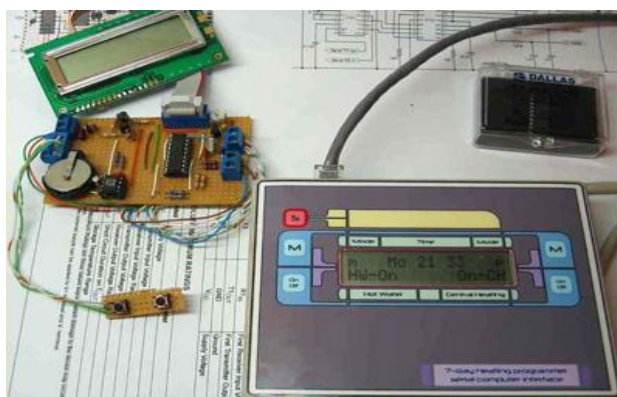
Εγχειρίδιο λειτουργίας και ρύθμιση από τον πίνακα πρόσοψης ή απομακρυσμένα

Υποστήριξη της μπαταρίας για RTC (Real Time Clock) .

Προγραμματιστής τοποθετημένος σε απόσταση από το λέβητα μπορεί να χρησιμοποιήσει 6 -πυρήνες καλωδίου συναγερμού .

Πρόσοψη μπορεί να κλειδωθεί

Βασισμένο στο Microchip PIC 16F628 (μικροελεγκτή) .



Εικόνα 2.21:Versatile Central Heating Program Controller

iv. A Versatile Temperature Data Logger Using PIC12F683 and DS1820

Εδώ παρουσιάζουμε το project καταγραφής θερμοκρασίας δεδομένων που βασίζεται σε 8 - pin micro - controller ενός μικροτσίπ (PIC12F683) . Μελετά τα στοιχεία της θερμοκρασίας από ένα ψηφιακό αισθητήρα (DS1820) και τα συσσωρεύει στο εσωτερικό της EEPROM. Ο μικροελεγκτής έχει 256 bytes των εγχώριων EEPROM και οι τιμές της θερμοκρασίας θα πρέπει να αποθηκεύονται σε μορφή 8 -bit. Αυτό σημαίνει ότι τα 8 ζωτικά bits των τιμών θερμοκρασίας από τον ψηφιακό αισθητήρα θα πρέπει να μελετηθούν και η ανάλυση της θερμοκρασίας θα είναι ένας βαθμός C.

Θερμοκρασία καταγραφικού χαρακτηριστικά :

Data Logger

Ερμηνεύει θερμοκρασία από ένα ψηφιακό αισθητήρα και συσσωρεύεται στην εσωτερική μνήμη EEPROM

Μπορεί να συσσωρεύονται περίπου 254 τιμές θερμοκρασίας. Η EEPROM θέση " 0 " χρησιμοποιείται για να αποθηκεύσουμε τις αλλαγές δειγματοληψίας , και η θέση "1" χρησιμοποιείται για να σώσει το ποσό των εγγραφών .

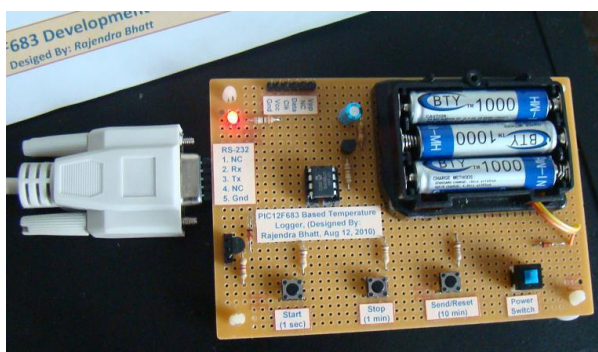
Υπάρχουν 3 εναλλακτικά διαλείμματα δειγματοληψίας: 1 δευτερόλεπτο , 1 λεπτό , και 10 λεπτών. Αυτό μπορεί να επιλεγεί κατά την ενεργοποίηση .

Πλήκτρα έναρξης και διακοπής για χειροκίνητο έλεγχο .

Οι καταγεγραμμένες τιμές αποστέλλονται σε υπολογιστή μέσω της σειριακής θύρας . Ένα κουμπί αποστολής είναι υφιστάμενο για να ξεκινήσει η μεταφορά δεδομένων .

Ένα LED για να δείξει διαφορετικές τρέχουσες διαδικασίες.

Επανατοποθέτηση κλειδιού για να διαγράψετε όλα τα προηγούμενα δεδομένα

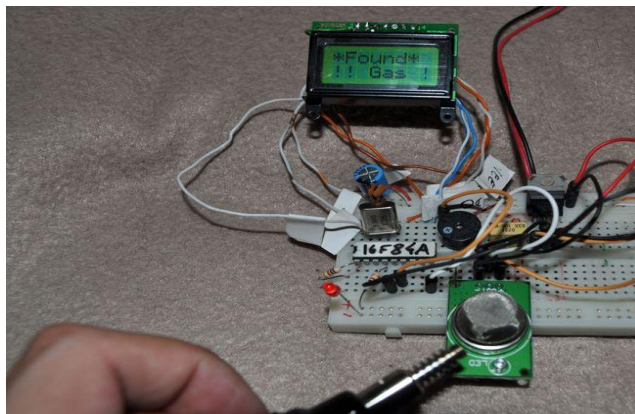


Εικόνα 2.22: A Versatile Temperature Data Logger Using PIC12F683 and DS18B20

v. Gas Sensor Using PIC16F84A

Εδώ επιδεικνύουμε ένα κύκλωμα αισθητήρα αερίου που υποστηρίζεται από τον μικροελεγκτή PIC16F84A & τον αισθητήρα GH-312. Ο GH-312 είναι ικανός να ανιχνεύει αέρια, όπως υδροποιημένο αέριο, προπάνιο, καπνό, το αλκοόλ, το βουτάνιο, το μεθάνιο, το υδρογόνο, κλπ. Όπως ανιχνεύει οποιοδήποτε από αυτά τα αέρια, ζητά από το μικροελεγκτή (PIC16F84A), ο οποίος σε αντάλλαγμα ανάβει το βομβητή και το LED λάμπει. Στο project έχουμε χρησιμοποιήσει 9 βολτ

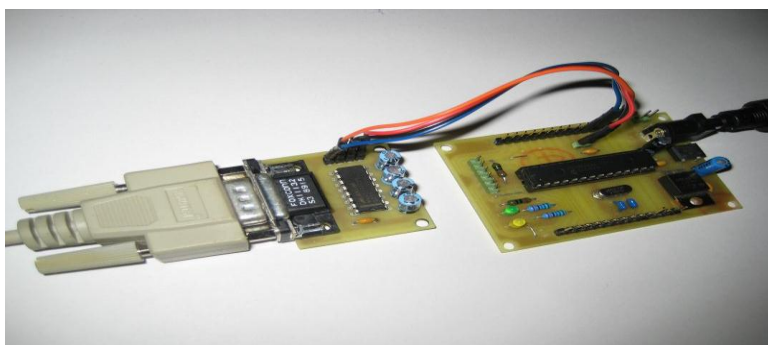
μπαταρία, καθώς ο αισθητήρας χρειάζεται μια είσοδο 9 Volts. Η έξοδος του αισθητήρα, όταν ζητάτε από το μικροελεγκτή είναι 5V το οποίο είναι ιδανικό για μια απaréγκλιτη ένωση σε κάθε μικροελεγκτή. Αν χρησιμοποιήσουμε μπαταρία 9V, κάθε παροχή 12 βολτ ενέργειας θα εργάζεται άψογα, όπως ο αισθητήρας μπορεί να διαχειριστεί από τα 9 volt σε 20 volt και η τάση του μικροελεγκτή συγχρονίζεται με έναν ελεγκτή 7805.



Εικόνα 2.23: Gas sensor

vi. RS232 Communication with PIC Microcontroller

Το project αυτό παρουσιάζει πώς να εκτελέσει μια απλή επικοινωνία μέσω μιας διεπαφής RS232 χρησιμοποιώντας PIC μικροελεγκτή. Το RS232 είναι φυσιολογικό για μια διαδοχική διεπαφή επικοινωνίας που να επιτρέπει να μεταδώσει και να λάβει δεδομένα. Με διεπαφή RS232 αυτό είναι εφικτό, να οργανώσει μια σύνδεση ανάμεσα σε ένα μικροελεγκτή και έναν υπολογιστή, μέσω της θύρας COM υπολογιστή ή μέσα σε 2 μικροελεγκτές. Το RS232 χρησιμοποιείται για διάφορους λόγους, όπως η μετάδοση εντολών-PC σε ένα μικροελεγκτή, μεταφέροντας πληροφορίες αποσφαλμάτωσης από ένα μικροελεγκτή σε ένα τερματικό, κατέβασμα τελευταίας έκδοσης firmware στο μικροελεγκτή και διάφορα άλλα πράγματα. Το PC θα ενσωματωθεί με πρόγραμμα τερματικού για να λαμβάνουμε και να στέλνουμε τα δεδομένα. Τα δεδομένα που μεταφέρονται μέσω του μικροελεγκτή εμφανίζεται στο παράθυρο του τερματικού και το πλήκτρο(as) που ωθείται εντός του τερματικού σταθμού θα μεταφέρουν το αντίστοιχο κωδικό κλειδί για τον μικροελεγκτή.



Εικόνα 2.24: RS232 Communication with PIC Microcontroller

vii. 3 LED Bike Light using PIC10F200

Σε αυτό το project είναι ένα πολύ-λειτουργικό LED φως ποδηλάτου, κάνοντας χρήση των 3 LEDs. Το project υποστηρίζεται από την αρχική τιμή (PIC10F200) μικροελεγκτή, που λειτουργεί από μια παροχή τάσης 2-5 volt. Στη stand-by μορφή χρησιμοποιεί μια δύναμη μικρότερη από 1μΑ δημιουργώντας το ιδανικό ταιρί για τη λειτουργία της μηχανοκίνητης μπαταρίας. Απασχολεί 3 ξεχωριστά υψηλής έντασης LED, και ένα πλήκτρο για την εναλλαγή του φωτός ON-OFF και την αλλαγή των τρόπων λειτουργίας.



Εικόνα 2.25: 3 Led bike light

viii. Telephone Operated Remote Control Using PIC16F84A Microcontroller

Αυτός ο σχεδιασμός του έργου διαχειρίζεται τουλάχιστον οκτώ συσκευές , φέρνοντας στο προσκήνιο ένα μικροελεγκτή PIC γνωστό ως PIC16F84A , σε συνδυασμό με την τηλεφωνική γραμμή . Το αποκλειστικό θέμα εδώ είναι ότι δεν είναι σαν τα άλλα τηλεχειριστήρια τηλεφωνικής γραμμής. Το εργαλείο αυτό δεν απαιτεί την κλήση για να απαντήσει στο απομακρυσμένο άκρο , ως εκ τούτου , θα

πρέπει να εφαρμόζονται χωρίς καμία χρέωση . Αυτή η μικροεφεύρεση εξαρτάται από την ποσότητα των δακτυλίων που δίνονται στην τηλεφωνική γραμμή για την τόνωση ή την αποσύνδεση των συσκευών .

Οδηγίες τηλεφώνου για τη λειτουργία τηλεχειριστήριου-κλειδιού :

Ενώ αναπτύσσουμε το κεντρικό κύκλωμα, πρέπει να βεβαιωθούμε ότι έχουμε φέρει στο προσκήνιο μια υποδοχή 18 ακίδων για το μικροελεγκτή. Δεν κολλάμε άμεσα IC στην πλακέτα κυκλώματος, καθώς μπορεί να απαιτήσει την απομάκρυνση του για τον προγραμματισμό . Πριν γίνει χρήση του PIC στο κεντρικό κύκλωμα, πρώτα το προγραμματίζουμε . Υπάρχει ένας αριθμός των προγραμματιστών που διατίθενται στο δίκτυο για να προγραμματίσουμε μικροελεγκτές PIC .

Βγάζουμε τον PIC από την υποδοχή προγραμματισμού 18 ακίδων και το τοποθετούμε μέσα στην υποδοχή του κεντρικού κυκλώματος.

Τώρα καθορίζουμε το κύκλωμα στην τηλεφωνική γραμμή και ενεργοποιούμε την παροχή ρεύματος .

Τώρα το κύκλωμα της πλακέτας είναι έτοιμο να δοκιμαστεί.

ix. Automated Town Water Management System

Ένα από τα σημαντικά χαρακτηριστικά της διαχείρισης κάθε πόλης, περιλαμβάνει τη διαχείριση των υδάτων . Πρόκειται για ένα θεμελιώδες χαρακτηριστικό, καθώς αυτές τις ημέρες οι πηγές νερού είναι εξαιρετικά περιορισμένες και κανείς δεν μπορεί να αντέξει αυτή την απώλεια. Αυτό το έργο διαχείρισης των υδάτων μιλά για την αυτοματοποίηση στην κατανομή και τη διαχείριση των υδάτων με τις τεχνολογικές εξελίξεις . Μια ποικιλία από τις πτυχές που ενσωματώνονται στο σύστημα είναι οι παρακάτω :

-Κινητή ελεγχόμενη κατανομή του νερού σε διάφορες περιοχές .

Ταχύτητα ελέγχου του κινητήρα σε σχέση με το επίπεδο του νερού της δεξαμενής .

Υπολογισμός λογαριασμού με βάση του νερού που καταναλώνεται .

Κατανομή του νερού , σύμφωνα με την εξόφληση του λογαριασμού .

Ενημερώσεις & status στα κινητά τηλέφωνα μέσω G.S.M Module .

Φωνητικές δηλώσεις στο γραφείο σχετικά με το status .

Data Logger στο διοικητικό κέντρο για τη στατιστική ανάλυση .

2.5 Αναλυτική παρουσίαση του μικροελεγκτή PIC

2.5.1 Διάφοροι τύποι PIC

Με την πάροδο του χρόνου, όλο και περισσότεροι τύποι PIC προστίθενται στη γραμμή παραγωγής. Αν και δεν έχει διερευνηθεί κάθε ένα μέλος της οικογένειας των PIC, ξεχωριστά, ωστόσο υπάρχουν αρκετές πληροφορίες σχετικά με τα χαρακτηριστικά των διαφόρων τύπων ώστε να επιλέξουμε τον καλύτερο για κάποια συγκεκριμένη εφαρμογή.

- **PIC 8 ακροδεκτών**

Προορίζονται για πολύ απλές εφαρμογές. Οι ελεγκτές αυτοί διαθέτουν μέχρι και έξι ακροδέκτες εισόδου- εξόδου (I/O) όπως επίσης και άλλα εξειδικευμένα χαρακτηριστικά (συμπεριλαμβανομένων και μονάδων μετατροπής αναλογικού σήματος σε ψηφιακό ή ADC). Τα μέλη της σειράς PIC12C5xx είναι ουσιαστικά ελεγκτές χαμηλής τάξης με δυνατότητα προγραμματισμού εντός του κυκλώματος. Επίσης διαθέτουν έναν εσωτερικό ταλαντωτή RC (που σημαίνει ότι δεν απαιτούνται εξωτερικοί ακροδέκτες χρονισμού του ελεγκτή) και εσωτερικό κύκλωμα επανατοποθέτησης. Με βάση αυτές τις ιδιότητες, οι συγκεκριμένοι ελεγκτές διατίθενται με έξι ακροδέκτες εισόδου-εξόδου και δύο ακροδέκτες τροφοδοσίας.

- **PIC χαμηλής τάξης**

Σχεδιάστηκε για πρώτη φορά προς τα τέλη του 1980. Αποτελούνται από ένα βασικό πυρήνα επεξεργαστή χωρίς συστήματα διακοπών και με πολύ περιορισμένη στοίβα μετρητή προγράμματος. Δεν διαθέτουν κανένα από τα χαρακτηριστικά των PIC μεσαίας τάξης (όπως εσωτερικούς μετατροπείς

ADC, σειριακές θύρες, διαύλους διασύνδεσης με επεξεργαστές κλπ.). Το μεγάλο πλεονέκτημά τους είναι το πολύ χαμηλό κόστος αν και το πλεονέκτημα αυτό το χάνει μέρα με τη μέρα επειδή όλο και περισσότεροι χαμηλού κόστους ελεγκτές μεσαίας τάξης είναι πλέον διαθέσιμοι.

- **PIC μεσαίας τάξης**

Στη σειρά των ελεγκτών PIC μεσαίας τάξης υπάρχει εξαιρετικά μεγάλη ποικιλία διαφορετικών διατάξεων στη γραμμή παραγωγής. Οι διατάξεις των μικροελεγκτών αυτών διαθέτουν από 13 μέχρι και 54 ακροδέκτες θυρών εισόδου-εξόδου, πολλαπλούς χρονιστές, εξόδους διαμόρφωσης εύρους παλμών (PWM I/O), δυνατότητα ασύγχρονης επικοινωνίας, άμεσο έλεγχο οθονών υγρών κρυστάλλων (LCD), μνήμη προγράμματος τύπου EEPROM και πολλά άλλα σημαντικά χαρακτηριστικά. Από τις πιο ενδιαφέρουσες εκδόσεις αυτής της σειράς των PIC, είναι εκείνοι που σχεδιάστηκαν με δυνατότητα άμεσης διασύνδεσης με οθόνες LCD. Η βασική CPU μεσαίας τάξης διαθέτει ένα ρεπερτόριο 35 εντολών, σύστημα διακοπών και μια στοίβα διακοπών και μια στοίβα μετρητή προγράμματος με βάθος οκτώ θέσεων. Οι πηγές του συστήματος διακοπών μπορεί να είναι καταστάσεις ακροδεκτών εισόδου, υπερχειλίση χρονιστών, λήψη σειριακών δεδομένων, σήμανση τέλους μετατροπής του εσωτερικού ADC και πολλές άλλες ακόμη. Το μέγεθος της μνήμης προγράμματος, που ανέρχεται μέχρι και τα 8Kbytes, επιτρέπει τη χρήση αυτών των διατάξεων σε αρκετά πολύπλοκες εφαρμογές.

