

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ



Διπλωματική Εργασία
**Μελέτη και Υλοποίηση πλατφόρμας πειραμάτων Λογικής
Σχεδίασης βασισμένη σε Μικροελεγκτή AVR™
και Γραφικής Εφαρμογής σε JAVA**

Κοττίκας Κωνσταντίνος Γεώργιος

Εξεταστική Επιτροπή:
Δόλλας Απόστολλος, Καθηγητής Π.Κ. (Επιβλέπων)
Πνευματικάτος Διονύσιος, Αναπληρωτής Καθηγητής Π.Κ.
Παπαευσταθίου Ιωάννης, Επίκουρος Καθηγητής Π.Κ.

ΧΑΝΙΑ, 2009

Copyright © Κοττίκας Κωνσταντίνος Γεώργιος, 2009
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ' ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πολυτεχνείου Κρήτης.

Περιεχόμενα

Περιεχόμενα	v
Περίληψη	ix
Ευχαριστίες	xi
Εισαγωγή	1
Λογική Σχεδίαση και Πειράματα	1
Σκοπός και Συνεισφορά της Διπλωματικής Εργασίας	1
Οργάνωση της Διπλωματικής Εργασίας	3
1 Διεξαγωγή Εργαστηρίων Λογικής Σχεδίασης	5
1.1 Στα Ιδρύματα του εξωτερικού	5
1.1.1 Πλατφόρμες Υλικού	7
1.1.2 Πλατφόρμες Λογισμικού	10
1.2 Στα Ιδρύματα του εσωτερικού	12
1.2.1 Πλατφόρμες Υλικού	13
1.2.2 Πλατφόρμες Λογισμικού	14
1.3 Στο Πολυτεχνείο Κρήτης	16
1.3.1 Πλατφόρμα Υλικού	17
1.3.2 Πλατφόρμα Λογισμικού	18
2 Η νέα πλατφόρμα	19
2.1 Η Φυσική Διεπαφή με το Χρήστη στο Breadboard	20
2.1.1 Δομή και Αρχιτεκτονική	21
2.1.2 Λογική Διάταξη	23
2.1.3 Φυσική Διάταξη	32
2.2 Η Λειτουργική Διεπαφή με το Χρήστη στον Η/Υ	35
2.2.1 Δομή και Αρχιτεκτονική	35
2.2.2 Φυσική Διάταξη	38

3	Η Φυσική Διεπαφή με το Χρήστη στο Breadboard	41
3.1	Ο μικροελεγκτής AVR	41
3.2	Τροφοδοσία Τάσης	44
3.3	Ο δίαυλος USB	44
3.4	Τα ολοκληρωμένα I ² C to GPIO	46
3.5	Οι ακροδέκτες Εισόδων	48
3.6	Γεννήτριες Συχνοτήτων	49
3.7	Γεννήτρια Τετραγωνικών Ωρολογιακών Παλμών	51
3.8	Προγραμματισμός Μικροελεγκτή	51
3.9	Δοκιμαστική Πλακέτα Ελέγχου	52
3.10	Εργαλεία Ανάπτυξης Λογισμικού(firmware)	53
3.11	Εργαλεία Σχεδίασης Κυκλώματος	54
3.12	Εκτύπωση και Συναρμολόγηση Τυπωμένου Κυκλώματος	54
4	Η Λειτουργική Διεπαφή με το Χρήστη στον Η/Υ	55
4.1	Ο JAVA LibUSB Wrapper	56
4.2	Οι οδηγοί LibUSB	56
4.3	Εργαλεία Ανάπτυξης	57
5	Επιβεβαίωση Λειτουργίας	59
5.1	Δοκιμές κατά την σχεδίαση και υλοποίηση	59
5.2	Δοκιμές σε πραγματικές συνθήκες	60
6	Συμπεράσματα	61
6.1	Απολογισμός	61
6.2	Μελλοντική Εργασία	61
	Παράρτημα Α	65
	Εγχειρίδια	67
	Εγχειρίδιο Χρήστη	67
	Εγκατάσταση οδηγών(Drivers)	67
	Λογισμικό	80
	Χρήση Αναπτυξιακής Πλακέτας	86
	Εγχειρίδιο Μηχανικού	91
	Φυσική Διεπαφή	91
	Λειτουργική Διεπαφή	91
	Δοκιμαστική Πλακέτα Ελέγχου	92
	Παράρτημα Β	95

Τεχνικά Αρχεία	97
Σχηματικό Διάγραμμα(Schematic)	99
Τυπωμένο Κύκλωμα(PCB)	103
Λίστα εξαρτημάτων	107
Τοποθέτηση Εξαρτημάτων	111
Βιβλιογραφία	117

Περίληψη

Η Λογική Σχεδίαση είναι ο θεμέλιος λίθος στη μελέτη και κατανόηση των βασικών αρχών του Hardware, για κάθε νέο μηχανικό Υπολογιστών. Ο πειραματισμός με λογικά κυκλώματα αποτελεί αναπόσπαστο κομμάτι της εκπαίδευσης του. Για το λόγο αυτό, η ανάπτυξη των πειραμάτων πρέπει να βασίζεται σε εύχρηστα και αξιόπιστα εργαλεία.

Σε αυτή τη διπλωματική εργασία σχεδιάσαμε και υλοποιήσαμε μία αναπτυξιακή πλατφόρμα, η οποία περιλαμβάνει τόσο υλικό όσο και λογισμικό, με σκοπό να παρέχουμε στους μαθητευόμενους μηχανικούς την δυνατότητα χρήσης πολλών πόρων, κατά την πειραματική διαδικασία και την εφαρμογή των γνώσεων που απόκτησαν κατά τη διδασκαλία του μαθήματος.

Η πλατφόρμα περιλαμβάνει πληθώρα Εξόδων(Outputs), όπως Διακόπτες SPDT (Single Pole, Double Throw), Δεκαεξαδικό πληκτρολόγιο, Μετρητή 8-bits, Γεννήτρια Ψευδοτυχαίων Αριθμών 8-bits (Galois Linear Feedback Shift Register), καθώς επίσης και Εισόδων(Inputs), όπως Φωτοдиодοι(LEDs) και Οθόνες 7 στοιχείων(Seven Segments Displays). Ακόμα, δύο γεννήτριες συχνοτήτων, μία του 1Hz και μία των 10KHz, καθώς επίσης και μία γεννήτρια τετραγωνικών παλμών της τάξης των 5KHz, η οποία πυροδοτείται με το πάτημα ενός κουμπιού. Τέλος, έχει γίνει πρόβλεψη, για "USB to Serial" επικοινωνία μεταξύ του H/Y και σειριακών συσκευών, όπως π.χ. FPGAs.

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον καθηγητή μου κ. Απόστολο Δόλλα, επιβλέποντα της παρούσας διπλωματικής εργασίας, για την εποικοδομητική συνεργασία που είχαμε στα πλαίσια της διπλωματικής εργασίας μου και την μοναδική ευκαιρία που μου έδωσε για την ολοκλήρωση αυτού του έργου. Επίσης, θα ήθελα να ευχαριστήσω τον κ. Διονύσιο Πνευματικάτο, Αναπληρωτή Καθηγητή, και τον κ. Ιωάννη Παπαευσταθίου, Επίκουρο Καθηγητή, για τη συμμετοχή τους ως μέλη της εξεταστικής επιτροπής.

Ένα μεγάλο ευχαριστώ στον διδακτορικό φοιτητή Δημήτρη Μεϊντάνη και τον μεταπτυχιακό φοιτητή Διονύση Ευσταθίου για τη συνεργασία τους, τις συμβουλές τους και την πολύτιμη βοήθεια τους για την περάτωση της διπλωματικής μου εργασίας.

Επίσης, ευχαριστώ θερμά τον κ. Μάρκο Κιμιωνή, μέλος ΕΕΔΙΠ και υπεύθυνο του εργαστηρίου Μικροεπεξεργαστών και Υλικού, για την υποστήριξή του και την ταχεία εξυπηρέτηση όταν χρειάστηκε να εφοδιαστώ με τον κατάλληλο υλικοτεχνικό εξοπλισμό.

Δεν γίνεται να παραλείψω να ευχαριστήσω τους φίλους μου, εντός και εκτός του εργαστηρίου Μικροεπεξεργαστών και Υλικού, που μου συμπαραστάθηκαν όλο αυτό τον καιρό της φοίτησής μου.

Περισσότερο απ' όλους, όμως, θα ήθελα να ευχαριστήσω τους γονείς μου Στέφανο και Μαρίνα και την αδερφή μου Ελίζα που είναι πάντα δίπλα μου σε όποια απόφαση και αν πάρω και την Ελένη που με συντρόφεψε και με στήριξε στις ευχάριστες αλλά και δυσάρεστες στιγμές.

Εισαγωγή

Λογική Σχεδίαση και Πειράματα

Η Λογική Σχεδίαση είναι ο θεμέλιος λίθος στη μελέτη και κατανόηση των βασικών αρχών του Hardware. Εισάγει κάθε νέο μηχανικό Υπολογιστών στα βασικά δομικά στοιχεία της "Λογικής", δηλαδή τις πύλες(Gates). Ο μηχανικός, έπειτα, μαθαίνει να χρησιμοποιεί αυτά τα δομικά στοιχεία, για την κατασκευή *συνδυαστικών* (combinational) λογικών κυκλωμάτων. Ακόμα, η Λογική Σχεδίαση εξηγεί έννοιες, όπως το δυαδικό/οκταδικό/δεκαεξαδικό σύστημα αναπαράστασης των αριθμών, την Άλγεβρα Boole, τους πίνακες Karnaugh, κ.ο.κ. Αναλύει τα αριθμητικά κυκλώματα, όπως τους αθροιστές/αφαιρέτες, και εισάγει τους νέους μηχανικούς στα ολοκληρωμένα κυκλώματα TTL, τους αποκωδικοποιητές, τους πολυπλέκτες, τους συγκριτές, τους μανδαλωτές, τους καταχωρητές (flip-flop), και άλλα. Απώτερος στόχος της διδασκαλίας του μαθήματος, είναι ο νέος μηχανικός να μπορεί να αναλύει και να σχεδιάζει, απλά αλλά και σύνθετα, ακολουθιακά κυκλώματα, μηχανές πεπερασμένων καταστάσεων, κ.λ.π.

Ο μηχανικός θα πρέπει να διδαχθεί, να κατανοήσει, και τελικά να εφαρμόσει, μέσα από την πειραματική διαδικασία, όλες τις γνώσεις που αποκόμισε. Ο πειραματισμός με λογικά κυκλώματα αποτελεί αναπόσπαστο κομμάτι της εκπαίδευσης του. Για το λόγο αυτό, η ανάπτυξη των πειραμάτων πρέπει να βασίζεται σε εύχρηστα και αξιόπιστα εργαλεία. Εκεί ακριβώς στόχευσε η παρούσα διπλωματική εργασία.

Σκοπός και Συνεισφορά της Διπλωματικής Εργασίας

Σε αυτή τη διπλωματική εργασία προσπαθήσαμε να κατασκευάσουμε μια αναπτυξιακή πλατφόρμα, η οποία θα παρέχει στο μηχανικό, την ελευθερία της χρήσης, πολλών και διαφόρων, Εισόδων(Inputs) και Εξόδων(Outputs), σε κάθε Λογικό κύκλωμα που σχεδιάζει. Ο χρήστης, αφού σχεδιάσει και υλοποιήσει το λογικό κύκλωμα στο Breadboard, έχει την ευχέρεια να το τροφοδοτήσει με όποια Είσοδο και να λάβει αποτελέσματα σε όποια Έξοδο τον ικανοποιεί. Επίσης, έμφαση δό-

θηκε στη φορητότητα της πλατφόρμας. Επιθυμούσαμε, ο μαθητευόμενος μηχανικός, να έχει τη δυνατότητα πειραματισμού εκτός των ορίων του εργαστηρίου. Η πλατφόρμα αποτελείται από δύο μέρη: μια συσκευή που ονομάζουμε **”MHL Development Board”** (υλικό) και μια γραφική διεπαφή (λογισμικό) που ονομάζουμε **”MHL Development Board Simulator”**.