- **PIC υψηλής τάξης**

Έχοντας τη δυνατότητα πρόσβασης σε μνήμη μέχρι και 64K λέξεων των 16-bits, προσφέρουν πλήρεις λύσεις περισσότερο στη σχεδίαση συστημάτων, παρά στο παραδοσιακό μόνο έλεγχο ευφυϊών κυκλωμάτων. Ο πυρήνας κεντρικής επεξεργασίας των ελεγκτών αυτών, επιτρέπει μεγάλη ευελιξία στη μετακίνηση εσωτερικών δεδομένων και προσφέρει πολλαπλά διανύσματα διακοπών για διάφορα συστήματα διαχείρισης διακοπών. Όλες αυτές οι ιδιότητες προσφέρουν πολύ πιο αποδοτικό προγραμματισμό για τις διάφορες εφαρμογές.

2.5.2 Τύποι της μνήμης προγράμματος

Οι διάφοροι τύποι μνημών για την αποθήκευση προγραμμάτων, που μπορούμε να συναντήσουμε στους ελεγκτές PIC, είναι οι τρεις που ακολουθούν:

- ❖ EPROM (Erasable Programmable ROM) ή “Έπαναπρογραμματιζόμενη Μνήμη Ανάγνωσης μόνο”.
- ❖ EEPROM (Electrically Erasable Programmable ROM) ή “Ηλεκτρικά Έπαναπρογραμματιζόμενη Μνήμη Ανάγνωσης μόνο”, η οποία συχνά αναφέρεται και ως E2PROM.
- ❖ Μνήμη καλωδιωμένης λογικής (Hardwired) ή “Μνήμη Ανάγνωσης μόνο” ROM.

Κάθε ένας από αυτούς τους τύπους μνήμης έχει τα πλεονεκτήματά του, σε διάφορες περιπτώσεις. Ο βασικός τύπος μνήμης που χρησιμοποιείται στους ελεγκτές PIC για την αποθήκευση του προγράμματος είναι ο τύπος EPROM. Ο λόγος για τον οποίο είναι επιθυμητό να προγραμματίζονται οι ελεγκτές αφού έχει κατασκευαστεί και συναρμολογηθεί το σχετικό κύκλωμα, είναι ότι, με τον τρόπο αυτό μειώνεται ο αριθμός των διαφορετικών PIC που πρέπει να καταχωρεί ο κατασκευαστής του κυκλώματος σε κάποιο κατάλογο. Επίσης με τον τρόπο αυτό απλουστεύεται η διαδικασία παραγωγής, αφού έχει παρακαμφθεί το στάδιο του προγραμματισμού καθώς και οι σχετικές διαδικασίες διακίνησης και αναγνώρισης των προγραμματισμένων εξαρτημάτων. Η μέθοδος αυτή είναι εξάλλου γνωστή και με την ονομασία In-System-Programming (ISP), που σημαίνει: “Προγραμματισμός Εντός του Συστήματος”.

2.5.3 Αρχιτεκτονική του επεξεργαστή PIC

▪ Η κεντρική μονάδα επεξεργαστή (CPU)

Ο επεξεργαστής του PIC είναι αρχιτεκτονικής τύπου RISC, με ξεχωριστή μνήμη εντολών και δεδομένων. Η ουσία στον επεξεργαστή αυτό είναι ο πλήρης διαχωρισμός του τμήματος των μεταβλητών και των καταχωρητών εισόδου-εξόδου ή απλά του τμήματος της μνήμης, από την μνήμη προγράμματος. Στους περισσότερους τύπους αρχιτεκτονικής μικροεπεξεργαστών, οι μεταβλητές, το πρόγραμμα και τα τμήματα εισόδου-εξόδου, καταλαμβάνουν τον ίδιο χώρο εκτέλεσης. Αυτό σημαίνει ότι η διαδικασία ανάκλησης των εντολών του προγράμματος βρίσκεται σε αναμονή, μέχρι να ολοκληρωθούν οι λειτουργίες εισόδου-εξόδου και ανάγνωσης/εγγραφής μεταβλητών, πριν από την ανάκληση της επόμενης εντολής από τη μνήμη. Η μονάδα

κεντρικής επεξεργασίας (CPU) του PIC μπορεί να θεωρηθεί ως μια αριθμητική λογική μονάδα (ALU) η οποία τροφοδοτείται και παραδίδει ή και παραλαμβάνει δεδομένα, από τους διάφορους καταχωρητές. Υπάρχει ένας αριθμός από καταχωρητές ειδικής χρήσης, που ελέγχουν τη λειτουργία της CPU, τους καταχωρητές εισόδου-εξόδου και τους καταχωρητές της RAM, που μπορούν να χρησιμοποιηθούν από το πρόγραμμα της εφαρμογής για την αποθήκευση μεταβλητών. Όλες οι διευθύνσεις προσδιορίζονται με σαφή τρόπο, μέσα στις εντολές. Στο αντίστοιχο κώδικα των εντολών, δεσμεύεται χώρος μήκους 7-bits, για τις διευθύνσεις. Αυτό σημαίνει ότι στο PIC μπορεί να πραγματοποιηθεί προσπέλαση μέχρι και 128 διευθύνσεων. Ένα ακόμη τμήμα του βασικού πυρήνα του PIC, που δεν έχει αναφερθεί μέχρι τώρα είναι ο καταχωρητής “w”. Ο καταχωρητής αυτός μπορεί να θεωρηθεί ως ένας συσσωρευτής ή καταχωρητής προσωρινής αποθήκευσης. Στο καταχωρητή “w”, δεν μπορεί να γίνει προσπέλαση άμεσα, αλλά μετακινούνται τα περιεχόμενά τους σε άλλους καταχωρητές, στους οποίους η πρόσβαση είναι άμεση. Κάθε αριθμητική πράξη που επιτελείται στο PIC, χρησιμοποιεί τον καταχωρητή “w”.

▪ Ο καταχωρητής κατάστασης (STATUS)

Ο καταχωρητής STATUS αποτελεί τον βασικό καταχωρητή που χρησιμοποιείται για τον έλεγχο της εκτέλεσης του προγράμματος. Χωρίζεται σε τρία τμήματα. Το πρώτο τμήμα περιέχει τις σημαίες (Flags) ή bits κατάστασης της εκτέλεσης (τις “z”, “dc”, “c”). Το bit “z”, ή η σημαία του μηδενός τίθεται σε λογικό «1» όταν το αποτέλεσμα κάποιας πράξης γίνεται ίσο με το μηδέν. Η σημαία κρατουμένου “c” τίθεται σε λογικό “1” όταν το αποτέλεσμα κάποιας πράξης γίνει μεγαλύτερο από 255 και χρησιμοποιείται για να δηλώσει ότι πρέπει να ενημερωθούν και τα υψηλότερης τάξης bytes που είναι σχετικά με το αποτέλεσμα. Η σημαία δεκαδικού κρατουμένου “dc”, τίθεται σε λογικό “1” όταν τα τέσσερα λιγότερο σημαντικά bits του αποτελέσματος μιας δεκαδικής πράξης, δώσουν μεγαλύτερο από 15. Τα bits που αντιπροσωπεύουν οι σημαίες κατάστασης, μπορούν να διαβαστούν και να εγγραφούν, καθώς και να ενημερώνεται η κατάστασή τους ανάλογα με την εκτέλεση κάθε εντολής. Στους ελεγκτές PIC χαμηλής τάξης (16C5x), τα bits αυτά χρησιμοποιούνται όπως οι καταχωρητές PCLATH στους ελεγκτές μεσαίας και υψηλής τάξης. Σε άλλους τύπους ελεγκτών PIC, τα bits αυτά χρησιμοποιούνται για να δηλώσουν σε ποια ομάδα καταχωρητών μνήμης/εισόδου/εξόδου/ελέγχου, γίνεται προσπέλαση. Πρέπει να σημειωθεί ότι οι βασικοί καταχωρητές (STATUS, INDF, FSR, PCL, PCLATH και

INTCON) βρίσκονται στις ίδιες διευθύνσεις σε κάθε τύπο ελεγκτή PIC και εμφανίζονται στις ίδιες διευθύνσεις κάθε μιας από τις σελίδες μνήμης. Η λογική της σχεδίασης αυτής επιτρέπει χρήση του ίδιου προγράμματος με διαφορετικούς τύπους ελεγκτών της σειράς PIC, χωρίς την ανάγκη αλλαγής των διευθύνσεων των καταχωρητών.

- **Διευθυνσιοδότηση καταχωρητών**

Υπάρχουν τρεις διαφορετικές μέθοδοι για την παραλαβή και τη σχεδίαση δεδομένων στους ελεγκτές PIC. Αυτοί οι μέθοδοι αναφέρονται στο τρόπο διευθυνσιοδότησης της μνήμης, και είναι η άμεση διευθυνσιοδότηση, η διευθυνσιοδότηση καταχωρητών και η έμμεση διευθυνσιοδότηση. Στην άμεση διευθυνσιοδότηση η τιμή που χρησιμοποιείται αποτελεί τμήμα της εντολής. Εκτελώντας πράξεις με σταθερές τιμές, μπορούμε να φορτώσουμε το καταχωρητή “w” με μια συγκεκριμένη τιμή να τον τροποποιήσουμε με κάποια πράξη στην οποία χρησιμοποιείται μια συγκεκριμένη σταθερή τιμή. Για να φορτώσουμε τον καταχωρητή “w” χρησιμοποιούμε συνήθως την εντολή “movf”. Η άμεση διευθυνσιοδότηση των καταχωρητών είναι αρκετά σημαντική αλλά υπάρχουν φορές που απαιτείται έμμεση προσπέλαση στα δεδομένα και αλλαγή της διεύθυνσης, αριθμητικά. Οι καταχωρητές INDF και FSR χρησιμοποιούνται για την δεικτοδομημένη ή την έμμεση διευθυνσιοδότηση δεδομένων. Αυτό σημαίνει ότι τα δεδομένα κρατούνται και είναι προσπελάσιμα ως μητρώα τιμών. Ο καταχωρητής INDF είναι ένας “ψευδοκαταχωρητής” και στην ουσία δεν υπάρχει. Για την προσπέλασή του χρησιμοποιείται ο FSR στον οποίο καταχωρείται η διεύθυνση των δεδομένων που θέλουμε να προσπελάσουμε.

- **Ο μετρητής προγράμματος.**

Οι μεταβολές του μετρητή προγράμματος αποτελούν ίσως , ένα από τα πιο περίπλοκα θέματα που πρέπει να μάθουμε από τους ελεγκτές PIC. Ερευνώντας τις διάφορες οικογένειες διατάξεων της σειράς PIC, θα παρατηρήσουμε ότι η εφαρμογή των εντολών κλήσης (call) και άλματος (goto), φαίνεται να είναι αντιφατική και δυσνόητη. Στην πραγματικότητα οι εντολές αυτές λειτουργούν με βάση την ίδια φιλοσοφία και από τη στιγμή που θα την κατανοήσουμε δεν θα υπάρξει κανένα άλλο πρόβλημα. Σε όλους τους ελεγκτές PIC κάθε εντολή καταλαμβάνει μια λέξη ή διεύθυνση. Το γεγονός αυτό αποτελεί μέρος της φιλοσοφίας σχεδίασης της αρχιτεκτονικής τύπου RISC. Σύμφωνα με τη λογική αυτή, είναι πιθανό, για μια

εντολή κλήσης ή άλματος να μην επαρκούν τα διαθέσιμα bits, του αντίστοιχου πεδίου εντολής, για ολόκληρη τη διεύθυνση της επόμενης θέσης του μετρητή προγράμματος. Τυπικά όλα τα λιγότερο σημαντικά bits, αυτής της διεύθυνσης, τοποθετούνται στο πεδίο διεύθυνσης της λέξης της εντολής. Τα περισσότερα σημαντικά bits, φορτώνονται σε έναν άλλο καταχωρητή που ονομάζεται PCLATH. Έτσι μια εντολή goto ή call λαμβάνει τυπικά τη νέα διεύθυνση από το αντίστοιχο πεδίο της εντολής και από τον καταχωρητή PCLATH. Ο καταχωρητής PCLATH χρησιμοποιείται για την αποθήκευση των περισσότερων σημαντικών bits της διεύθυνσης. Αυτό σημαίνει ότι πριν από την εκτέλεση μιας εντολής υπολογισμένου άλματος (computed goto), πρέπει να είναι έγκυρα τα περισσότερα σημαντικά bits της νέας διεύθυνσης (τα περιεχόμενα του PCLATH)

▪ Περιφερειακά

Κάθε περιφερειακή μονάδα είναι στην πραγματικότητα ένα δομικό τμήμα (block) το οποίο συνδέεται μόνιμα με το τμήμα αποκωδικοποίησης διευθύνσεων, το δίαυλο δεδομένων και ελέγχεται από τα κυκλώματα αποκωδικοποίησης εντολών. Έτσι, για να δημιουργήσει μια νέα σχεδίαση με νέα χαρακτηριστικά, ο σχεδιαστής ολοκληρωμένων τοποθετεί τη νέα διάταξη πάνω στο δίαυλο δεδομένων και χρησιμοποιεί γραμμές από τα τμήματα αποκωδικοποίησης διευθύνσεων και εντολών.

▪ Διακοπές

Οι διακοπές αποτελούν δυστυχώς, μια από τις ιδιότητες του PIC που χρησιμοποιείται λιγότερο από οποιαδήποτε άλλη. Εφαρμόζονται αρκετά εύκολα και προσφέρουν μεγάλη ευελιξία στην ανάπτυξη εφαρμογών του PIC. Είναι ότι κάθε ένας τύπος διακοπής εξυπηρετείται με διαφορετικό τρόπο, αν και η κατανόηση του τρόπου αυτού είναι ανεξάρτητο από το είδος της διακοπής. Σε όλους τους ελεγκτές PIC, ο καταχωρητής INTCON είναι η κύρια πηγή ελέγχου των διακοπών. Το περισσότερο σημαντικό bit του καταχωρητή αυτού (bit γενικής ενεργοποίησης διακοπών ή “Global Interrupt Enable” ή GIE) χρησιμοποιείται για να επιτρέπει να συμβεί οποιαδήποτε διακοπή. Αν το bit αυτό βρίσκεται σε λογικό επίπεδο “0” τότε δεν μπορεί να συμβεί καμία διακοπή. Στην αρχή της εμφάνισης μιας διακοπής, το bit GIE μηδενίζεται εσωτερικά, από σχετικά κυκλώματα του ελεγκτή, πριν αρχίσει να εκτελείται η ρουτίνα διαχείρισης της διακοπής. Τα επόμενα bits του καταχωρητή αυτού, είναι οι

σημαίες ενεργοποίησης των διακοπών του συστήματος. Τα bits αυτά τίθενται σε λογικό επίπεδο “1” για να επιτρέψουν την εξυπηρέτηση των διακοπών, όταν αυτές σηματοδοτηθούν από τις αντίστοιχες λειτουργίες. Είναι εύκολο να αναγνωρίσουμε τα αντίστοιχα bits ενεργοποίησης κάθε διακοπής διότι σύμφωνα με το σύστημα οργάνωσης του PIC, όλες οι ετικέτες των bits αυτών λήγουν στα γράμματα “IE”. Οι μεσαίες ενδείξεις ενεργών διακοπών (των οποίων οι ονομασίες λήγουν στα γράμματα “IF”) μπορούν να τεθούν σε υψηλό λογικό επίπεδο, σηματοδοτώντας έτσι την αντίστοιχη διακοπή, ακόμα και αν αυτή είναι απενεργοποιημένη ή είναι όλες απενεργοποιημένες, με το bit GIE σε λογικό “0”. Είναι επίσης πολύ σημαντικό να σημειωθεί ότι τα bits αυτά πρέπει να μηδενίζονται πριν εκτελεστεί η εντολή επιστροφής από την ρουτίνα εξυπηρέτησης της διακοπής (“retfie”). Πρέπει να σημειωθεί ότι περισσότερες από μια πηγές, μπορούν να έχουν μια αίτηση διακοπής σε αναμονή, οποιαδήποτε χρονική στιγμή. Η διαχείριση κάποιας διακοπής πραγματοποιείται μετά από έλεγχο του αντίστοιχου bit σημαίας ενεργοποίησης (IF).

▪ Παράλληλοι εισόδοι-εξόδοι

Οι ακροδέκτες εισόδου-εξόδου που χρησιμοποιούνται στον PIC είναι στην πραγματικότητα αρκετά απλοί και το αντίστοιχο κύκλωμά τους αποτελείται από μια θύρα ανάγνωσης/εγγραφής και ένα καταχωρητή ελέγχου (TRIS) δεδομένων εξόδου ή οδηγό τριών καταστάσεων. Τόσο ο καταχωρητής TRIS όσο και ο καταχωρητής δεδομένων της θύρας, εμφανίζονται ως τυπικοί καταχωρητές μέσα στο χώρο διευθύνσεων των καταχωρητών των ελεγκτών μεσαίας και υψηλής τάξης. Στους καταχωρητές χαμηλής τάξης, η προσπέλαση του καταχωρητή TRIS επιτυγχάνεται μόνο με την εντολή “TRIS”. Η εγγραφή δεδομένων στη θύρα εξόδου μπορεί να γίνει οποιαδήποτε στιγμή, αλλά κανένα bit δεν εμφανίζεται στην έξοδο αν δεν έχει προηγουμένως μηδενιστεί το αντίστοιχο bit ελέγχου του καταχωρητή TRIS. Εκτός από το τυποποιημένο μοντέλο της θύρας υπάρχει και ένας αριθμός από χαρακτηριστικά που εμπλουτίζουν το σύστημα παράλληλης θύρας εισόδου-εξόδου, ώστε να γίνει πιο ευέλικτο. Ένα από τα πιο συνηθισμένα μέσα εμπλουτισμού που χρησιμοποιείται, είναι το σύστημα ακροδέκτη τύπου ανοιχτού συλλέκτη. Άλλο μέσο εμπλουτισμού των κυκλωμάτων ακροδεκτών, περιλαμβάνει προγραμματιζόμενες αντιστάσεις πρόσδεσης στη τάση τροφοδοσίας (Pull-Up) παρόμοιες με αντιστάσεις των 50KΩ, που συνδέονται στους ακροδέκτες της θύρας PORTB. Έτσι επιτρέπεται η χρήση

διακοπών για την προσγείωση των αντίστοιχων ακροδεκτών, χωρίς επιπλέον εξαρτήματα. Επίσης υπάρχουν άλλοι δυο τύποι εσωτερικών κυκλωμάτων αίτησης διακοπής που μπορούν να προστεθούν στο κύκλωμα των ακροδεκτών με τη μορφή ξεχωριστού ακροδέκτη ενεργοποίησης διακοπής ή διακοπής η οποία συμβαίνει όταν αλλάξει η κατάσταση εισόδου κάποιον ακροδεκτών.

▪ Χρονιστές

Ένα από τα χαρακτηριστικά του PIC που χρησιμοποιείται πολύ συχνά, είναι οι χρονιστές. Κάθε ελεγκτής PIC περιέχει δύο τύπους χρονιστών, που μπορούν να χρησιμοποιηθούν από το σχεδιαστή της εφαρμογής. Ο πρώτος χρονιστής που συμβολίζεται και ως TMR0, είναι ένας απαριθμητής των 8-bits που δέχεται με εναλλακτικά χρονισμό, από δύο διαφορετικές πηγές και μπορεί να σηματοδοτήσει διακοπή στους PIC μεσαίας τάξης ή και μεγαλύτερης. Ο δεύτερος χρονιστής είναι ο χρονιστής επιτήρησης ή Watchdog Timer, ο οποίος ανήκει στα εσωτερικά κυκλώματα του PIC και χρησιμοποιείται κύρια, για να τον επανατοποθετεί, ανιχνεύοντας προβλήματα λογισμικού. Διαφορετικές εκδόσεις του PIC μπορεί να περιέχουν επιπλέον χρονιστές για διάφορες λειτουργίες. Ο TMR0, είναι ένας πλήρως προγραμματιζόμενος απαριθμητής. Η πηγή χρονισμού του μπορεί να είναι το εσωτερικό σήμα χρονισμού των κύκλων των εντολών ή ένας εξωτερικός παλμός συγχρονισμού. Η επιλογή του είδους της πηγής χρονισμού του TMR0, επιτυγχάνεται με τη χρήση του bit RTS του καταχωρητή OPTION. Η πιο προφανής χρήση του μετρητή είναι αυτή κατά την οποία μετريέται ένα συγκεκριμένο χρονικό διάστημα, μετά την λήξη του οποίου, πραγματοποιείται διακοπή του προγράμματος για να επιτελεστεί μια καθορισμένη ενέργεια. Ένα τέτοιο παράδειγμα μπορεί να αποτελέσει η χρήση ενός χρονιστή, ο οποίος βρίσκεται σε αναμονή μέχρι να συμβεί κάποιο γεγονός και αμέσως μετά να πραγματοποιηθεί η σχετική επεξεργασία. Ο χρονιστής μπορεί να χρησιμοποιηθεί και σα χρονόμετρο για τη μέτρηση του χρόνου συγκεκριμένων γεγονότων και να αντιδρά ανάλογα με την τιμή που βρίσκεται στο καταχωρητή του χρονιστή. Μια ακόμα χρήση του χρονιστή μπορεί να είναι και ο έλεγχος σφάλματος χρονικής υπέρβασης. Ο χρονιστής επιτήρησης (γνωστός και ως WDT) χρονίζεται από έναν εσωτερικό ταλαντωτή μέσα στο ολοκληρωμένο του PIC. Ο χρονιστής επιτήρησης ενεργοποιείται με κατάλληλη τοποθέτηση των προγραμματιζόμενων εντολών του PIC και δεν μπορεί να απενεργοποιηθεί από το

πρόγραμμα. Η επανατοποθέτησή του μπορεί να πραγματοποιηθεί μόνο με την εκτέλεση της εντολής “clrwdt”. Η εντολή αυτή έχει ως αποτέλεσμα να μηδενιστεί τόσο ο μετρητής WDT, όσο και ο προμετρητής (αν έχει επιλεγεί για το WDT) και να αρχίσει ξανά η απαρίθμηση από το μηδέν. Ο χρονιστής WDT χρησιμοποιείται επίσης και για την ενεργοποίηση του ελεγκτή μετά από έναν ανενεργό κύκλο (sleep). Με βάση αυτό το τύπο λειτουργίας, η εφαρμογή μπορεί να τεθεί σε ανενεργό κύκλο και να ενεργοποιείται μόνο για να ελέγξει το περιβάλλον σύστημα, να εκτελέσει κάποιες ενέργειες ανάλογα με αυτό και στη συνέχεια να επιστρέψει στην ανενεργή κατάσταση μέχρι τη στιγμή που θα χρειαστεί ξανά να γίνει κάποιος έλεγχος.

2.5.4 Ρεπερτόριο εντολών του PIC

Οι εντολές των ελεγκτών PIC μπορούν να χωριστούν σε τέσσερις κατηγορίες. Η πρώτη κατηγορία αποτελείται από τις “αριθμητικές εντολές”, οι οποίες χρησιμοποιούνται για την ανάγνωση ή την εγγραφή των περιεχομένων των καταχωρητών μνήμης. Αυτές περιλαμβάνουν τη πράξη της πρόσθεσης και της αφαίρεσης από τα περιεχόμενα των καταχωρητών καθώς και πράξεις αύξησης ή μείωσης των τιμών τους και πράξεις σε επίπεδο bit. Η κατηγορία των αριθμητικών πράξεων μπορεί να χωριστεί σε δύο υποκατηγορίες. Η πρώτη περιλαμβάνει τις “αριθμητικές εντολές καταχώρησης” (όπου χρησιμοποιούνται μόνο τα περιεχόμενα των καταχωρητών) ενώ η δεύτερη περιλαμβάνει τις “άμεσες αριθμητικές εντολές”. Στην επόμενη λειτουργική κατηγορία εντολών, ανήκουν οι εντολές “ελέγχου εκτέλεσης”. Αυτές τις αποτελούν οι εντολές άλματος(goto), κλήσης(call) και οι εντολές επιστροφής(return) από κάποια ρουτίνα καθώς επίσης και οι εντολές διακλάδωσης υπό όρους. Στην επόμενη κατηγορία οι εντολές “ελέγχου του μικροεπεξεργαστή”. Οι εντολές αυτές επηρεάζουν βασικά τη λειτουργία του επεξεργαστή και τα κυκλώματα που συσχετίζονται με αυτόν. Στην κατηγορία αυτή, η εντολή απενεργοποίησης, ή “sleep” (που τοποθετεί τον PIC σε προσωρινή αδρανή κατάσταση) και η εντολή μηδενισμού του χρονιστή επιτήρησης ή WDT (με την οποία επανατοποθετείται ο χρονιστής αυτός), αποτελούν τις κύριες εντολές. Η κατηγορία αυτή περιλαμβάνει επίσης, τις εντολές “option” και “tris”, που χρησιμοποιούνται για την άμεση εγγραφή δεδομένων στους καταχωρητές ελέγχου κυκλωμάτων, που ανήκουν στην ομάδα καταχωρητών “Bank1”. Η τελευταία κατηγορία αποτελείται από

τις εντολές “χειρισμού των bit των καταχωρητών” (τοποθέτηση ή μηδενισμός bit). Οι εντολές αυτές στα πλαίσια της αρχιτεκτονικής του PIC είναι μοναδικές. Με τις εντολές αυτές ελέγχουμε άμεσα, κάθε ένα ξεχωριστό bit των καταχωρητών. Η πιο προφανής χρήση των εντολών αυτών, είναι ο άμεσος έλεγχος των κυκλωμάτων του μικροελεγκτή.

2.5.5 Software

Προκειμένου να αναπτυχθεί το λογισμικό μας και να οργανώσουμε τα αρχεία μας θα πρέπει να χρησιμοποιήσετε ένα ολοκληρωμένο περιβάλλον ανάπτυξης. Το υπ' αριθμόν ένα IDE που χρησιμοποιείται με μικροελεγκτές PIC είναι το MPLAB IDE από το Microchip Technology. Το MPLAB IDE είναι δωρεάν και εύκολο στη χρήση. Το MPLAB IDE ® X είναι ένα πρόγραμμα λογισμικού που τρέχει σε PC (Windows ®, Mac OS ®, Linux ®) για την ανάπτυξη εφαρμογών για μικροελεγκτές Microchip και ψηφιακών ελεγκτών σήματος. Λέγεται ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE), διότι παρέχει ένα ενιαίο ολοκληρωμένο «περιβάλλον» για την ανάπτυξη κώδικα για embedded μικροελεγκτές. Το MPLAB ® X ολοκληρωμένο περιβάλλον ανάπτυξης φέρνει πολλές αλλαγές στην αλυσίδα εργαλείων ανάπτυξης του μικροελεγκτή PIC. Σε αντίθεση με προηγούμενες εκδόσεις του MPLAB ® που αναπτύχθηκαν ολοκληρωτικά στο σπίτι, το MPLAB ® X βασίζεται στο open source Net Beans IDE από την Oracle. Ακολουθώντας αυτό το μονοπάτι, μας επέτρεψε να προσθέσουμε πολλά χαρακτηριστικά που απαιτούνται συχνά, πολύ γρήγορα και εύκολα, ενώ επίσης μας παρέχει μια πολύ πιο επεκτάσιμη αρχιτεκτονική για να μας φέρει ακόμα περισσότερες νέες δυνατότητες στο μέλλον.

MPLAB ® X IDE Χαρακτηριστικά:

- Παρέχει ένα νέο Γράφημα κλήσης για την πλοήγηση πολύπλοκου κώδικα
- Υποστηρίζει πολλαπλές διαμορφώσεις στο πλαίσιο των έργων μας
- Υποστήριξη πολλαπλών εκδόσεων του ίδιου compiler
- Υποστήριξη για πολλαπλά εργαλεία εντοπισμού σφαλμάτων του ίδιου τύπου
- Υποστηρίζει Live Parsing
- Υποστηρίζει υπερσυνδέσεις για γρήγορη πλοήγηση σε δηλώσεις
- Υποστηρίζει πρότυπα Live Code
- Υποστηρίζει τη δυνατότητα να εισέλθουν πρότυπα File Code με κεφαλίδες άδειας ή πρότυπο κώδικα

- Με το MPLAB ® X μπορούμε να παρακολουθούμε τις αλλαγές στο δικό μας σύστημα, χρησιμοποιώντας την τοπική ιστορία
- Μέσα σε MPLAB ® X, ο χρήστης μπορεί να διαμορφώσει το δικό του κώδικα Format Style

▪ Γλώσσες προγραμματισμού

Οι μικροελεγκτές PIC μπορούν να προγραμματιστούν σε Assembly, C ή σε συνδυασμό αυτών των δύο. Μπορούν να χρησιμοποιηθούν και άλλες γλώσσες προγραμματισμού υψηλού επιπέδου, αλλά το ενσωματωμένο λογισμικό σύστημα είναι κυρίως γραμμένο σε C. Τα επόμενα 3 παραδείγματα αποδεικνύουν τα στυλ προγραμματισμού.

Example 1 – Assembly

MAIN

clrf PORTB ;Clear PORTB output latches

bsf STATUS,RP0 ;Switch to bank 1

movlw b'11110000' ;Load value to make lower 4 bits outputs

movwf TRISB ;Move value to TRISB

bcf STATUS,RP0 ;Switch to bank 0

LOOP

bsf PORTB,0 ;Turn on LED on RB0

call DELAY ;Call delay routine

bcf PORTB,0 ;Turn off LED on RB0

call DELAY ;Call delay routine

goto LOOP ;Repeat main loop

DELAY

decfsz COUNTERL ;Decrement COUNTERL

goto DELAY ;If not zero, keep decrementing COUNTERL

decfsz COUNTERH ;Decrement COUNTERH

goto DELAY ;If not zero, decrement COUNTERL again

return

END

Example 2 – Assembly and C

main()

{

short first_pass = TRUE;

//----- Set up port direction reg's -----

#asm

movlw 0 // Set port b to outputs

tris port_b

clrf port_b

movlw 0xff // Set port a to inputs

tris port_a

#endasm

//----- Wait for powerup, Initilize LCD -----

```

delay_ms(200);

init_lcd();

//----- Write a startup message -----

msg_1();

//----- Write status message -----

msg_2();

...

```

Example 3 – C

```

void main() {

    U8 i = 0; // General purpose loop var.

    U16 num ; // General purpose number var.

    U8 row = 0; // Current display row.

    U16 blinkc = 0; // LED blinker counter.

    U16 blink_onoff = 1; // LED state.

    U8 bcd_h,bcd_m,bcd_s; // BCD numbers.

    init_ports();

    init();

    enable_interrupts();

    ROW_RESET;

    ///////////////////////////////////

```

```

for (;;) { // infinite loop

// FLASH LED @ RA3

if (++blinkc>500) { // time to change state ?

blinkc=0;

...

```

Η προτεινόμενη γραφή κώδικα είναι αποκλειστικά σε C διότι είναι γρηγορότερο και ευκολότερο από το να γράψουμε τον κώδικά μας με Assembly ή σε συνδυασμό των δύο γλωσσών.

▪ MPLAB

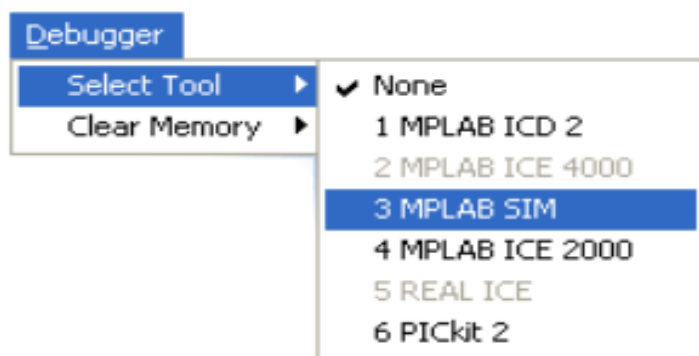
Το MPLAB είναι ένα πακέτο προγραμμάτων των Windows που επιτρέπει την εύκολη γραφή του προγράμματος, καθώς και την εύκολη ανάπτυξη του προγράμματος. Το καλύτερο είναι να το περιγράψουμε ως περιβάλλον ανάπτυξης για μια τυπική γλώσσα πρόγραμμα που έχει σχεδιαστεί για τον προγραμματισμό του υπολογιστή. Το MPLAB απλοποιεί τεχνικά ορισμένες λειτουργίες που αποτελούνται από πολλές παραμέτρους, οι οποίες, εμφανίστηκαν μέχρι το περιβάλλον IDE και εκτελέστηκαν από τη γραμμή εντολών. Ωστόσο, τα γούστα είναι διαφορετικά και υπάρχουν κάποιοι προγραμματιστές που προτιμούν πρότυπο εκδότες και μεταφραστές της γραμμής εντολών. Κάθε πρόγραμμα γραμμένο σε MPLAB είναι σαφές, αλλά υπάρχει και τεκμηρίωση βοήθειας, σε ορισμένες περιπτώσεις.

Προσομοιωτής

Ο simulator είναι ένα μέρος του MPLAB περιβάλλον, το οποίο παρέχει καλύτερη εικόνα για τη λειτουργία του μικροελεγκτή. Σε γενικές γραμμές, μια προσομοίωση είναι μια προσπάθεια να διαμορφώσει μια πραγματική ζωή ή υποθετική κατάσταση, έτσι ώστε να μπορούν να μελετηθούν για να δούμε πώς λειτουργεί το σύστημα. Με τη βοήθεια του προσομοιωτή, είναι επίσης δυνατό να ελέγχουμε τις τρέχουσες τιμές των μεταβλητών, τα μητρώα καθώς επίσης και τις καρφίτσες λιμάνι. Ένας

προσομοιωτής δεν είναι της ίδιας σημασίας για όλα τα προγράμματα . Εάν ένα πρόγραμμα είναι απλούστερο, η προσομοίωση δεν έχει μεγάλη σημασία, διότι η ρύθμιση της καρφίτσας B με λογικό ένα (1) δεν είναι καθόλου περίπλοκη . Ωστόσο, σε πιο πολύπλοκα προγράμματα που περιέχουν χρονόμετρα , διαφορετικές συνθήκες και αιτήματα (κυρίως μαθηματικές πράξεις), ο προσομοιωτής μπορεί να είναι μεγάλης χρήσης . Όπως το ίδιο το όνομα δείχνει , μια προσομοίωση σημαίνει να προσομοιώσει τη λειτουργία του μικροελεγκτή . Όπως και ο μικροελεγκτής, ένας προσομοιωτής εκτελεί τις οδηγίες τη μια μετά την άλλη (ανά γραμμή) και ενημερώνει συνεχώς την κατάσταση όλων των μητρώων . Με αυτόν τον τρόπο , ο χρήστης παρακολουθεί απλώς την εκτέλεση του προγράμματος . Στο τέλος της γραφής του προγράμματος , ο χρήστης θα πρέπει πρώτα να το δοκιμάσει στον προσομοιωτή πριν την εκτέλεση σε ένα πραγματικό περιβάλλον . Δυστυχώς , αυτό είναι ένα από τα πολλά καλά πράγματα που αγνοείται από τον προγραμματιστή , λόγω της έλλειψης των προσομοιωτών υψηλής ποιότητας.

Ο Simulator ενεργοποιείται κάνοντας κλικ στο Debugger> SELECT TOOL> MPLAB SIM, όπως φαίνεται στο σχήμα.