Η συσκευή ή αλλιώς αναπτυξιακή πλακέτα είναι το κομμάτι που παρέχει στο χρήστη τη φυσική επαφή με τις Εισόδους/Εξόδους της πλατφόρμας. Πιο αναλυτικά, επάνω στην πλακέτα βρίσκονται ακροδέκτες που ο καθένας αντιστοιχεί είτε σε μια Είσοδο, είτε σε μια Έξοδο. Οι Είσοδοι(Inputs) μπορεί να είναι Φωτοδίοδοι(LEDs) και Οθόνες 7 στοιχείων(Seven Segments Displays), ενώ οι Έξοδοι(Outputs) μπορεί να είναι Διακόπτες S.P.D.T.(Single Pole, Double Throw), ένα Δεκαεξαδικό πληκτρολόγιο, ένας Μετρητής 8-bits ή μία Γεννήτρια Ψευδοτυχαίων Αριθμών 8-bits (Galois Linear Feedback Shift Register)[26]. Βέβαια, οι δομικές μονάδες που εκφράζονται από περισσότερα του ενός bit καταλαμβάνουν και τον αντίστοιχο αριθμό ακροδεκτών. Ακόμα, στην πλακέτα βρίσκονται δύο γεννήτριες συχνοτήτων, μία του 1Hz και μία των 10KHz, καθώς επίσης και μία γεννήτρια τετραγωνικών παλμών της τάξης των 5KHz, η οποία πυροδοτείται με το πάτημα ενός πιεστικού διακόπτη. Τέλος, έχει γίνει πρόβλεψη επικοινωνίας μεταξύ του H/Y και σειριακών συσκευών, όπως π.χ. FPGAs.

Όλες οι Είσοδοι και Έξοδοι μπορούν να λειτουργούν παράλληλα και να μεταβάλλονται σε πραγματικό χρόνο (Live Insertion)[24].

Η γραφική διεπαφή(λογισμικό) αποτελεί το δεύτερο μέρος της πλατφόρμας. Πρόκειται για ένα προσομοιωτή μιας αναπτυξιακής πλακέτας(με φυσική υποστήριξη). Στη γραφική διεπαφή ο χρήστης δύναται να μεταβάλλει κάθε Έξοδο και να παρατηρεί όλες τις αλλαγές των Εισόδων. Όπως γίνεται κατανοητό, κάθε ακροδέκτης της πλακέτας αντιστοιχεί σε κάποια Είσοδο/Έξοδο του λογισμικού.

Η επικοινωνία μεταξύ της γραφικής διεπαφής και της συσκευής επιτυγχάνεται με τη χρήση διασύνδεσης USB διαύλου. Οι τιμές των bits των Εξόδων παράγονται από το λογισμικό, το οποίο με τη σειρά του αποστέλλει τα δεδομένα μέσω της USB θύρας στην πλακέτα. Στην τελευταία, ένας AtmelTM AVR[22] μικροελεγκτής αναλαμβάνει να διαμοιράσει τα δεδομένα στους ακροδέκτες εξόδων. Ο ίδιος AVR είναι υπεύθυνος και για τη συλλογή των δεδομένων εισόδων και την αποστολή τους πίσω στο λογισμικό, κατά απαίτηση του τελευταίου.

Οργάνωση της Διπλωματικής Εργασίας

Μετά την ανάθεση της διπλωματικής, προχωρήσαμε στην οργάνωση και στη κατάταξη της σε επιμέρους κομμάτια.

Αρχικά, ορίσαμε τις προδιαγραφές της υλοποίησής μας. Αποφασίσαμε πόσες και ποιες δομικές μονάδες θα περιελάμβανε η πλατφόρμα μας, καθώς και τον αριθμό των Εισόδων και Εξόδων, δηλαδή πόσα LEDs θα υπήρχαν, πόσοι διακόπτες, κ.ο.κ. Ακόμα, επιλέξαμε την JAVA ως γλώσσα προγραμματισμού της γραφικής εφαρμογής μας και προσανατολιστήκαμε προς τον AVR μικροελεγκτή για την διασύνδεση του software με το hardware.

Εν συνεχεία, μοντελοποιήσαμε το hardware στη JAVA με σκοπό να διαπιστώσουμε τους υλικοτεχνικούς πόρους που χρειαζόμασταν. Το στάδιο αυτό υπήρξε διερευνητικό, αλλά σίγουρα καθοριστικό για την εξέλιξη της δουλειάς μας.

Ακολούθησε ο σχεδιασμός του hardware. Πρώτα, πειραματιστήκαμε με τα επιμέρους κομμάτια του υλικού. Υλοποιήσαμε την USB διασύνδεση και επιβεβαιώσαμε την λειτουργία της ανταλλάσσοντας τυχαία πακέτα δεδομένων μεταξύ H/Y και πλακέτας. Στη συνέχεια, υλοποιήσαμε την υποστήριξη μεγάλου αριθμού εισόδων/εξόδων, με τη χρήση I^2C to $GPIO$ [25, 23] ολοκληρωμένων κυκλωμάτων, δημιουργήσαμε τις γεννήτριες συχνοτήτων και, τελικά, σχεδιάσαμε το τελικό τυπωμένο κύκλωμα και την πρώτη έκδοση του λογισμικού(Firmware) του μικροελεγκτή συνδυάζοντας όλα τα προηγούμενα.

Ένα πρωτότυπο της πλακέτας τυπώθηκε στο εργαστήριο, συναρμολογήθηκε και τελικά συνδέθηκε με το γραφικό περιβάλλον. Ακολούθησε η αποσφαλμάτωση του λογισμικού, η βελτίωση του Firmware του AVR και, τέλος, η επιβεβαίωση λειτουργίας της πλατφόρμας μέσα από μια σειρά πειραματικών κυκλωμάτων.

Κεφάλαιο 1

Διεξαγωγή Εργαστηρίων Λογικής Σχεδίασης

Στο κεφάλαιο αυτό, θα δώσουμε μια εικόνα του περιεχομένου των εργαστηριακών ασκήσεων του μαθήματος της Λογικής Σχεδίασης σε διάφορα τμήματα του εξωτερικού και του εσωτερικού, καταλήγοντας στο Τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών του Πολυτεχνείου Κρήτης. Χρήσιμο θα ήταν να αναφέρουμε πως πηγή των πληροφοριών μας αποτέλεσε το διαδίκτυο.