Εικόνα 2.26: ενεργοποίηση του simulator

Ως αποτέλεσμα, να εμφανίζονται μόνο εικονίδια που σχετίζονται με τον προσομοιωτή:



Ξεκινά την εκτέλεση του προγράμματος σε πλήρη ταχύτητα. Ο εξομοιωτής εκτελεί το πρόγραμμα σε πλήρη (κανονική) ταχύτητα μέχρι να σταματήσει κάνοντας κλικ στο παρακάτω εικονίδιο.



Διακόπτει προσωρινά την εκτέλεση του προγράμματος. Το πρόγραμμα μπορεί να συνεχίσει την εκτέλεση βήμα προς βήμα ή σε πλήρη ταχύτητα και πάλι.



Εκτέλεση του προγράμματος ξεκινά από το προαιρετικό ταχύτητα. Η ταχύτητα της εκτέλεσης βρίσκεται σε διάλογο Debugger / Settings / Ψυχαγωγία / Realtime ενημερώσεις.



Ξεκινά βήμα-βήμα την εκτέλεση του προγράμματος. Οδηγίες εκτελούνται μία μετά την άλλη. Επιπλέον, κάνοντας κλικ σε αυτό το εικονίδιο μπορούμε να μετακινηθούμε σε υπορουτίνες και μακροεντολές.

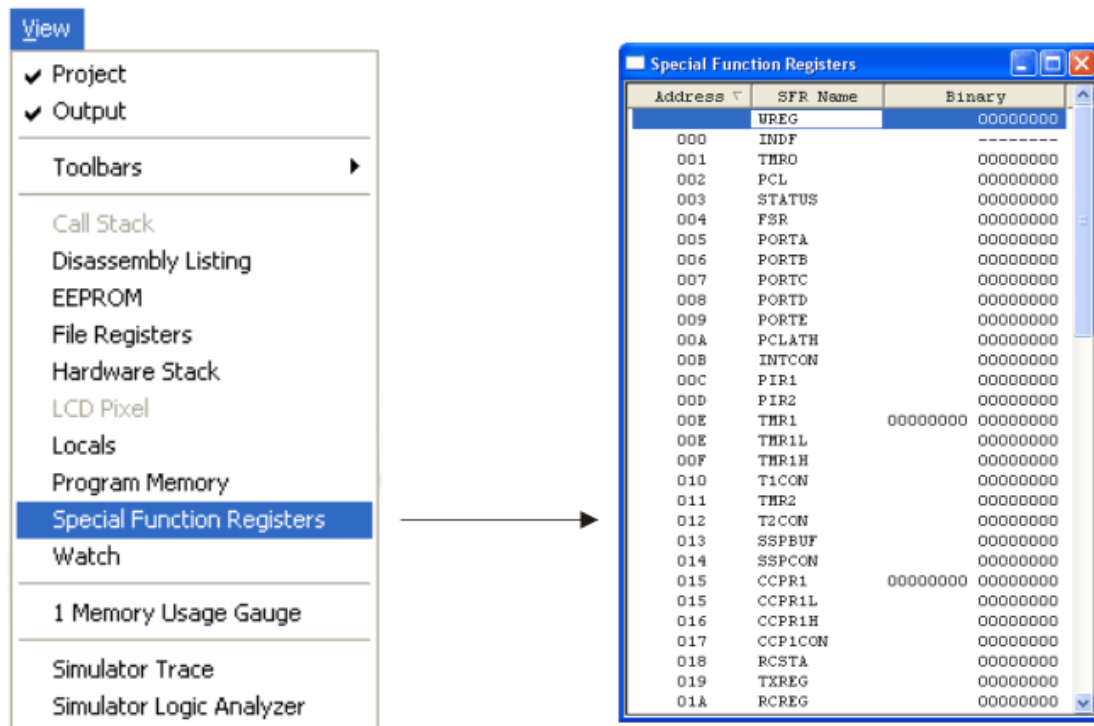


Η εικόνα έχει την ίδια λειτουργία με την προηγούμενη εκτός από το ότι έχει την ικανότητα να μπει σε υπορουτίνες.



Επαναφέρει τον μικροελεγκτή. Πατώντας το εικονίδιο αυτό, ο μετρητής του προγράμματος είναι τοποθετημένος κατά την έναρξη του προγράμματος και η προσομοίωση μπορεί να ξεκινήσει.

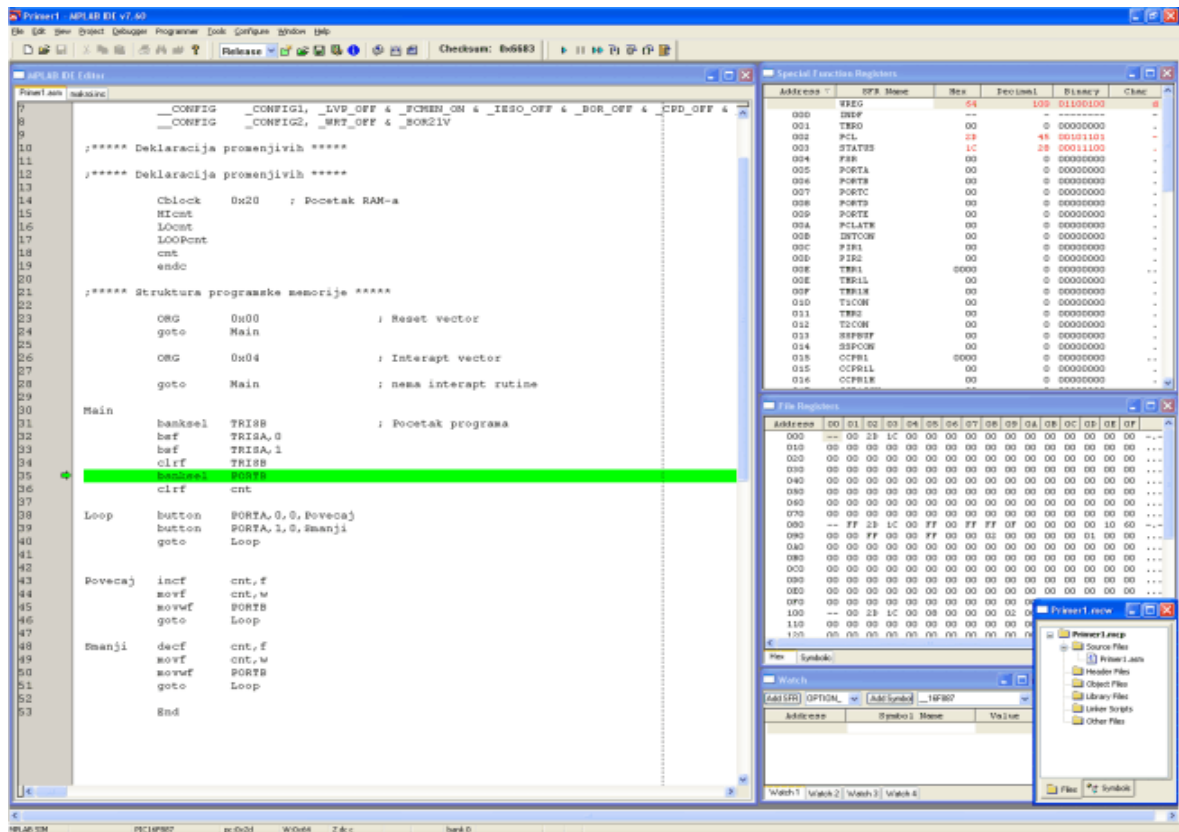
Παρόμοια με το πραγματικό περιβάλλον, το πρώτο πράγμα που πρέπει να γίνει είναι να επαναφέρουμε το μικροελεγκτή χρησιμοποιώντας το πρόγραμμα εντοπισμού σφαλμάτων **επιλογή> ΕΠΑΝΑΦΟΡΑ** ή κάνοντας κλικ στο εικονίδιο επαναφοράς. Ως συνέπεια αυτού, μια πράσινη γραμμή είναι τοποθετημένη στην αρχή του προγράμματος και ο μετρητής προγράμματος PCL εκκαθαρίζεται στο μηδέν.



Εικόνα 2.27: special function registers

Εκτός από το SFRs, είναι καλό να έχουμε μια εικόνα σε μητρώα αρχείων. Ένα παράθυρο που περιέχει αυτά που εμφανίζεται, πατώντας την επιλογή VIEW-> FILE REGISTERS.

Αν το πρόγραμμα περιλαμβάνει μεταβλητές, είναι καλό να παρακολουθούνται οι τιμές τους. Κάθε μεταβλητή έχει εκχωρηθεί σε ένα παράθυρο (ρολόι των Windows) κάνοντας κλικ στην επιλογή VIEW-> WATCH.



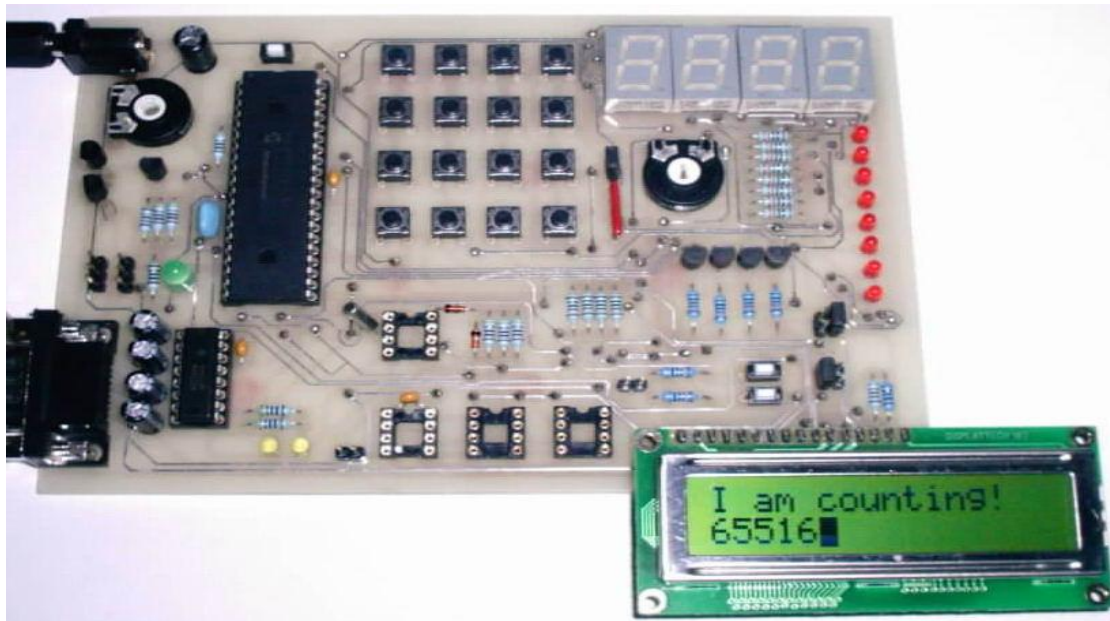
Εικόνα 2.28: εικόνα προγράμματος

Μετά από όλες τις μεταβλητές και τα μητρώα των τόκων γίνονται διαθέσιμα στον προσομοιωτή χώρο εργασίας και η διαδικασία της προσομοίωσης μπορεί να ξεκινήσει. Η επόμενη εντολή μπορεί να είναι είτε βήμα βήμα μέσα ή πάνω, ανάλογα με το αν θέλουμε πάμε σε υπορουτίνα ή όχι. Οι ίδιες οδηγίες μπορεί να καθοριστούν με τη χρήση του πληκτρολογίου-κουμπιά <F7> ή <F8> (γενικά, όλες οι σημαντικές οδηγίες έχουν τα αντίστοιχα πλήκτρα στο πληκτρολόγιο).

2.5.7 PIC EXAMPLE

Παράδειγμα 1

Basic LCD test



Εικόνα 2.29: Basic LCD test

CODE

```
Define CONF_WORD = 0x3f72
```

```
Define CLOCK_FREQUENCY = 12
```

```
AllDigital
```

```
Define LCD_BITS = 8
```

```
Define LCD_DREG = PORTD
```

```
Define LCD_DBIT = 0
```

```
Define LCD_RSREG = PORTE
```

```
Define LCD_RSBIT = 0
```

```
Define LCD_RWREG = PORTE
```

```
Define LCD_RWBIT = 1
```

```
Define LCD_EREG = PORTE
```

```
Define LCD_EBIT = 2
```

```
Define LCD_READ_BUSY_FLAG = 1
```

```
Lcdinit 3
```

```
Dim a As Word
```

```
a = 65535
```

```
loop:
```

```
    Lcdcmdout LcdClear
```

```
    Lcdout "I am counting!"
```

```
    Lcdcmdout LcdLine2Home
```

```
    Lcdout #a
```

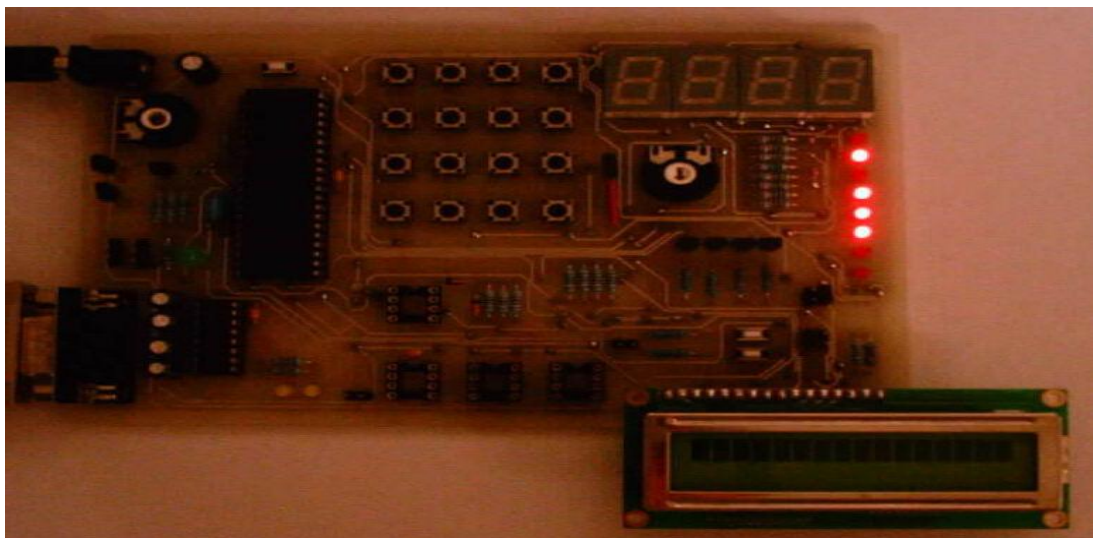
```
    a = a - 1
```

```
    WaitMs 1000
```

```
Goto loop
```

Παράδειγμα 2

Playing with LEDs



Εικόνα 2.30: Playing with LEDs

CODE

```
Define CONF_WORD = 0x3f72
```

```
Define CLOCK_FREQUENCY = 12
```

```
AllDigital
```

```
Dim i As Byte
```

```
TRISB = 0
```

```
loop:
```

```
For i = 0 To 255
```

```
PORTB = i
```

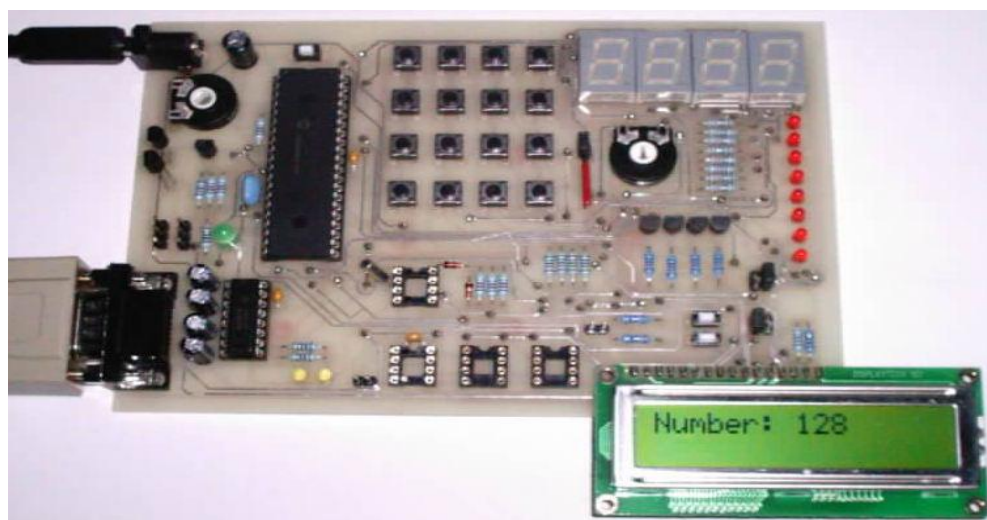
```
WaitMs 100
```

```
Next i
```

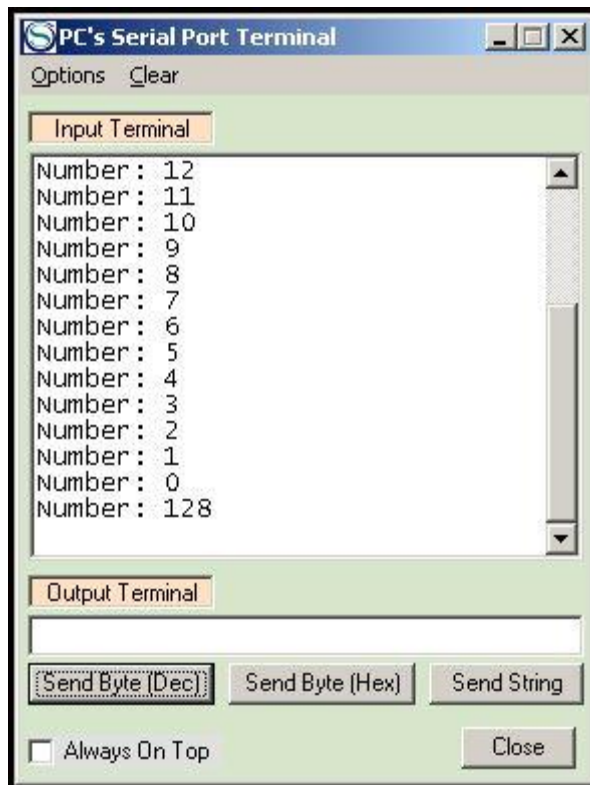
```
Goto loop
```

Παράδειγμα 3

RS-232 communication with PC



Εικόνα 2.31: RS-232 communication with PC



Εικόνα 2.32: PC's serial port terminal

CODE

```
Define CONF_WORD = 0x3f72
```

```
Define CLOCK_FREQUENCY = 12
```

```
AllDigital
```

```
Define LCD_BITS = 8
```

```
Define LCD_DREG = PORTD
```

```
Define LCD_DBIT = 0
```

```
Define LCD_RSREG = PORTE
```

```
Define LCD_RSBIT = 0
```

```
Define LCD_RWREG = PORTE  
Define LCD_RWBIT = 1  
Define LCD_EREG = PORTE  
Define LCD_EBIT = 2  
Define LCD_READ_BUSY_FLAG = 1  
  
Lcdinit
```

```
Dim i As Byte
```

```
Hseropen 19200
```

```
WaitMs 1000
```

```
For i = 20 To 0 Step -1
```

```
    Hserout "Number: ", #i, CrLf
```

```
    Lcdcmdout LcdClear
```

```
    Lcdout "Number: ", #i
```

```
    WaitMs 1000
```

```
Next i
```

```
loop:
```

```
    Hserin i
```

```
    Hserout "Number: ", #i, CrLf
```

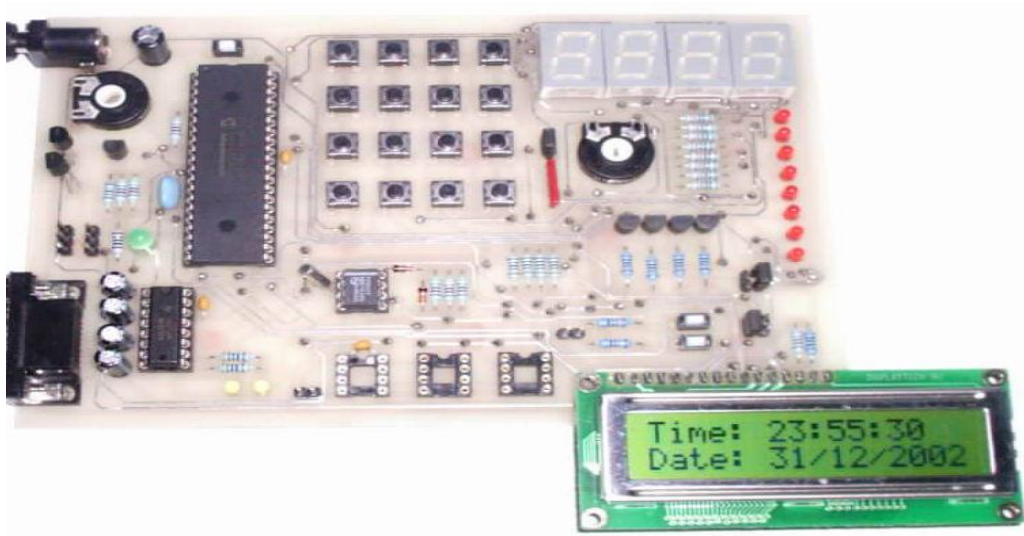
```
    Lcdcmdout LcdClear
```

```
    Lcdout "Number: ", #i
```

```
Goto loop
```


Παράδειγμα 4

I2C communication with PCF8583 real time clock



Εικόνα 2.33: I2C communication with PCF8583 real time clock

CODE

```
Define CONF_WORD = 0x3f72
```

```
Define CLOCK_FREQUENCY = 12
```

```
AllDigital
```

```
Define LCD_BITS = 8
```

```
Define LCD_DREG = PORTD
```

```
Define LCD_DBIT = 0
```

```
Define LCD_RSREG = PORTE
```

```
Define LCD_RSBIT = 0
```

```
Define LCD_RWREG = PORTE
```

```
Define LCD_RWBIT = 1
```

```
Define LCD_EREG = PORTE
```

```
Define LCD_EBIT = 2
```

```
Define LCD_READ_BUSY_FLAG = 1
```

```
Lcdinit
```

```
Symbol scl = PORTC.3
```

```
Symbol sda = PORTC.4
```

```
Dim addr As Byte
```

```
Dim data As Byte
```

```
Dim sec As Byte
```

```
Dim min As Byte
```

```
Dim hour As Byte
```

```
Dim day As Byte
```

```
Dim month As Byte
```

```
Dim year As Byte
```

```
Dim d_hi As Byte
```

```
Dim d_lo As Byte
```

```
WaitMs 1000
```

```
addr = 2
```

```
I2CWrite sda, scl, 0xa2, addr, 0
```

```
addr = 3
```

```
I2CWrite sda, scl, 0xa2, addr, 0x55
```

```
addr = 4
```

```
I2CWrite sda, scl, 0xa2, addr, 0x23
```

```
addr = 5
```

```
I2CWrite sda, scl, 0xa2, addr, 0xb1
```

```
addr = 6
```

```
I2CWrite sda, scl, 0xa2, addr, 0x12
```

```
loop:
```

```
addr = 2
```

```
I2CRead sda, scl, 0xa2, addr, sec
```

```
addr = 3
```

```
I2CRead sda, scl, 0xa2, addr, min
```

```
addr = 4
```

```
I2CRead sda, scl, 0xa2, addr, hour
```

```
addr = 5
```

```
I2CRead sda, scl, 0xa2, addr, day
```

```
year = 0
```

```
year.0 = day.6
```

```
year.1 = day.7
```

```
day.6 = 0
```

```
day.7 = 0
```

```
addr = 6
```

```
I2CRead sda, scl, 0xa2, addr, month
```

```
month.5 = 0
```

```
month.6 = 0
```

```
month.7 = 0
```

```
Lcdcmdout LcdClear
```

```

Lcdout "Time: "

data = hour

Gosub print

Lcdout ":"

data = min

Gosub print

Lcdout ":"

data = sec

Gosub print

Lcdcmdout LcdLine2Home

Lcdout "Date: "

data = day

Gosub print

Lcdout "/"

data = month

Gosub print

Lcdout "/200", #year


WaitMs 200

Goto loop

End


print:

d_hi = data / 16

d_lo = data Mod 16

If d_hi < 10 Then

```

```

Lcdout #d_hi

Else

Lcdout "X"

Endif

If d_lo < 10 Then

Lcdout #d_lo

Else

Lcdout "X"

Endif

Return

```

ΑΝΑΦΟΡΕΣ:

- [1] http://en.wikipedia.org/wiki/PIC_microcontroller
- [2]<http://roboengineers.blogspot.gr/2008/04/history-of-pic-microcontrollers.html>
- [3] <http://www.mikroe.com/chapters/view/1/>
- [4]<http://www.mikroe.com/chapters/view/11/appendix-a-programming-a-microcontroller/>
- [5] <http://whatis.techtarget.com/definition/PIC-microcontrollers>
- [6] http://wiki.answers.com/Q/What_are_the_applications_of_pic_microcontroller
- [7] <http://www.engineersgarage.com/articles/pic-microcontroller-tutorial>
- [8] <http://www.electronics-lab.com/pic-in-greek/>
- [9] <http://www.best-microcontroller-projects.com/pic-microcontroller.html>
- [10]<http://www.robometricschool.com/2013/02/introducing-pic-microcontroller-history.html>
- [11]<http://ezinearticles.com/?Functions-and-Advantages-of-Microchip-PIC-Microcontroller&id=6881185>

- [12] <http://www.elprocus.com/top-pic-microcontroller-projects-ideas/>
- [13] <http://www.engineersgarage.com/articles/pic-microcontroller-tutorial?page=5>
- [14] PIC microcontrollers, an introduction to microelectronics, Third Edition, Martin Bates.
- [15] PIC microcontrollers: an introduction to software & hardware interfacing, Han-Way Haung
- [16] Προγραμματίζοντας τον μικροελεγκτή PIC, MYKE PREDKO

3.1 Σχεδιασμός διδασκαλίας εκπαιδευτικής ρομποτικής με τον ARDUINO & PIC

3.1.1 Η διδασκαλία της εκπαιδευτικής ρομποτικής- στόχοι αντικείμενο

Γιατί η Ρομποτική στην Εκπαίδευση

Α) Τα παιδιά όταν σχεδιάζουν, κατασκευάζουν και προγραμματίζουν ρομπότ έχουν την ευκαιρία να μάθουν παίζοντας και να αναπτύξουν δεξιότητες. Η ρομποτική αφενός, είναι μία διασκεδαστική και ενδιαφέρουσα δραστηριότητα που δίνει τη δυνατότητα στο μαθητή να εμπλακεί με τη δράση, αφετέρου μπορεί να χρησιμοποιηθεί σε όλες τις βαθμίδες εκπαίδευσης για τη **διδασκαλία διαφόρων εννοιών**, κυρίως, από τις Φυσικές Επιστήμες και άλλα γνωστικά αντικείμενα.

- Φυσική (μελέτη της κίνησης, μελέτη της επίδρασης της τριβής, μελέτη της σχέσης των δυνάμεων, μεταφορά ενέργειας κ.α)
- Μαθηματικά και Γεωμετρία (αναλογίες, μέτρηση αποστάσεων, κατανόηση βασικών γεωμετρικών ιδιοτήτων όπως η περίμετρος κ.α)
- Μηχανική (κατασκευή, έλεγχος και αξιολόγηση μηχανικών λύσεων κ.α)
- Τεχνολογία (τεχνολογικός αλφαριθμητισμός κ.α)
- Ιστορία (πχ. με την κατασκευή ενός ρομπότ καταπέλτη - του Αρχιμήδη - τα παιδιά έχουν την ευκαιρία να γνωρίσουν την ανάπτυξη της τεχνολογίας εκείνης της εποχής καθώς και το έργο και την προσωπικότητα του Αρχιμήδη κ.α)
- Ο συνδυασμός εννοιών από διαφορετικές, γνωστικές περιοχές (τεχνολογία, τέχνη, περιβάλλον, κοινωνία, μαθηματικά, φυσικές επιστήμες) με διαθεματικά project (συνθετικές εργασίες)

κ.λπ.

Η εκπαιδευτική Ρομποτική έχει θετικές επιπτώσεις εκτός από το γνωστικό τομέα και στο συναισθηματικό (αυτοεκτίμηση, αυτοπεποίθηση) και κοινωνικό(κοινωνικοποίηση, απομυθοποίηση).

Επιπλέον, με τη βοήθεια της ρομποτικής στη διδασκαλία του ο εκπαιδευτικός μπορεί να επικεντρωθεί στην ανάπτυξη και άλλων κρίσιμων δεξιοτήτων του 21ου αιώνα:

- Ομαδική εργασία
 - επίλυση προβλημάτων (ανάλυση, σχεδίαση, υλοποίηση, δοκιμή και πειραματισμός, αξιολόγηση)
 - καινοτομία
 - διαχείριση έργου (διαχείριση χρόνου, κατανομή έργου και πόρων κ.α)
 - προγραμματισμός
 - δεξιότητες επικοινωνίας
 - πολύτιμες νοητικές δεξιότητες (αναλυτική και συνθετική σκέψη, δημιουργικότητα, κριτική σκέψη κ.α)
- κ.λπ.

Το όραμα της ρομποτικής είναι όλοι οι μαθητές να αναπτύξουν αυτές τις δεξιότητες, οι οποίες στα πλαίσια της παγκοσμιοποίησης να αποτελούν επιτακτική ανάγκη για την προετοιμασία πολιτών του κόσμου που θα μπορούν να συνεισφέρουν θετικά σε παγκόσμια κλίμακα.

B) Η ρομποτική αλλάζει τον παραδοσιακό χαρακτήρα της διδασκαλίας

Σύγχρονες θεωρήσεις για την μάθηση

Χαρακτηριστικά της παραδοσιακής θεώρησης για τη μάθηση	Χαρακτηριστικά της σύγχρονης θεώρησης για τη μάθηση
Η γνώση προκύπτει από την παθητική αποδοχή	Η γνώση ανακαλύπτεται και κατακτάται από τον μαθητή
Η μάθηση είναι μια μοναχική πορεία	Η μάθηση είναι κοινωνική διαδικασία
Η μάθηση είναι μονοδιάστατη και ακολουθεί έναν μόνο δρόμο	Η μάθηση είναι σφαιρική(ολιστική) και οδηγούμαστε σε αυτήν από διάφορα μονοπάτια
Η μάθηση εστιάζεται στις γνωστικές ανεπάρκειες των μαθητών	Η μάθηση εστιάζεται στα ενδιαφέροντα και τις ικανότητες των

	μαθητών
Πηγή πληροφόρησης είναι μόνο ο δάσκαλος και το βιβλίο	Η πληροφόρηση προέρχεται από ποικίλες, διαφορετικές πηγές
Η γνώση αναπαράγεται	Η γνώση παράγεται από τους μαθητές
Ο δάσκαλος μεταφέρει γνώση	Ο δάσκαλος διευκολύνει την σύνδεση της γνώσης με την πραγματικότητα

Πίνακας 4: χαρακτηριστικά για την παραδοσιακή και σύγχρονη θεωρία μάθησης

- Η εκπαιδευτική ρομποτική συνδυάζει τη μάθηση με το παιχνίδι και έτσι μετατρέπει την εκπαίδευση σε μια διασκεδαστική δραστηριότητα- είναι γνωστό άλλωστε πως η μάθηση επιτυγχάνεται ευκολότερα, ταχύτερα και ουσιαστικότερα όταν συνδυάζεται με το παιχνίδι. Η πτυχή του παιχνιδιού που εμπεριέχουν τα προγραμματιζόμενα ρομπότ αποτελεί ένα σημαντικό παράγοντα θετικού κινήτρου και παρώθησης κυρίως στην πρωτοβάθμια εκπαίδευση.
- Ευνοεί την ανάπτυξη ερευνητικού ενδιαφέροντος. Η εκπαιδευτική ρομποτική δίνει τη δυνατότητα στα παιδιά να δράσουν ως επιστήμονες – εφευρέτες και να ανακαλύψουν δικές τους καινοτόμες ιδέες και λύσεις.
- Εμπλέκει ενεργά τους μαθητές στη μάθησή τους με την επίλυση αυθεντικών προβλημάτων.
- Υποστηρίζει την διερευνητική μάθηση και ενισχύει την διερευνητική στάση των μαθητών.
- Δίνει κίνητρα στους μαθητές να μελετήσουν την επιστήμη και την τεχνολογία.
- Η ρομποτική εμπλέκει τους μαθητές σε καταστάσεις που απαιτούν από αυτούς να εφαρμόσουν τα μαθηματικά και την επιστήμη και όχι

απλά να τα μελετήσουν. Διότι η κατανόηση είναι κάτι περισσότερο από μάθηση, είναι μάθηση και γνώση πώς να εφαρμόσεις αυτό που γνωρίζεις στο πλαίσιο αυτό.

- Η εκπαιδευτική ρομποτική παρέχει στους μαθητές ευκαιρίες επίλυσης προβλημάτων με προσωπικό νόημα για τους ίδιους μέσω χειρισμού και κατασκευής πραγματικών ή ιδεατών αντικειμένων. Η γνώση που προκύπτει από προβληματικές καταστάσεις δίνει την ευκαιρία στους μαθητές να αναπτύξουν μια ισχυρή εννοιολογική βάση για την ανακατασκευή των γνώσεών τους σε μεταγενέστερο χρόνο.
- Επιτρέπει την ελεύθερη έκφραση και την ανάπτυξη της δημιουργικότητας και φαντασίας.
- Μέσα από την κατασκευή θέτει πραγματικά προβλήματα και παρέχει άμεση ανατροφοδότηση.
- Επιτρέπει την πρόσκτηση γνώσεων και δεξιοτήτων που συνδέονται με πολλά γνωστικά αντικείμενα (και συνεπώς την προώθηση της διεπιστημονικής και διαθεματικής προσέγγισης).
- Δίνει την δυνατότητα για πιθανή διαισθητική συνειδητοποίηση σύνθετων φαινομένων, όπως η σχέση ανάμεσα στην ταχύτητα, το χρόνο και την μετακίνηση.
- Μπορεί ακόμα να αξιοποιηθεί σαν «σκαλωσιά» για την κατάκτηση συνθετότερων εννοιών όπως των δυναμικών συστημάτων της Βιολογίας κ.α.
- Στα πλαίσια διαθεματικών εκπαιδευτικών δραστηριοτήτων, μέσω αυτής, είναι δυνατόν να αναπτυχθούν κίνητρα για μάθηση και σε άλλα μαθήματα (Μαθηματικά, Φυσική κλπ.).
- Στηρίζεται στην συνεργασία και στην αλληλεπίδραση ατόμων και ομάδων και στην προώθηση της σκέψης μέσω γνωστικών και κοινωνικογνωστικών συγκρούσεων.
- Καλλιεργείται η επικοινωνία και η λεκτική έκφραση ιδεών μέσα από τη λειτουργία της ομάδας όταν οι μαθητές αναγκάζονται να εξηγήσουν ιδέες και σκέψεις τους.
- Υποστηρίζει την βιωματική μάθηση.
- Ως παιδαγωγική προσέγγιση εγγράφεται στο πλαίσιο του κλασικού εποικοδομητισμού και ειδικότερα του κατασκευαστικού

εποικοδομητισμού. Η εποικοδομητική αντίληψη για το φαινόμενο της μάθησης υποστηρίζει ότι το μαθησιακό περιβάλλον θα πρέπει να παρέχει αυθεντικές δραστηριότητες ενταγμένες σε διαδικασία επίλυσης ανοιχτών προβλημάτων από τον πραγματικό κόσμο, να ενθαρρύνει την έκφραση και την προσωπική εμπλοκή στη μαθησιακή διαδικασία και να υποστηρίζει την κοινωνική αλληλεπίδραση. Επιπλέον ο «κατασκευαστικός» εποικοδομητισμός υποστηρίζει ότι οι μαθητευόμενοι οικοδομούν πιο αποτελεσματικά τη γνώση όταν εμπλέκονται ενεργά στη σχεδίαση και κατασκευή πραγματικών αντικειμένων που έχουν νόημα για τους ίδιους είτε αυτά είναι κάστρα από άμμο, είτε κατασκευές Lego και προγράμματα υπολογιστών.