1.1 Στα Ιδρύματα του εξωτερικού

Στα Πανεπιστήμια του εξωτερικού, η Λογική Σχεδίαση περιλαμβάνεται πάντα στο πρόγραμμα προπτυχιακών σπουδών κάθε τμήματος σχετικό με Η/Υ. Αν και σε όλα τα τμήματα οι στόχοι που θέτονται στα πλαίσια της διδασκαλίας του μαθήματος παραμένουν ίδιοι σε γενικές γραμμές, ωστόσο, η προσέγγιση των εκπαιδευτικών πειραμάτων και της πειραματικής διαδικασίας ποικίλουν και σε αρκετές περιπτώσεις αποκλίνουν αρκετά μεταξύ τους.

Πιο αναλυτικά, μετά τη σχετική έρευνα που πραγματοποιήσαμε, διαπιστώσαμε πως σε κάποια τμήματα οι εργαστηριακές ασκήσεις πραγματοποιούνταν με την χρήση αποκλειστικά και μόνο πυλών(ολοκληρωμένα κυκλώματα) και άλλου ηλεκτρονικού υλικού (breadboards, LEDs, Διακόπτες, κ.ο.κ.), σε άλλα μόνο με χρήση γλωσσών περιγραφής υλικού(VHDL, Verilog) ή λογισμικού προσομοίωσης κυκλωμάτων, και σε άλλα με τον συνδυασμό των δύο προηγούμενων.

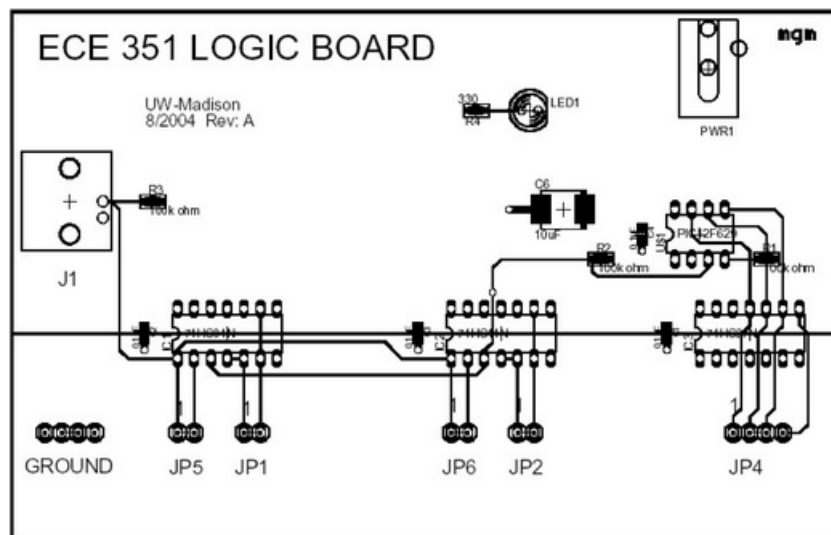
Τα πειράματα που αναθέτονται στους φοιτητές των πανεπιστημίων του εξωτερικού, στα πλαίσια του μαθήματος της Λογικής Σχεδίασης, εστιάζουν και περιλαμβάνουν τα παρακάτω:

- Δυαδικό, Οκταδικό, Δεκαδικό, Δεκαεξαδικό, BCD αριθμητικό σύστημα [18, 17, 9, 19, 1].
- Ομοιότητες/Διαφορές TTL και CMOS πυλών [18, 16].
- Ελαχιστοποίηση Boolean Συναρτήσεων [18, 17, 4, 14, 19, 1, 6, 16].
- Πίνακες Karnaugh [4, 14, 16].
- Κώδικες Gray [17].
- Συνδυαστικά κυκλώματα με πύλες [17, 4, 9, 14, 1].
- Ασύγχρονα Ακολουθιακά κυκλώματα με πύλες [18, 17, 1].
- Κυκλώματα Αθροιστών/Αφαιρετών/Πολλαπλασιαστών και Αριθμητικών και Λογικών μονάδων(ALU) [17, 9, 6, 16, 13].
- Πολυπλέκτες, Αποπλέκτες, Κωδικοποιητές, Αποκωδικοποιητές [18, 17, 14, 6, 16].
- Latches, Flip-Flops(D,J-K,T), Καταχωρητές [18, 17, 4, 9, 16].
- Σύγχρονες και Ασύγχρονες Μηχανές Πεπερασμένων Καταστάσεων(FSM) [17, 4, 14, 1, 6].
- Κυκλώματα Μετρητών [9, 19, 6, 16].

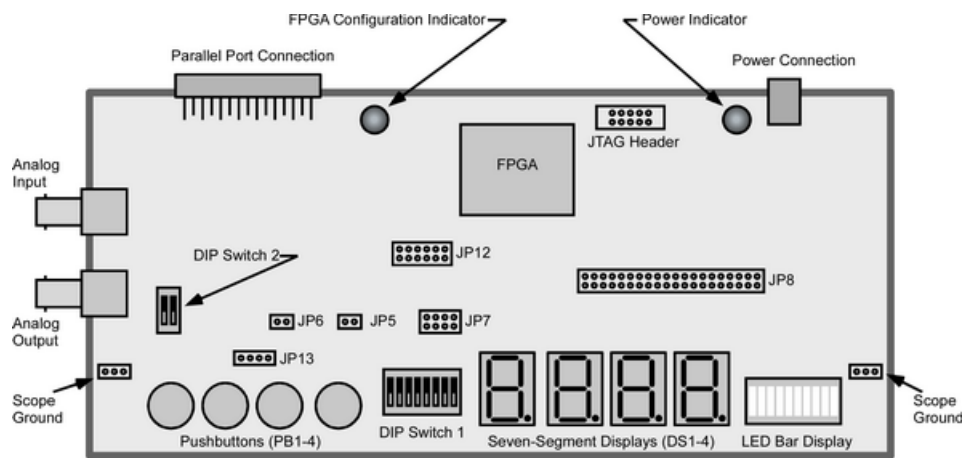
Το σύνολο των εργαστηριακών ασκήσεων που εντοπίσαμε δεν παρουσιάζουν ιδιαίτερο ενδιαφέρον ως προς την πρωτοτυπία του περιεχομένου. Με τον όρο *πρωτοτυπία* εννοούμε τον τρόπο με τον οποίο οι φοιτητές εφαρμόζουν την διδαχθείσα θεωρία διαμέσου των εργαστηριακών ασκήσεων. Ωστόσο, κάποιες εργαστηριακές ασκήσεις, πραγματικά, μας κίνησαν το ενδιαφέρον και μας εντυπωσίασαν. Χαρακτηριστικό παράδειγμα είναι το γνωστό σε όλους παιχνίδι "*Πέτρα, Ψαλίδι, Χαρτί*". Αυτή η εργαστηριακή άσκηση θέλοντας να παρουσιάσει στους φοιτητές τις έννοιες της BCD κωδικοποίησης, των μετρητών και των ακολουθιακών και συνδυαστικών κυκλωμάτων τούς ζητά να σχεδιάσουν και να υλοποιήσουν ένα κύκλωμα που να αποφασίζει ποιος από τους δύο παίκτες, ανάλογα με τις επιλογές που έχουν κάνει στις εισόδους του κυκλώματος ("*Πέτρα, Ψαλίδι, Χαρτί*"), κερδίζει την κάθε παρτίδα και να κρατάει σκορ για τον κάθε παίκτη ξεχωριστά[17]. Ακόμα, συναντήσαμε ασκήσεις όπως κυκλώματα συναγερμών, ελέγχου σηματοδοτών, σειριακής επικοινωνίας, ελέγχου robot, αισθητήρων και άλλα.