- Η μάθηση διαδικασιών ανάλυσης, σχεδιασμού δράσεων και στη συνέχεια η υλοποίησή τους συνιστά μια νοητική διέξοδο υψηλού επιπέδου- που εντάσσεται στην μεγάλη κατηγορία έργων που οι ψυχολόγοι ονομάζουν επίλυση προβλημάτων.
- Διευκολύνει την εκμάθηση του προγραμματισμού.
- Ο προγραμματισμός ρομποτικών κατασκευών δημιουργεί ένα εντελώς νέο περιβάλλον εργασίας για τους μαθητές με τα εξής χαρακτηριστικά:
 - α) Είναι έντονα παρακινητικό και συνεπώς παράγοντας υψίστης σημασίας για την διδακτική.
 - β) Ο προγραμματισμός της συμπεριφοράς των ρομπότ προκύπτει από μεταφορά υπαρχόντων και ήδη γνωστών συμπεριφορών από τους ζώντες οργανισμούς.
 - γ) Ευνοεί την στρατηγική δοκιμής-πλάνης, που είναι στρατηγική οικεία στους μαθητές του δημοτικού.
 - δ) Αναδεικνύει παραδεκτές προσεγγίσεις και λύσεις και όχι μια και μοναδική σωστή λύση αφού μια συμπεριφορά μπορεί να αποδοθεί με πολλούς τρόπους.
 - ε) Υποστηρίζει μεταγνωστικές διεργασίες μάθησης, δεδομένου ότι η προγραμματιστική δραστηριότητα οδηγεί στη συγκρότηση, την ανάλυση και την εξωτερίκευση νοητικών διεργασιών. Αυτή η προσπάθεια έχει μεταγνωστικό χαρακτήρα αφού μας αναγκάζει να

σκεφτόμαστε πάνω στον τρόπο που σκεφτόμαστε και ενεργούμε.

Η εκπαιδευτική ρομποτική στη διδασκαλία του προγραμματισμού

Αδυναμίες και δυσκολίες της παραδοσιακής διδασκαλίας προγραμματισμού

Μαθητές και φοιτητές αντιμετωπίζουν δυσκολίες στον προγραμματισμό. Βασική αιτία θεωρείται το γεγονός ότι οι μαθητές διδάσκονται κυρίως κάποια συγκεκριμένη γλώσσα προγραμματισμού και όχι κάποια μεθοδολογία για την επίλυση προβλημάτων. Τα κυριότερα προβλήματα που έχουν παρατηρηθεί κατά τη διδασκαλία της πληροφορικής και του προγραμματισμού:

- οι μαθητές δεν αντιλαμβάνονται τη διαδικασία επικοινωνίας μεταξύ ανθρώπου - μηχανής
- παρουσιάζουν δυσκολίες στην αντίληψη και την έκφραση των αλγορίθμων
- συνήθως παρουσιάζουν προβλήματα στην κατανόηση των επαναληπτικών δομών και δομών επιλογής, δυσκολία στο χειρισμό μεταβλητών κ.α
- κατανοούν τον αλγόριθμο σε φυσική γλώσσα, αλλά δυσκολεύονται στη μεταφορά του στην αυστηρή και λιτή σύνταξη και δομή του κώδικα
- τα περισσότερα προγραμματιστικά περιβάλλοντα είναι προσανατολισμένα προς έμπειρους προγραμματιστές
- οι γλώσσες γενικού σκοπού διαθέτουν ένα μεγάλο ρεπερτόριο εντολών και είναι πολύπλοκες
- η προσοχή των μαθητών επικεντρώνεται στην εκμάθηση της ίδιας της γλώσσας (αποστήθιση εντολών, σύνταξη κώδικα) και όχι στην ανάπτυξη ικανοτήτων επίλυσης προβλημάτων
- τα περισσότερα διδακτικά προβλήματα είναι μαθηματικού τύπου χωρίς αντιστοίχιση σε καθημερινά προβλήματα που να κεντρίζουν το ενδιαφέρον των διδασκομένων

Μία νέα διδακτική προσέγγιση για τη διδασκαλία του προγραμματισμού

Η εκπαιδευτική ρομποτική σε συνδυασμό με τον οπτικό προγραμματισμό μπορούν να

αξιοποιηθούν για τη διδασκαλία του προγραμματισμού και κατ' επέκταση για την αντιμετώπιση των δυσκολιών στον προγραμματισμό.

- Με τα φυσικά μηχανικά μοντέλα όπως της Lego πετυχαίνεται υψηλός βαθμός αλληλεπίδρασης μεταξύ υπολογιστή και πραγματικού αντικειμένου.
- Τα προγραμματιζόμενα 'τουβλάκια' Lego είναι φυσικά μοντέλα με παρατηρήσιμες φυσικές συμπεριφορές.
- Ενισχύουν τους βασικούς στόχους της διδασκαλίας του προγραμματισμού όπως την τεκμηρίωση και ανακάλυψη, τη μάθηση νέων συστημάτων συμβόλων, την επικοινωνία μεταξύ μηχανών και τη μάθηση αλγορίθμων (Lawhead et al., 2003).
- Υπάρχει η δυνατότητα για πειραματισμό και δοκιμή και ενεργός συμμετοχή από τους μαθητές, αναπτύσσεται η κριτική σκέψη, καλλιεργείται η δημιουργική σκέψη, η διορατικότητα και η πρωτοτυπία, υπάρχει άμεση εμπειρία και ο μαθητής απαλλάσσεται από την εκμάθηση και απομνημόνευση συντακτικών κανόνων μιας γλώσσας προγραμματισμού (Καγκάνη κ.α., 2005).
- Ο μαθητής είναι σε θέση να γράψει κώδικα σε σύντομο χρονικό διάστημα.
- Στο περιβάλλον συγγραφής του κώδικα γίνεται ταυτόχρονα και συντακτικός έλεγχος (Τσοβόλας & Κόμης, 2005).
- Οι προγραμματιστικές έννοιες αποκτούν νόημα για τους μαθητές χάρη στην άμεση ανάδραση που υπάρχει ανάμεσα στον αλγόριθμο και στην υλοποίησή του.
- Το οπτικό περιβάλλον της οπτικής γλώσσας προγραμματισμού είναι φιλικό σε όλους τους μαθητές, αφού εξοικειώνονται εύκολα με τα εικονίδια και τη σημασία της κάθε οπτικής εντολής. Ο αναπαραστατικός χαρακτήρας των εντολών βοηθάει τα παιδιά να αναγνωρίζουν τη λειτουργία των εντολών από το εικονίδιό τους.
- Δίνει τη δυνατότητα υλοποίησης ρεαλιστικών σεναρίων (πχ. παρκάρισμα του οχήματος).
- Διευκολύνει τη μεταφορά γνώσεων από το ρομποτικό περιβάλλον σε άλλα προγραμματιστικά περιβάλλοντα.
- Η πτυχή του παιχνιδιού που εμπεριέχουν τα προγραμματιζόμενα ρομπότ Mindstorms προτρέπει τους μαθητές να είναι περισσότερο δημιουργικοί

αντιμετωπίζοντας τον προγραμματισμό του ρομπότ ως μια ψυχαγωγική και ευχάριστη ενασχόληση, ενισχύοντας σημαντικά τη διάθεσή τους για ενασχόληση με τον προγραμματισμό.

- Χαρακτηρίζεται από υψηλό βαθμό αλληλεπίδρασης μεταξύ του υπολογιστή και του πραγματικού αντικειμένου, με αποτέλεσμα, ο διδασκόμενος να μπορεί να συσχετίσει τις αντιδράσεις του ρομπότ με τις εντολές του προγράμματος και να παρατηρήσει τις συνέπειες που έχουν στη συμπεριφορά του ρομπότ οι αλλαγές που πραγματοποιεί στο πρόγραμμα (Eden et al. 1996).
- Τα ρομπότ προσφέρουν απτή ανατροφοδότηση (feedback) στους μαθητές για την αποτελεσματικότητα των προγραμμάτων τους (Barnes 2002). Η δυνατότητα αυτή βοηθά τους μαθητές να οδηγούνται σταδιακά σε καλύτερες, αποτελεσματικότερες, πληρέστερες και ακριβέστερες λύσεις.
- Ακόμη, επιτρέπει στον εκπαιδευτικό να διδάξει την ίδια θεωρία και τις ίδιες έννοιες όχι μόνο πιο αποτελεσματικά αλλά και σε λιγότερο χρόνο (Fagin 2000). Το γεγονός αυτό σημαίνει ότι ο καθηγητής έχει στη διάθεσή του περισσότερο χρόνο για να παρακολουθήσει την πορεία κάθε μαθητή ξεχωριστά και να εντοπίσει τις αδυναμίες, παρανοήσεις, δυσκολίες που αυτός αντιμετωπίζει. Επίσης, ο μαθητής εργάζεται με το δικό του ρυθμό μάθησης, έτσι ώστε να μπορεί να σταθεί όσο χρόνο χρειάζεται σε κάποια σημεία.
- Υποστηρίζει το σχεδιασμό δραστηριοτήτων διδασκαλίας προγραμματισμού στηριγμένων στη διερευνητική μάθηση (Ramadhan, 2000; Kolikant & Pollack, 2004) και στη συνεργατική μάθηση (Γρηγοριάδη κ.α., 2004). Οι Williams & Kessler (2000) αναφέρουν ότι η συνεργασία μεταξύ των μαθητών αποτελεί καθοριστικό παράγοντα στη μάθηση του προγραμματισμού.
- Υποστηρίζει τη διδασκαλία προγραμματισμού σε διάφορα γνωστικά επίπεδα και βαθμίδες εκπαίδευσης. Ειδικότερα, η εκπαιδευτική πλατφόρμα της LEGO MINDSTORMS υποστηρίζει μία μεγάλη ποικιλία γλωσσών προγραμματισμού που μπορούν να χρησιμοποιηθούν για τον προγραμματισμό ρομπότ Mindstorms - γεγονός που την καθιστά κατάλληλη τόσο για μικρές ηλικίες (με τον οπτικό προγραμματισμό) όσο και για τη διδασκαλία του προγραμματισμού (με άλλες γλώσσες πχ. RobotC) σε μεγαλύτερες ηλικίες ή

σε μαθητές που έχουν διδαχθεί τις βασικές προγραμματιστικές δομές (πχ. στη δευτεροβάθμια εκπαίδευση).

Μειονεκτήματα

Η χρήση φυσικών μοντέλων κατά τη διδασκαλία των αρχών του προγραμματισμού μπορεί να δημιουργήσει διδακτικά προβλήματα, τα σημαντικότερα από τα οποία οφείλονται σε φυσικούς περιορισμούς και τεχνικούς περιορισμούς του υλικού - περιορισμοί που μπορεί να δημιουργήσει η τριβή στην κίνηση ενός αντικειμένου κ.α, χρονικούς περιορισμούς (χρόνος εκμάθησης του συστήματος) και περιορισμούς κόστους επένδυσης (μεγάλη οικονομική επιβάρυνση για την απόκτηση και συντήρηση - επικαιροποίηση απαιτούμενου αριθμού συστημάτων).

3.1.2 Λεπτομερής σχεδιασμός μιας διδασκαλίας με τους μικροελεγκτές

Οι μικροελεγκτές αποτελούν πολύ μικρές ηλεκτρονικές συσκευές ο οποίες έχουν κυριαρχήσει στον κόσμο των ηλεκτρονικών καθώς πλέον βρίσκονται σε κάθε συσκευή. Το air condition, το DVD, η τηλεόραση ή ακόμη και το κινητό λειτουργούν με μικροελεγκτές. Η δυνατότητά τους είναι απεριόριστη και μάλιστα πολλές φορές αγγίζει και τα όρια της φαντασίας. Επιπλέον, είναι ιδιαίτερα εύκολο να χρησιμοποιηθούν καθώς δε χρειάζονται κάποια ιδιαίτερη γνώση στα ηλεκτρονικά και αρκεί μονάχα μία σχετική κατανόηση σε θέματα ψηφιακών κυκλωμάτων.

Κάθε μικροελεγκτής αποτελείται από ενσωματωμένα κυκλώματα και με τη χρήση ενός προγράμματος μπορεί να εκτελέσει μία λειτουργία όπως για παράδειγμα να ενεργοποιεί ένα κουμπί, να αλλάζει το κανάλι στην τηλεόραση ή ακόμη και να ελέγχει κάποιες πιο σύνθετες κινήσεις ενός ρομπότ.

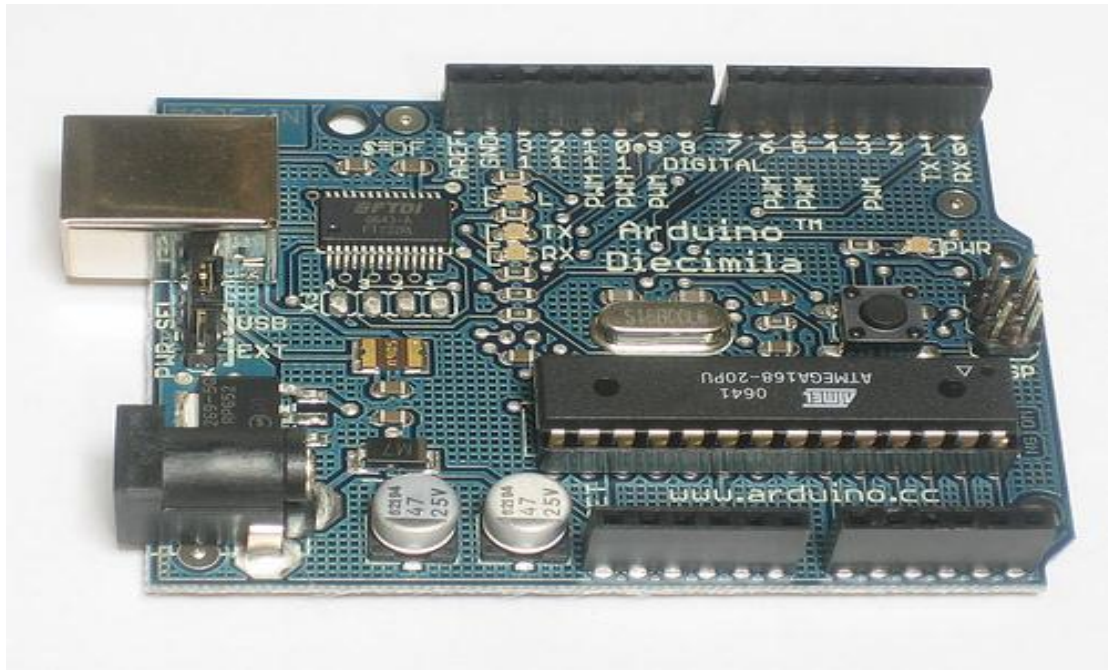
Κύριο μέρος του μικροελεγκτή είναι η CPU η οποία σε συνδυασμό με ένα αριθμό εξωτερικών συσκευών επικοινωνεί με τον πραγματικό κόσμο. Ένα τυπικό σύστημα το οποίο χρησιμοποιεί μικροελεγκτή διαβάσει εισόδους από το πληκτρολόγιο και παρουσιάζει τις εξόδους στην οθόνη αποθηκεύοντας τες ταυτόχρονα στη μνήμη που

διαθέτει. Ο μικροελεγκτής από την άλλη, συνδυάζει όλες αυτές τις συσκευές σε μία καθώς περιέχει, CPU, RAM, ROM, I/O κλπ στο ίδιο ενσωματωμένο κύκλωμα. Το γεγονός αυτό διευκολύνει ιδιαίτερα την διαδικασία ανάπτυξης ενώ ταυτόχρονα μειώνει τις απαιτήσεις σε εξωτερικές συσκευές.

Οι εφαρμογές στις οποίες μπορεί να χρησιμοποιηθεί ένας μικροελεγκτής ποικίλουν και συνήθως είναι απλές χωρίς να απαιτούν ιδιαίτερη επεξεργασία όπως οι εφαρμογές που εκτελούνται σε έναν υπολογιστή.

Ο πρώτος μικροελεγκτής δημιουργήθηκε από την εταιρία INTEL το 1969 και μετά από αίτημα μίας ιαπωνικής εταιρίας για βοήθεια σε κάποιο πρόβλημα που αντιμετώπιζε σε ένα Project που αφορούσε στα ενσωματωμένα κυκλώματα. Από τότε η εξέλιξή τους είναι ραγδαία ενώ πολλοί και διάφοροι τύποι μικροελεγκτών είναι πλέον διαθέσιμοι στο εμπόριο.

1. Arduino



Εικόνα 3.1: πλακέτα Arduino

Ο μικροελεγκτής Arduino είναι ένα ιδιαίτερα χρήσιμο εργαλείο το οποίο μπορεί να δώσει στους υπολογιστές τη δυνατότητα τόσο να ανιχνεύσουν όσο και να ελέγξουν ένα πολύ μεγαλύτερο κομμάτι του φυσικού κόσμου από ότι ένας επιτραπέζιο υπολογιστής. Πρόκειται για μία πλατφόρμα ανοιχτού λογισμικού που βασίζεται στην ύπαρξη ενός μικροελεγκτή και ενός περιβάλλοντος ανάπτυξης λογισμικού για την πλακέτα αυτή.

Ο Arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη διαδραστικών αντικειμένων καθώς μπορεί να δεχθεί εισόδους από μία πληθώρα διακοπών ή αισθητήρων και να ελέγξει μία πληθώρα συσκευών όπως ο φωτισμός, οι μηχανές αλλά και άλλες φυσικές έξοδοι. Οι εφαρμογές του Arduino μπορούν είτε να λειτουργούν ανεξάρτητα είτε σε επικοινωνία με τον υπολογιστή.