1.1.1 Πλατφόρμες Υλικού

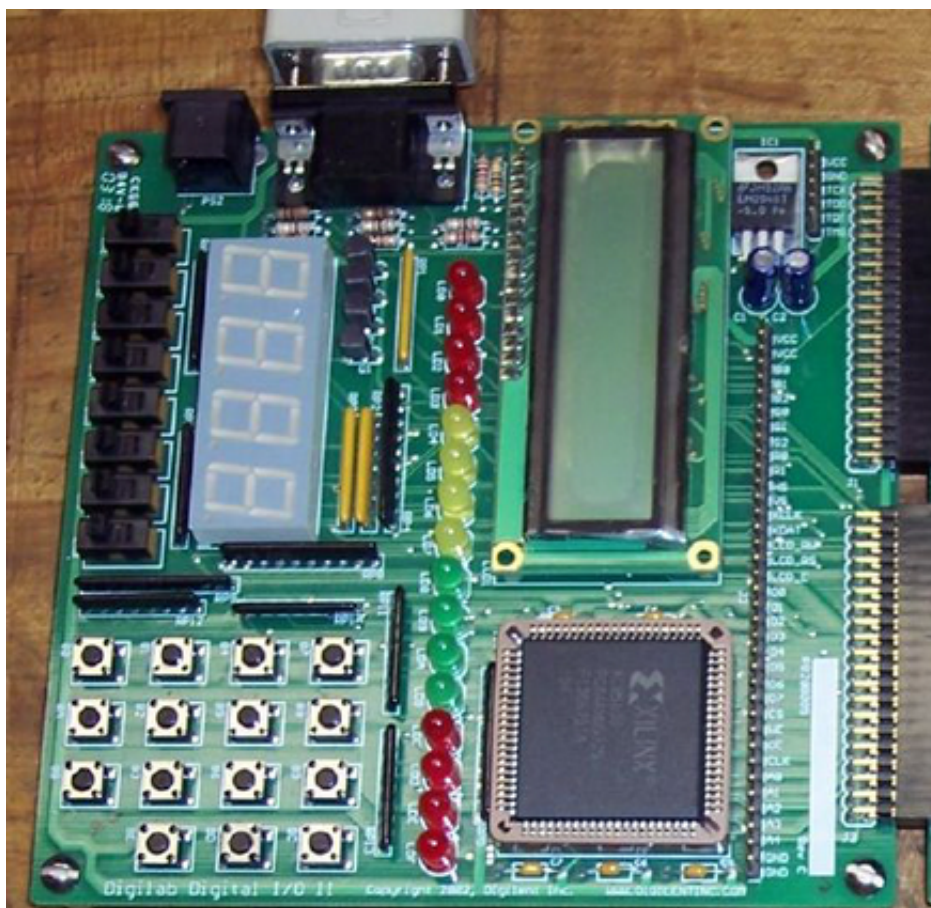
Στην έρευνά μας δεν θα μπορούσαμε να παραλείψουμε το υλικοτεχνικό πλαίσιο που εκτελούνταν οι εργαστηριακές ασκήσεις. Το σύνολο των ιδρυμάτων που συμπεριελάμβανε υλικό για την εκτέλεση των πειραμάτων, χρησιμοποιούσε breadboards και εξαρτήματα όπως αντιστάσεις, LEDs, οθόνες 7 στοιχείων, ολοκληρωμένα κυκλώματα πυλών, κ.ο.κ. Μόνο σε ένα ίδρυμα[19] συναντήσαμε κάποια αναπτυξιακή πλατφόρμα κατασκευασμένη εντός του πανεπιστημίου, η οποία δεν προσέφερε κάποια ουσιαστική ευκολία σε εισόδους ή εξόδους, παρά μόνον διευκόλυνε στην χρήση κάποιας γεννήτριας συχνότητας. Όσον αφορά τα ιδρύματα που ως κύριο εργαλείο ανάπτυξης χρησιμοποιούσαν κάποια γλώσσα περιγραφής υλικού, εκεί συναντήσαμε boards με FPGAs της XilinxTM και της AlteraTM και πιο συγκεκριμένα FPGAs XilinxTM Spartan 2, Spartan 3 και AlteraTM Cyclone. Ακολουθούν ενδεικτικές φωτογραφίες:



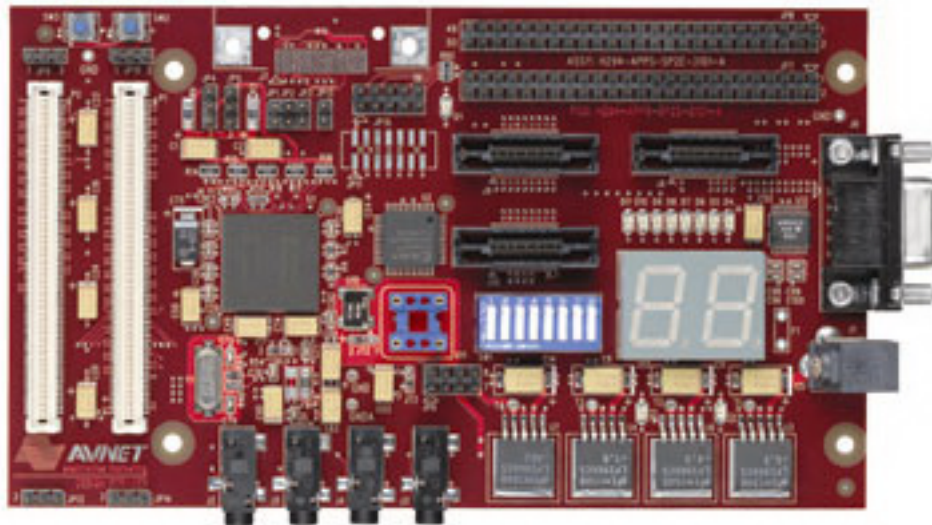
Σχήμα 1.1: Αναπτυξιακή πλακέτα, Πανεπιστήμιο του Wisconsin