Η γλώσσα προγραμματισμού που χρησιμοποιεί είναι μία υλοποίηση του Wiring, μίας παρόμοιας πλατφόρμας προγραμματισμού η οποία βασίζεται στο λογισμικό Processing.

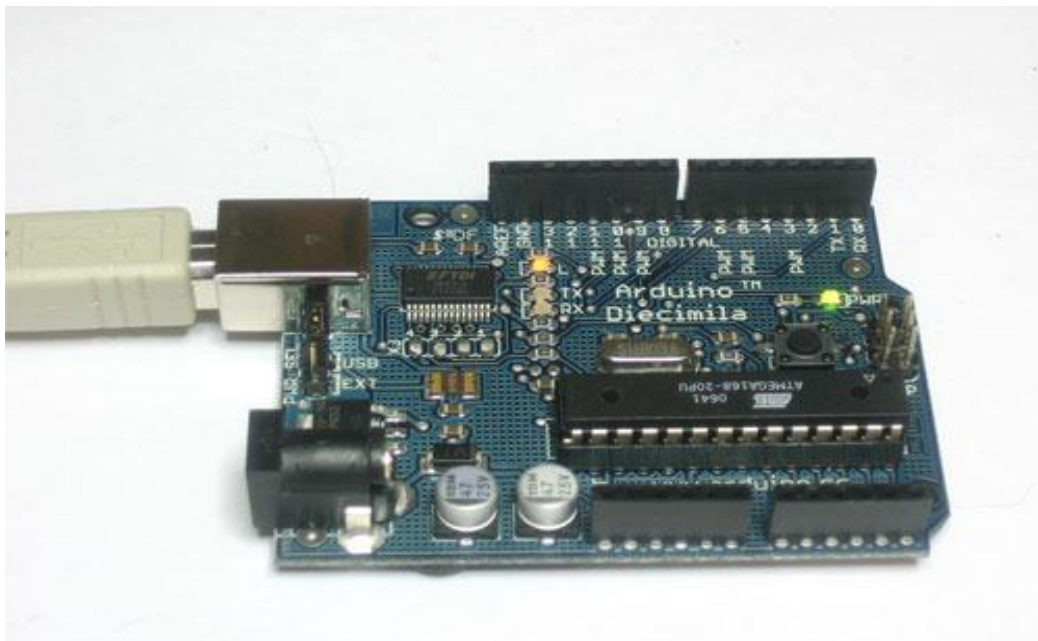
Παρακάτω θα παρουσιαστεί και θα αναλυθεί μία εισαγωγή στη χρήση του Arduino η οποία θα αφορά περισσότερο στη σωστή χρήση και σύνδεση των συσκευών ώστε

αυτές να μπορούν να χρησιμοποιηθούν και στο μέλλον. Τα υλικά τα οποία θα χρειαστούν αρχικά είναι :

- ⇒ Arduino Uno / Diecimila / NG
- ⇒ Καλώδιο USB
- ⇒ Τροφοδοσία 9V
- ⇒ Ένα LED

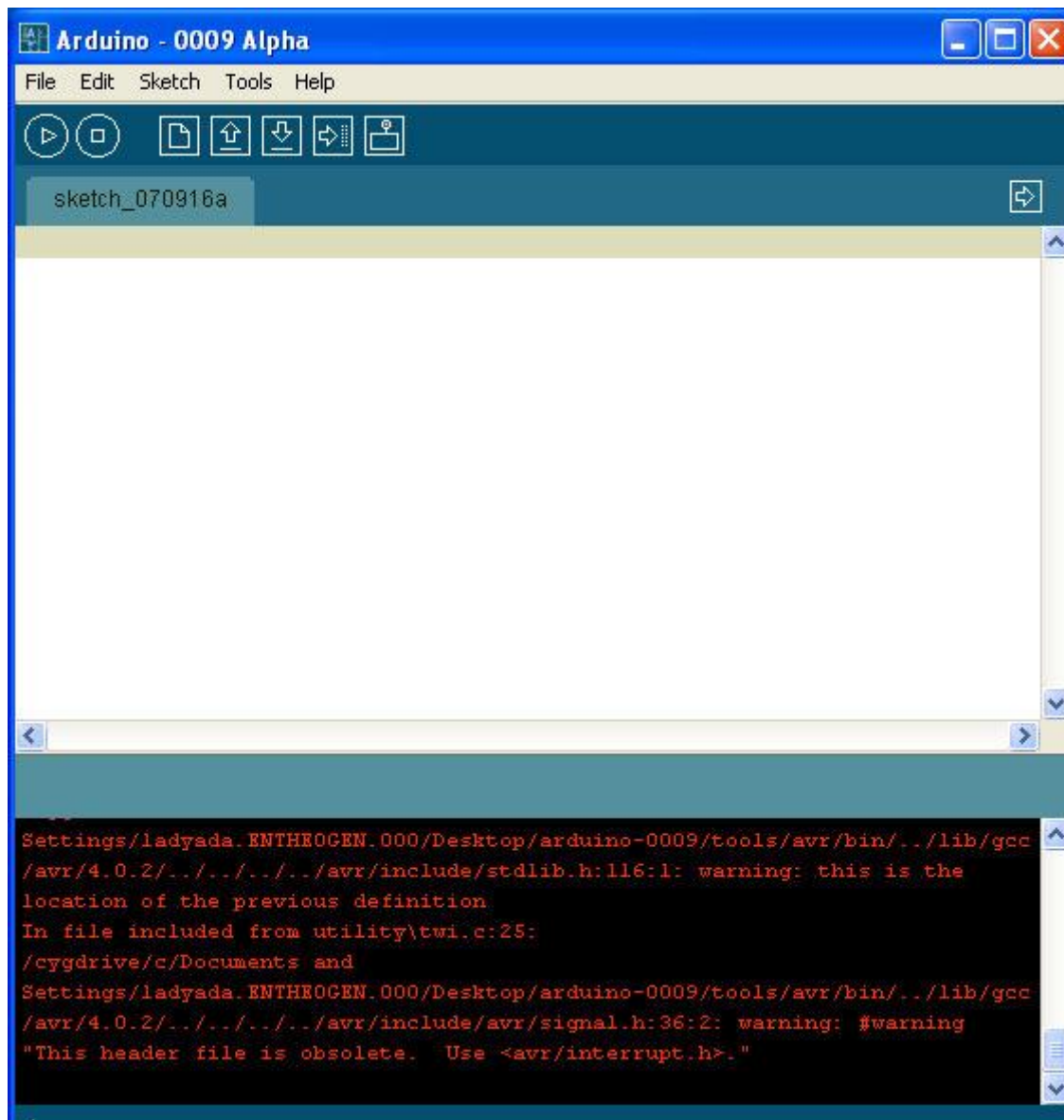
Αφού επιλεγεί ο Arduino που θα χρησιμοποιηθεί θα πρέπει να γίνει εγκατάσταση των κατάλληλων drivers που θα χρησιμοποιηθούν ανάλογα με το λογισμικό του υπολογιστή. Ακόλουθα, συνδέεται ο υπολογιστής με το Arduino διαμέσου του καλωδίου USB ούτως ώστε να τροφοδοτηθεί ο Arduino με ρεύμα. Βέβαια, η τροφοδοσία του Arduino μπορεί να γίνει επίσης και με μία μπαταρία 9V ή με μία πρίζα.

Υποθέτουμε λοιπόν ότι χρησιμοποιούμε ένα USB καλώδιο. Αρχικά τοποθετούμε την μία υποδοχή στον Arduino και τη δεύτερη στην κατάλληλη θύρα του υπολογιστή. Μόλις τελειώσει η διαδικασία αυτή θα πρέπει να φανεί ένα πράσινο λαμπάκι να ανάβει στη δεξιά μεριά της πλακέτας και κάποια πορτοκαλί να αναβοσβήνουν στην αριστερή όπως φαίνεται στην εικόνα.



Εικόνα 3.2: λειτουργία της πλακέτας

Περνώντας στον προγραμματισμό του μικροελεγκτή, θα πρέπει να υπάρχει εγκατεστημένο στον υπολογιστή το κατάλληλο λογισμικό. Στην επίσημη σελίδα του Arduino μπορεί κανείς να βρει όλες τις διαθέσιμες εκδόσεις του διαθέσιμου λογισμικού και να κατεβάσει αυτή που είναι κατάλληλη για το λειτουργικό του σύστημα. Αφού εγκατασταθεί το πρόγραμμα στην επιφάνεια εργασίας γίνεται διπλό κλικ σε αυτό ώστε να ανοίξει. Το περιβάλλον του προγράμματος είναι αντίστοιχο με αυτό της παρακάτω εικόνας.



Εικόνα 3.3: περιβάλλον ανάπτυξης προγραμματισμού

Στη συνέχεια γίνεται επιλογή του κατάλληλου τσιπ που θα χρησιμοποιηθεί για τον Arduino. Τα πλέον συνήθη είναι το ATmega 8 και το ATmega 128. Η πόρτα με την οποία επιλέγεται συνήθως να συνδεθεί ο Arduino είναι η COM3 ή η COM4.

Το πρώτο παράδειγμα που θα χρησιμοποιηθεί είναι αυτό του Blinkie ο κώδικας του οποίου παρατίθεται παρακάτω. Η λειτουργία που εκτελεί ουσιαστικά το πρόγραμμα είναι το άναμμα και το σβήσιμο κάποιων από τα LED που υπάρχουν στην πλακέτα του Arduino. Τα προγράμματα κυρίως είναι γραμμένα σε γλώσσα C το λογισμικό όμως έχει την ιδιότητα να μεταφράζει τη C στη γλώσσα του Arduino.

```
/*
 * Blink
 *
 * The basic Arduino example. Turns on an LED
 on for one second,
 * then off for one second, and so on... We
 use pin 13 because,
 * depending on your Arduino board, it has
 either a built-in LED
 * or a built-in resistor so that you need
 only an LED.
 *
 * http://www.arduino.cc/en/Tutorial/Blink
 */

int ledPin = 13;                      // LED
connected to digital pin 13

void setup()                          // run once,
when the sketch starts
{
    pinMode(ledPin, OUTPUT);          // sets the
digital pin as output
}

void loop()                           // run over
and over again
{
```

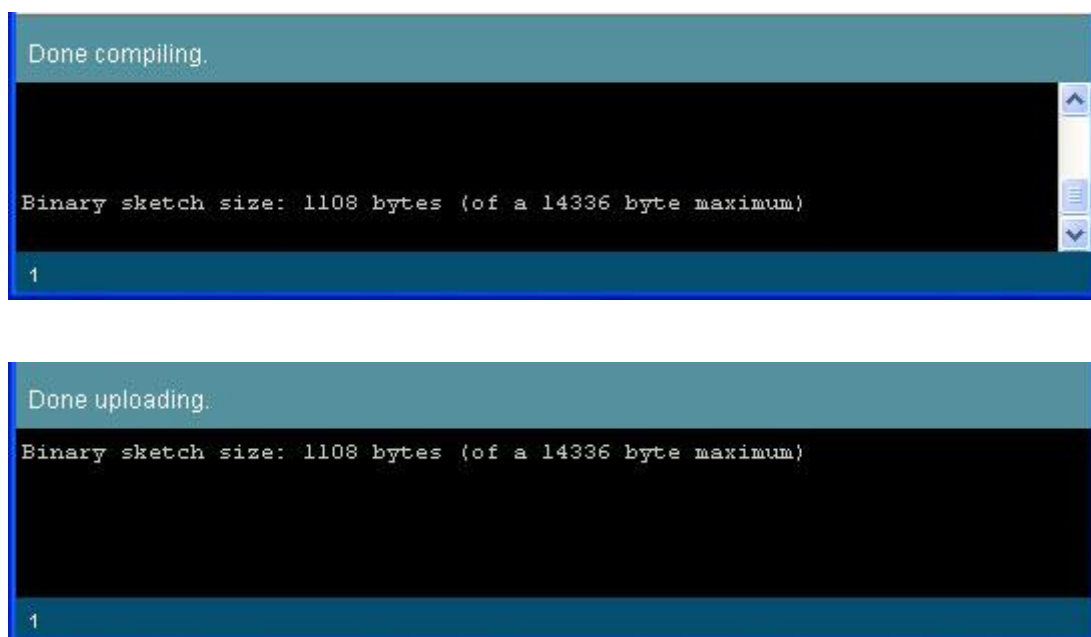
```

    digitalWrite(ledPin, HIGH);    // sets the
LED on
    delay(1000);                  // waits for a
second
    digitalWrite(ledPin, LOW);    // sets the
LED off
    delay(1000);                  // waits for a
second
}

```

Εικόνα 3.4: παράδειγμα Blink

Αφού αποθηκευθεί ως αρχείο ο παραπάνω κώδικας ξεκινά η διαδικασία της μεταγλώττισής του με τη χρήση του λογισμικού. Όταν αυτή ολοκληρωθεί όπως φαίνεται στην εικόνα τα LED θα αρχίσουν να αναβοσβήνουν στους χρόνους που έχουν οριστεί.

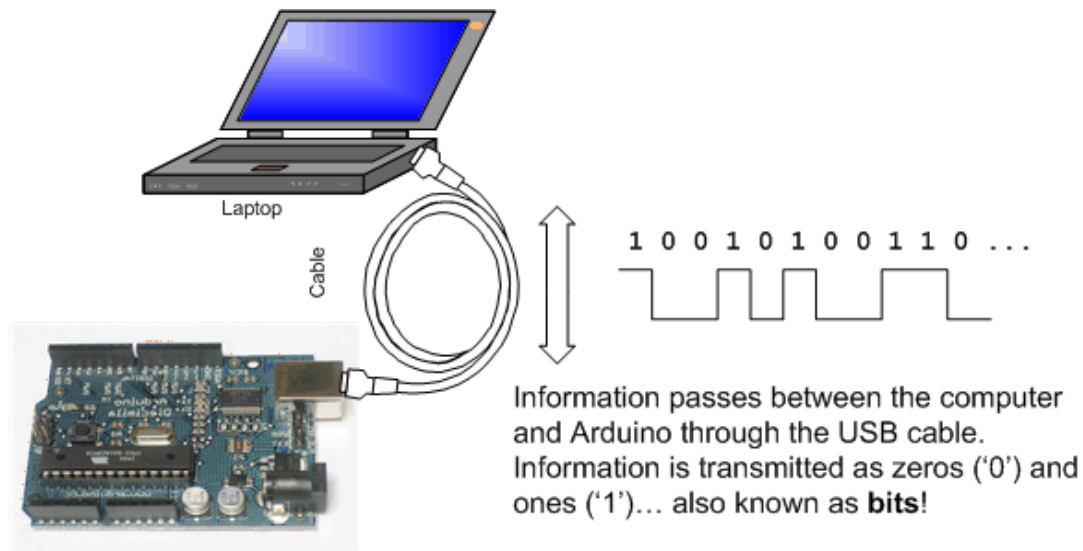


Εικόνα 3.5: διαδικασία ολοκλήρωσης μεταγλώττισης

Εάν θελήσει κανείς να φορτώσει ένα νέο πρόγραμμα στον μικροελεγκτή δεν έχει παρά να γράψει το πρόγραμμα του, να το αποθηκεύσει και έπειτα από του μενού Edit να επιλέξει το υπό – μενού Upload.

Ακόλουθα θα μελετηθεί το θέμα των βιβλιοθηκών που τόσο περιέχονται όσο και μπορούν να χρησιμοποιηθούν από τον Arduino. Οι βιβλιοθήκες λογισμικού

αποτελούν μία συλλογή διαδικασιών οι οποίες σχετίζονται άμεσα μεταξύ τους. Η βιβλιοθήκη που χρησιμοποιείται στην περίπτωση αυτή είναι η σειριακή βιβλιοθήκη η οποία επιτρέπει στον Arduino να στείλει τα δεδομένα πίσω στον υπολογιστή. Η παρακάτω εικόνα παρουσιάζει αυτήν ακριβώς τη διαδικασία, δηλαδή τη σειριακή επικοινωνία του Arduino με τον υπολογιστή.



Εικόνα 3.6: συριακή επικοινωνία του Arduino με τον υπολογιστή

Στη συνέχεια θα δημιουργηθεί και θα παρουσιαστεί ένα νέο αρχείο που θα ονομάζεται Hello World. Αρχικά δημιουργείται το αρχείο και αποθηκεύεται σε κάποιο φάκελο. Στην συνέχεια προστίθεται σε αυτό ο παρακάτω κώδικας.

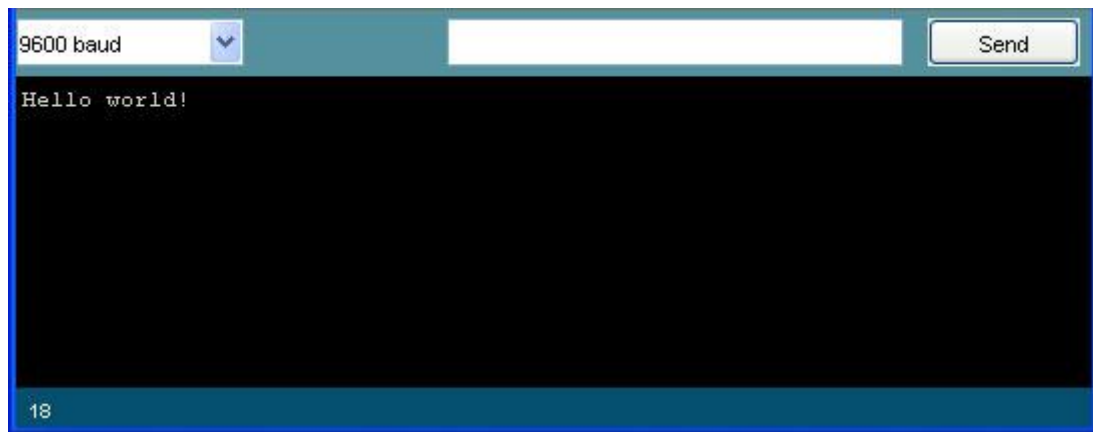
```
void setup() // run once, when the
sketch starts
{
  Serial.begin(9600); // set up Serial library
  at 9600 bps

  Serial.println("Hello world!"); // prints hello with
  ending line break
}
```

```
void loop() // run over and over  
again  
{  
    // do nothing!  
}
```

Εικόνα 3.7: κώδικα του hello word

Αφού αποθηκευθεί και εκτελεστεί ο παραπάνω κώδικας στην οθόνη θα παρουσιαστεί το μήνυμα Hello World. Για να γίνει αυτό θα πρέπει στην οθόνη του υπολογιστή και στο λογισμικό του Arduino να ενεργοποιηθεί το κουμπί Serial Monitor. Το μήνυμα θα εμφανιστεί ακριβώς όπως παρουσιάζεται στην παρακάτω εικόνα.



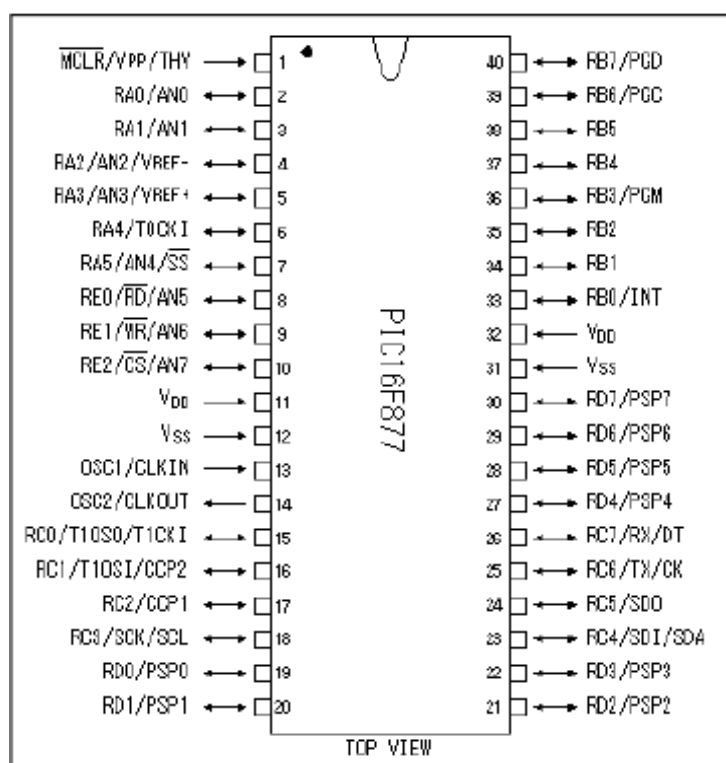
Εικόνα 3.8: μήνυμα του εκτελέσιμου προγράμματος

2. PIC



Εικόνα 3.90: μικροελεγκτής PIC

Ο μικροελεγκτής PIC παράγεται από την εταιρία Microchip η οποία αρχικά τους χρησιμοποιούσε ως υποστηρικτικές συσκευές για τους PDP υπολογιστές ώστε να ελέγχουν τις περιφερειακές συσκευές τους. Το όνομα PIC προέρχεται από τις λέξεις Peripheral Interface Controller.



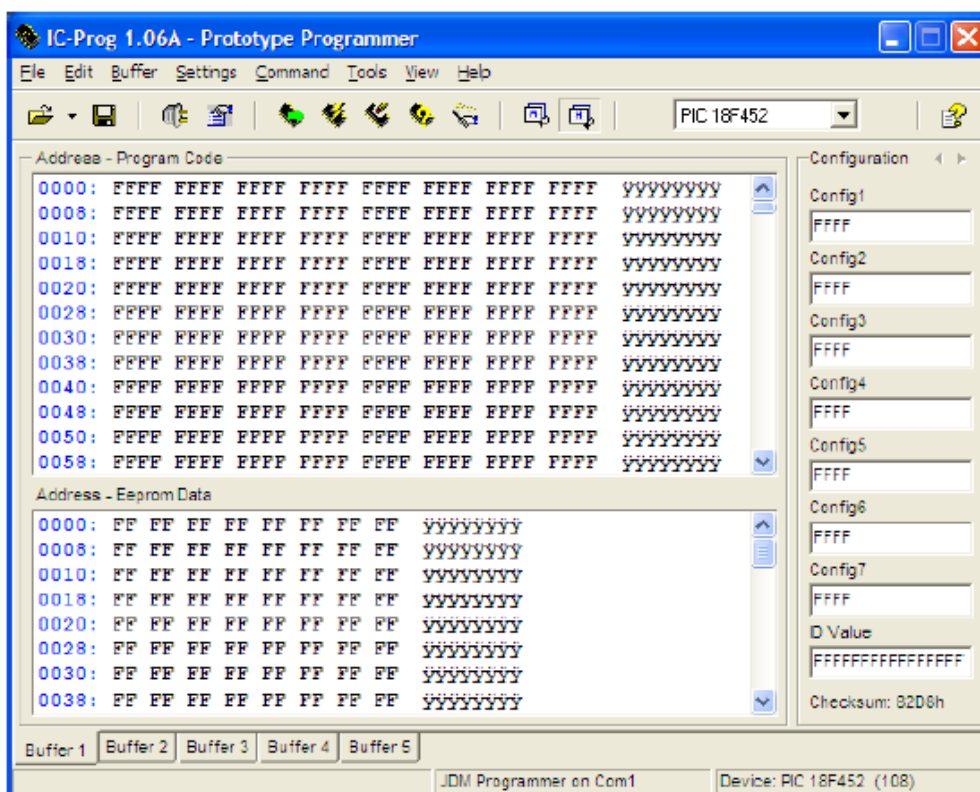
Εικόνα 3.10: δομή του μικροελεγκτή PIC

Η παραπάνω εικόνα παρουσιάζει τη γενικότερη δομή του μικροελεγκτή στην οποία όπως μπορεί κανείς να διακρίνει κάθε pin έχει συνδυαστεί με έναν αριθμό συναρτήσεων. Η επιλογή της συνάρτησης που θα ανατεθεί σε κάθε pin έχει να κάνει με τους εσωτερικούς καταχωρητές και με το πώς αυτοί χρησιμοποιούν τα διάφορα bit.

Η τροφοδοσία του μικροελεγκτή PIC μπορεί να ποικίλει από τα 3,5V έως τα 5V καθώς απαιτεί πολύ χαμηλή ισχύ ρεύματος. Έτσι λοιπόν ένα σήμα το οποίο θα έχει τάση από 0 έως 2V μπορεί να θεωρηθεί ως λογικό 0 και ένα σήματα από 3.5V έως 4.5V ως λογικό 1.

Σε κάθε μικροελεγκτή PIC μπορεί κανείς να συναντήσει ένα MCLR Pin το οποίο χρησιμοποιείται κυρίως για να γίνεται το reset του μικροελεγκτή.

Για να μπορέσει κανείς να προγραμματίσει τον μικροελεγκτή θα πρέπει αρχικά να εγκαταστήσει το λογισμικό του στον υπολογιστή. Το πιο δημοφιλές λογισμικό είναι το ICPROG το οποίο διατίθεται ελεύθερο στο διαδίκτυο. Μετά την εγκατάσταση του προγράμματος παρουσιάζεται στον υπολογιστή μία εικόνα όπως η παρακάτω.



Εικόνα 3.11: εγκατάσταση του προγράμματος στον υπολογιστή

Αφού γίνει η πλήρης εγκατάσταση του λογισμικού συνδέεται ο PIC με τον υπολογιστή διαμέσου ενός καλωδίου το οποίο τοποθετείται και στις δύο μεριές του μικροελεγκτή δηλαδή την κεφαλή και τον ICSP connector. Μόλις γίνει η σύνδεση θα ανάψει το LED της τροφοδοσίας του μικροελεγκτή καθώς αυτός πλέον θα λαμβάνει τάση 5V από τον υπολογιστή.

Η συγγραφή ενός προγράμματος για τον μικροελεγκτή είναι μία εύκολη διαδικασία καθώς χρειάζεται μόνο να υπάρχει ένα αρχείο με την προέκταση .hex. Η γλώσσα προγραμματισμού που χρησιμοποιείται είναι η BASIC.

Το πρώτο πρόγραμμα που χρησιμοποιείται σε κάθε προγραμματισμό είναι το Hello World και το ίδιο θα γίνει και σε αυτήν την περίπτωση. Θα γραφεί ένα πρόγραμμα Hello World επειδή όμως δεν υπάρχει οθόνη απεικόνισης η επιτυχία του θα φανεί στα LEDs που θα αναβοσβήσουν. Για να γίνει αυτό θα πρέπει να συνδεθούν όλα τα LED στην PORTC έτσι ώστε να είναι ενεργά. Το μόνο που χρειάζεται να γίνει είναι να γραφεί ένα λογικό 1 στα bit του PORTC. Η παρακάτω εικόνα δείχνει πως θα γραφεί το πρόγραμμα αυτό.

```
Device=18F452      'Select Microcontroller
XTAL 20            'Select Clock frequency 20MHz
ALL_DIGITAL true   'Make All lines digital as we are not going to use Analog
Output PORTC       'Make All bits of PORTC as output Same as TRISC=0
PORTC= %11111111   'Assign all bits of PORTC a logical 1
End                'Put processor into an endless loop
```

Εικόνα 3.12: πρόγραμμα hello word

Το πρόγραμμα αυτό θα πρέπει να αποθηκευθεί ως Hello.bas, να μεταγλωττιστεί και να φορτωθεί στον μικροελεγκτή. Αφού ολοκληρωθεί η διαδικασία θα πρέπει ο μικροελεγκτής να αποσυνδεθεί από το καλώδιο και να απενεργοποιηθεί. Κατά την επανενεργοποίηση του μικροελεγκτή θα πρέπει όλα τα LEDs να ανάβουν γεγονός το οποίο θα σημαίνει και την επιτυχία του προγράμματος.

Επόμενα θα σχεδιαστεί ένα πρόγραμμα το οποίο θα κάνει τα LED να αναβοσβήνουν, το γνωστό blink. Στο αρχείο γράφεται και εκτελείται ο παρακάτω κώδικας.

```

Device=18F452      'Select Microcontroller
XTAL 20            'Select Clock frequency 20MHz
ALL_DIGITAL true   'Make All lines digital as we are not going to use Analog
Output PORTC       'Make All bits of PORTC as output Same as TRISC=0
Loop:
    PORTC= 255      'Assign all bits of portc a logical 1
    DelayMS 1000    'let the LEDs remain ON for 1 second
    PORTC= 0        'Turn LEDs OFF
    DelayMS 1000    'Keep Them OFF for 1 Second
GoTo Loop          'Go back and repeat the process

```

Εικόνα 3.13: πρόγραμμα που αναβοσβήνει τα LED

Όπως μπορεί κανείς να διακρίνει στον κώδικα τα LED θα αναβοσβήσουν με συχνότητα 1 sec όπως έχει οριστεί.

3.1.3 Δυσκολίες και πλεονεκτήματα

Η χρήση των μικροελεγκτών παρουσιάζει κάποια μοναδικά συγκριτικά πλεονεκτήματα όπως

- ✓ Χαμηλό κόστος : Οι περισσότεροι από αυτούς έχουν χαμηλό κόστος που φτάνει τα ελάχιστον ευρώ και έτσι μπορούν να αγοραστούν άμεσα από οποιονδήποτε ενδιαφερόμενο.
- ✓ Υψηλή δια-λειτουργικότητα : Οι μικροελεγκτές παρουσιάζουν ιδιαίτερα υψηλή δια-λειτουργικότητα καθώς υποστηρίζουν τα διάφορα λειτουργικά συστήματα και μπορούν να λειτουργήσουν επιτυχώς όποιο σύστημα και αν χρησιμοποιεί ο χρήστης.
- ✓ Εύκολο περιβάλλον προγραμματισμού : Τα λογισμικά προγραμματισμού που χρησιμοποιούν οι μικροελεγκτές υποστηρίζονται από ένα ιδιαίτερα εύχρηστο και φιλικό προς το χρήστη περιβάλλον είτε αυτός είναι άπειρος είτε ιδιαίτερα εξοικειωμένος με τη χρήση.

- ✓ Αντικατάσταση συσκευών : Ένας μικροελεγκτής μπορεί να αντικαταστήσει πολλές από τις περιφερειακές συσκευές και να αναλάβει το ρόλο τους στο σύστημα γεγονός το οποίο τόσο το απλοποιεί όσο και του χαμηλώνει το κόστος λειτουργίας.
- ✓ Ευελιξία και προσαρμοστικότητα : Οι μικροελεγκτές είναι ιδιαίτερα ευέλικτοι ως υλικό και προσαρμόζονται σε όλα τα περιβάλλοντα λειτουργίας υιοθετώντας πάρα πολύ γρήγορα τις όποιες αλλαγές σε υλικό ή σε κώδικα.

Πέραν των πλεονεκτημάτων τους παρουσιάζουν και κάποια μειονεκτήματα τα οποία όμως δεν είναι ικανά να υπερνικήσουν. Η χρήση των μικροελεγκτών συνήθως απαιτεί γνώσεις και κατανόηση προγραμματισμού την οποία δυστυχώς δεν διαθέτουν όλοι οι δυνητικοί τους χρήστες. Πέραν τούτου, πρόκειται για σχετικά εύθραυστο υλικό οπότε απαιτείται ιδιαίτερη προσοχή στον τρόπο που χρησιμοποιείται. Τέλος, πολλές από τις ρομποτικές εφαρμογές στις οποίες χρησιμοποιούνται οι μικροελεγκτές μπορεί τελικά να καταλήξουν να είναι ασύμφορες οικονομικά για το χρήστη καθώς θα απαιτούν την προσθήκη πολλών εξαρτημάτων.

Σε ότι αφορά το κομμάτι της διδασκαλίας η χρήση των μικροελεγκτών είναι ιδιαίτερα σημαντική για τους μαθητές καθώς τους εισάγει στον ιδιαίτερα έξυπνο και χρήσιμο κόσμο των ηλεκτρονικών. Οι μικροελεγκτές μπορούν να χρησιμοποιηθούν σε οποιαδήποτε έκφανση της καθημερινότητας και άρα η γνωριμία ενός μαθητή με αυτούς μπορεί να τον βοηθήσει ώστε να απλοποιήσει τη ζωή του. Πολλές από τις περιττές διαδικασίες στις οποίες θα έμπαινε ενδεχόμενα να εξαφανιστούν και να πάρουν τη θέση τους οι μικροελεγκτές που βάση προγράμματος θα μπορούν για παράδειγμα να ανοίγουν τη θέρμανση, να λειτουργούν ως σύστημα ασφαλείας, να ελέγχουν τη θερμοκρασία του χώρου κλπ. Επιπλέον, η γνωριμία με αυτού του τύπου τις συσκευές διευρύνει και τη δημιουργικότητα των μαθητών οι οποίοι θα μπορούν να υλοποιήσουν δικές τους ιδέες και να τις προτείνουν ακόμη και ως επιχειρηματικές. Τέλος, η εκτενής χρήση των μικροελεγκτών σήμερα αποτελεί εξέλιξη της τεχνολογίας και είναι καλό για κάθε μαθητή να συμβαδίζει με αυτήν και να είναι ενημερωμένος σε ότι αφορά το γίνεσθε.

Δυστυχώς, ένας τέτοιος εξοπλισμός είναι ιδιαίτερα δύσκολο να συντηρηθεί ώστε να μη χαλάσει και να διαρκέσει περισσότερο του αναμενόμενου. Η συνεχής επανάληψη αλλά και η λάθος από άγνοια πολλές φορές χρήση οδηγούν στην καταστροφή του χωρίς να είναι πολλές φορές δυνατή η αντικατάσταση.

Ακόμη, σημαντικά μεγάλη είναι η δυσκολία και στο να αγοραστεί και να διατεθεί στους μαθητές τέτοιου είδους εξοπλισμός καθώς στα περισσότερα εκπαιδευτικά ιδρύματα δεν υπάρχουν τα αντίστοιχα κονδύλια.

3.1.4 Προτάσεις

Πολλές είναι οι προτάσεις εκείνες που θα μπορούσαν να γίνουν ώστε ο εξοπλισμός αυτός να υπάρχει στα εκπαιδευτικά ιδρύματα και να χρησιμοποιείται ώστε να εισαγάγει τους μαθητές στον κόσμο των ηλεκτρονικών. Αρχικά θα μπορούσε να γίνει μία άμεση επικοινωνία με τις ίδιες τις εταιρίες και να λειτουργήσει η ύπαρξη του υλικού ως μία μορφή διαφήμισης της εταιρίας προς το δυνητικό καταναλωτικό κοινό της.

Πέραν τούτου θα μπορούσαν συχνά να διοργανώνονται διαγωνισμοί δημιουργικότητας και ρομποτικής με τη χρήση εξοπλισμού από διάφορες εταιρίες ώστε να υπάρχει χώρος και για διαφήμιση των εταιριών αλλά και για επιπλέον ενδιαφέρον από τους ίδιους τους μαθητές.

Σε ότι αφορά το ίδιο το μάθημα τώρα, θα πρέπει να υπάρχει κάποιος υπεύθυνος του εξοπλισμού ώστε να μπορεί να τον προφυλάσσει από κακή λειτουργία αλλά και για να είναι υπεύθυνος της μεταφοράς και της διαφύλαξής του.

ΑΝΑΦΟΡΕΣ:

[1] <http://chazstefos.blogspot.gr/2013/01/blog-post.html>

[2] <https://sites.google.com/site/robotokosmos/home/ekpaideutike-rompotike-paideia>

[3] <http://dide.ilei.sch.gr/keplinet/education/articles-rob.php>