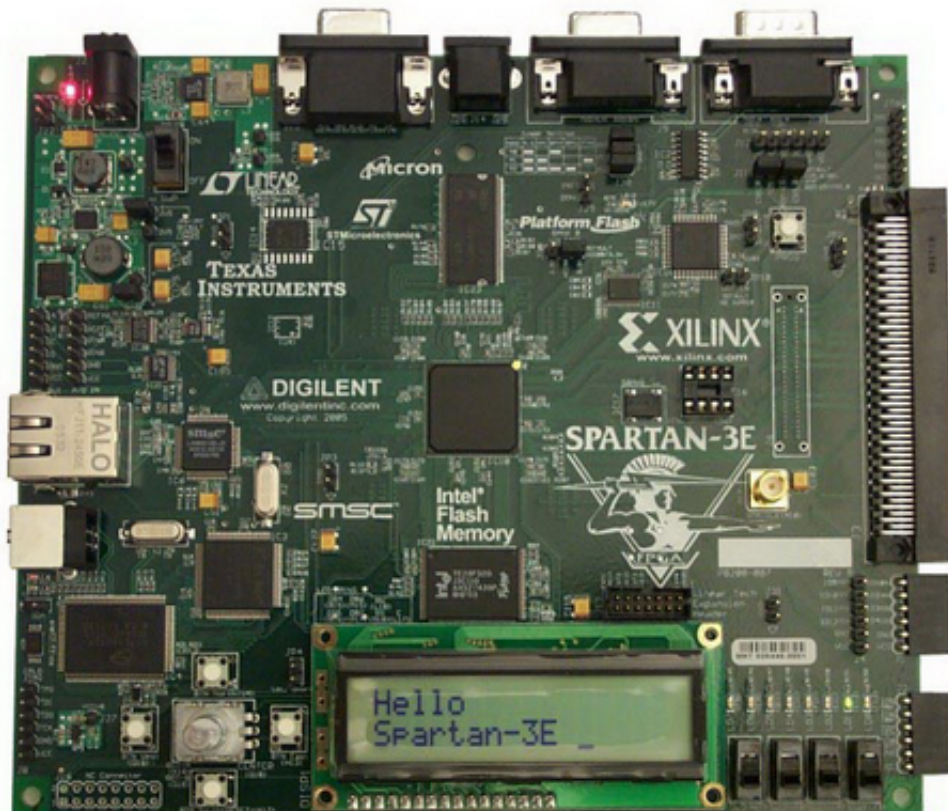
Σχήμα 1.2: *FPGA Altera Cyclone*



Σχήμα 1.3: *FPGA Spartan 2*



Σχήμα 1.4: *FPGA Spartan 2*

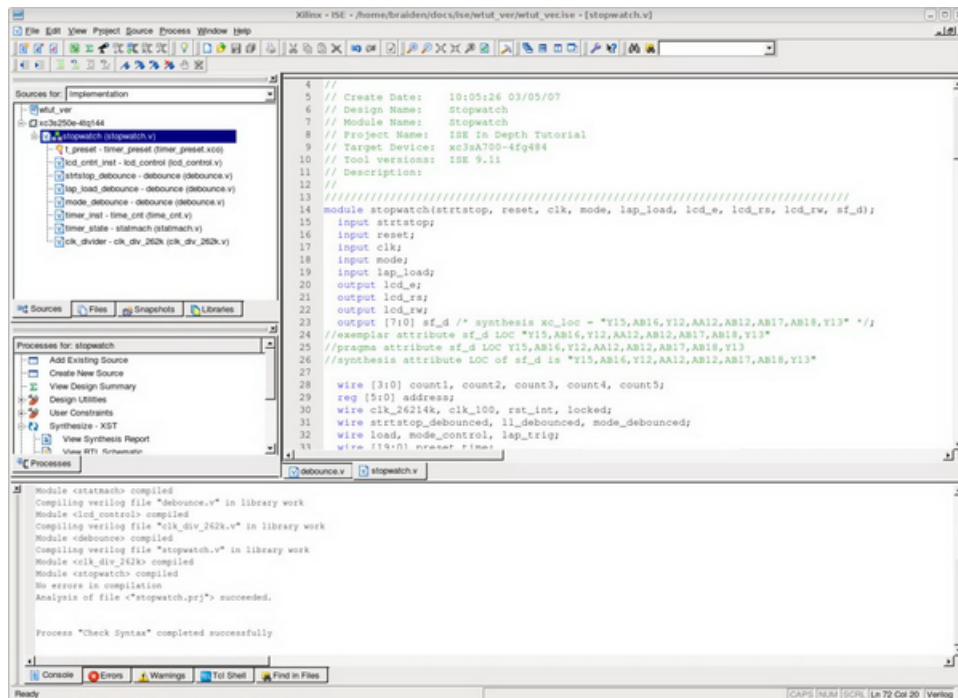


Σχήμα 1.5: *FPGA Spartan 3*

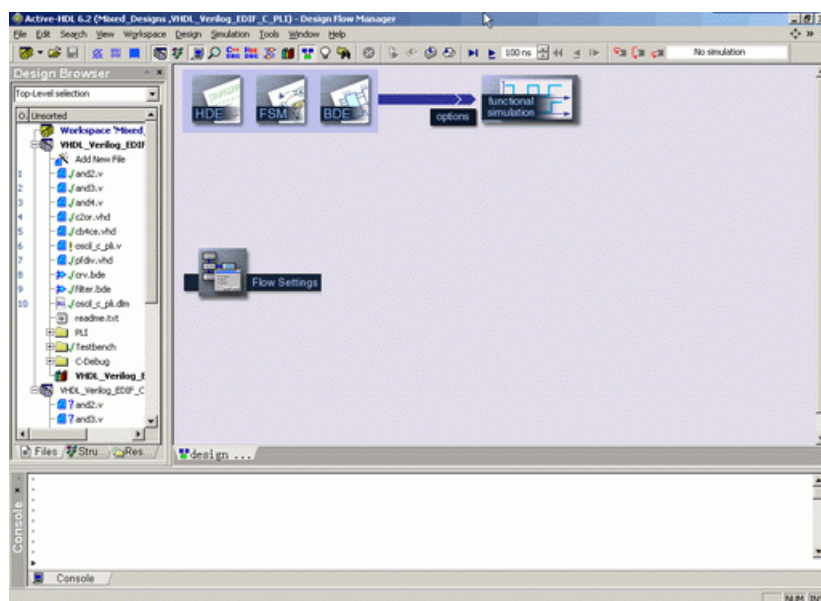
1.1.2 Πλατφόρμες Λογισμικού

Το λογισμικό που χρησιμοποιείται κατά κόρον στα πανεπιστημιακά ιδρύματα που χρησιμοποιούν FPGAs για την διεξαγωγή των εργαστηριακών ασκήσεων είναι το Xilinx ISE και, ενίοτε, σε συνδυασμό με το Mentor Graphics ModelSim. Ακόμα, συναντήσαμε FPGAs της Altera (Altera Cyclone) σε συνδυασμό με τον αντίστοιχο compiler που προσφέρει η εταιρεία και, τέλος, τον compiler Aldic Active-HDL.

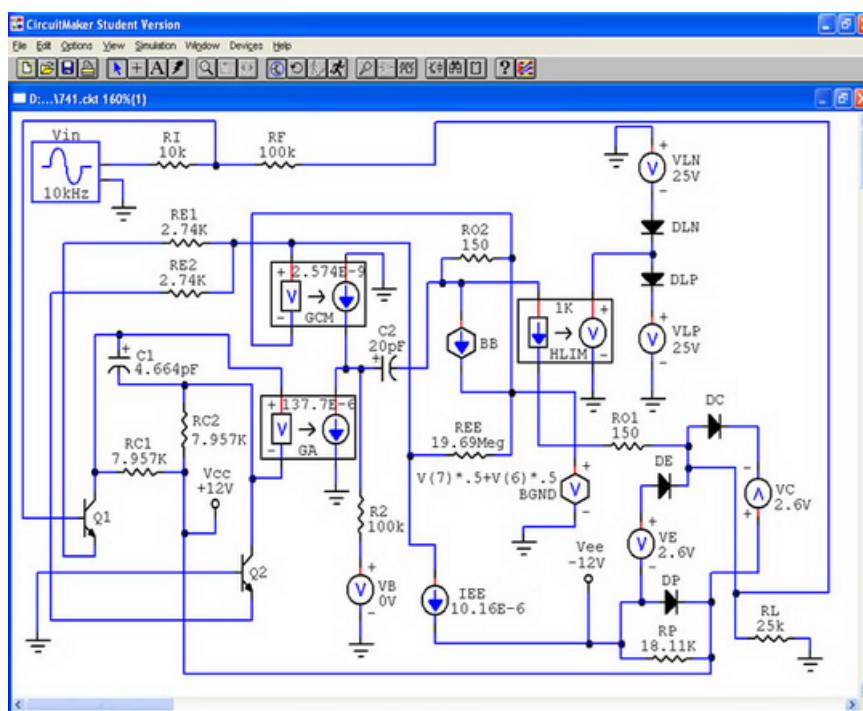
Μικρός είναι ο αριθμός των ιδρυμάτων που χρησιμοποιούν για την εκτέλεση της πειραματικής διαδικασίας προσομοιωτές hardware. Ο προσομοιωτής που συναντήσαμε είναι ο Altium Circuit Maker. Ακολουθούν ενδεικτικές φωτογραφίες:



Σχήμα 1.6: Xilinx ISE



Σχήμα 1.7: *Aldic Active-HDL*



Σχήμα 1.8: *Altium Circuit Maker*

1.2 Στα Ιδρύματα του εσωτερικού

Τα προπτυχιακά προγράμματα σπουδών σε όλα τα ελληνικά πανεπιστημιακά ιδρύματα περιλαμβάνουν το μάθημα της Λογικής Σχεδίασης. Όπως στα ιδρύματα του εξωτερικού, έτσι και σε αυτά του εσωτερικού ο τρόπος διεξαγωγής και το περιεχόμενο των εργαστηριακών ασκήσεων ποικίλει. Ωστόσο, οι θεματικές ενότητες που πραγματεύονται τα εργαστήρια των διαφόρων τμημάτων περιστρέφονται γύρω από τον ίδιο θεωρητικό άξονα. Η έρευνά μας έδειξε πως οι θεματικές ενότητες αυτές είναι οι παρακάτω:

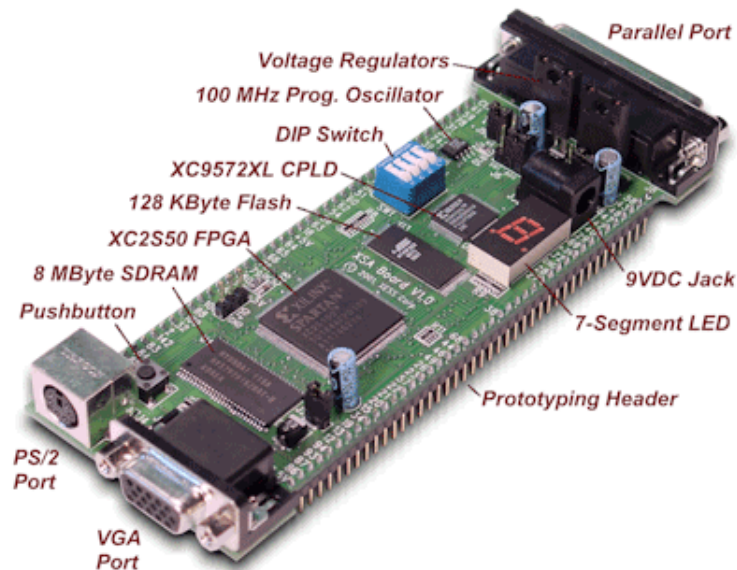
- Δυαδικό, Οκταδικό, Δεκαδικό, Δεκαεξαδικό, BCD αριθμητικό σύστημα [28, 30, 31, 29, 27].
- Ελαχιστοποίηση Boolean Συναρτήσεων [28, 30, 31, 29, 27].
- Πίνακες Karnaugh [31, 29].
- Κώδικες Gray [29].
- Συνδυαστικά κυκλώματα με πύλες [30, 31, 27].
- Ασύγχρονα Ακολουθιακά κυκλώματα με πύλες [30].
- Κυκλώματα Αθροιστών/Αφαιρετών/Πολλαπλασιαστών και Αριθμητικών και Λογικών μονάδων(ALU) [28, 30, 29].
- Πολυπλέκτες, Αποπλέκτες, Κωδικοποιητές, Αποκωδικοποιητές [28, 31, 29, 27].
- Latches, Flip-Flops(D,J-K,T), Καταχωρητές [28, 30, 31, 29].
- Σύγχρονες και Ασύγχρονες Μηχανές Πεπερασμένων Καταστάσεων(FSM) [27].
- Κυκλώματα Μετρητών [30, 31, 29, 27].
- Κυκλώματα βασισμένα σε P.L.D.s [27].

Το περιεχόμενο των ασκήσεων περιορίζεται σε απλές εφαρμογές. Δεν συναντήσαμε ασκήσεις που να μας κέντρισαν ιδιαίτερα το ενδιαφέρον όπως σε αντίστοιχα ιδρύματα του εξωτερικού. Ωστόσο, οφείλουμε να αναφέρουμε πως το Εθνικό Μετσόβιο Πολυτεχνείο διαφοροποιείται σε σχέση με τα υπόλοιπα ιδρύματα υλοποιώντας μεταξύ άλλων και ασκήσεις όπως κυκλώματα πλειοψηφίας, παλμογεννήτριες ρολογιού και κυκλώματα διαύλων δεδομένων.

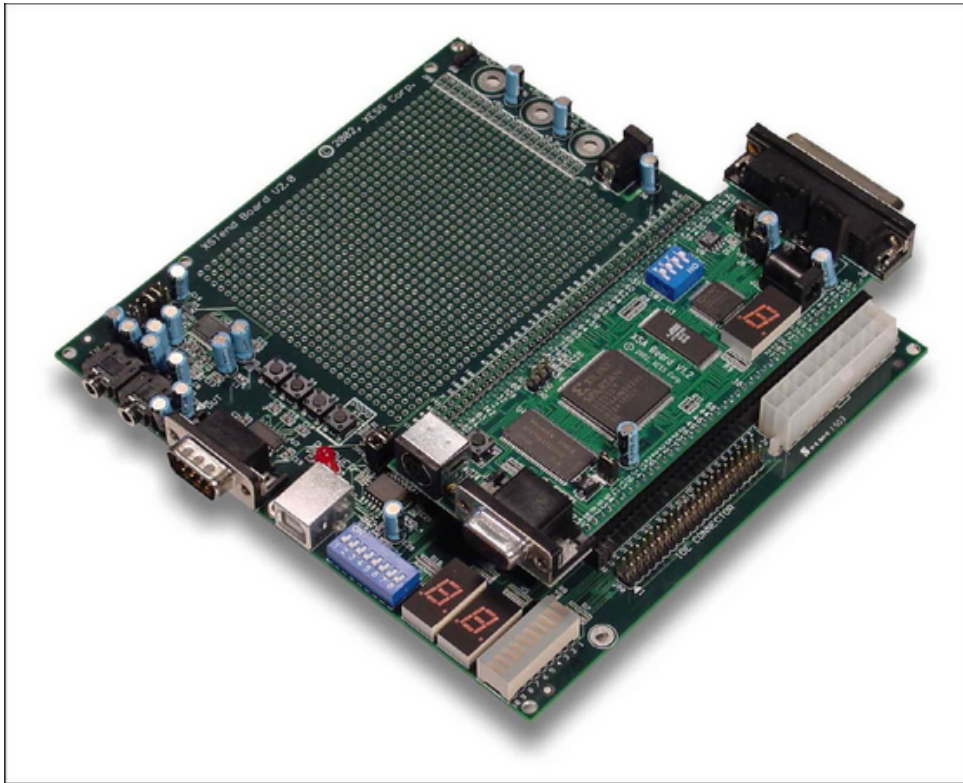
Ακόμα, οφείλουμε να αναφερθούμε στα μέσα που χρησιμοποιούνται για την εκτέλεση των εργαστηριακών ασκήσεων. Διαπιστώσαμε πως σε κάποια τμήματα οι εργαστηριακές ασκήσεις πραγματοποιούνταν με την χρήση αποκλειστικά και μόνο πυλών(ολοκληρωμένα κυκλώματα) και άλλου ηλεκτρονικού υλικού (bread-boards, LEDs, Διακόπτες, κ.ο.κ.), σε άλλα μόνο με χρήση γλωσσών περιγραφής υλικού (VHDL) ή λογισμικού προσομοίωσης κυκλωμάτων, και σε άλλα με τον συνδυασμό των δύο προηγούμενων. Στις επόμενες παραγράφους, θα αναφερθούμε πιο αναλυτικά στον εργαστηριακό εξοπλισμό.

1.2.1 Πλατφόρμες Υλικού

Οι πλατφόρμες υλικού που χρησιμοποιούνται από τα ελληνικά ιδρύματα περιορίζονται σε βασικά δομικά υλικά για κυκλώματα, δηλαδή σε ολοκληρωμένα πυλών, LEDs, διακόπτες, κ.λ.π. Στο Εθνικό Μετσόβιο Πολυτεχνείο γίνεται χρήση PLDs και μόνο το πανεπιστήμιο Αιγαίου χρησιμοποιεί FPGA της Xilinx και πιο συγκεκριμένα το board Xilinx XSA-50 που περιλαμβάνει Spartan 2 FPGA και ένα CPLD σε συνδυασμό με την πλακέτα XST 2.x της ίδιας εταιρίας, η οποία διευρύνει την πλακέτα της FPGA παρέχοντας USB , RS-232, IDE και Stereo Sound Input/Output διεπαφές. Ακολουθούν ενδεικτικές φωτογραφίες:



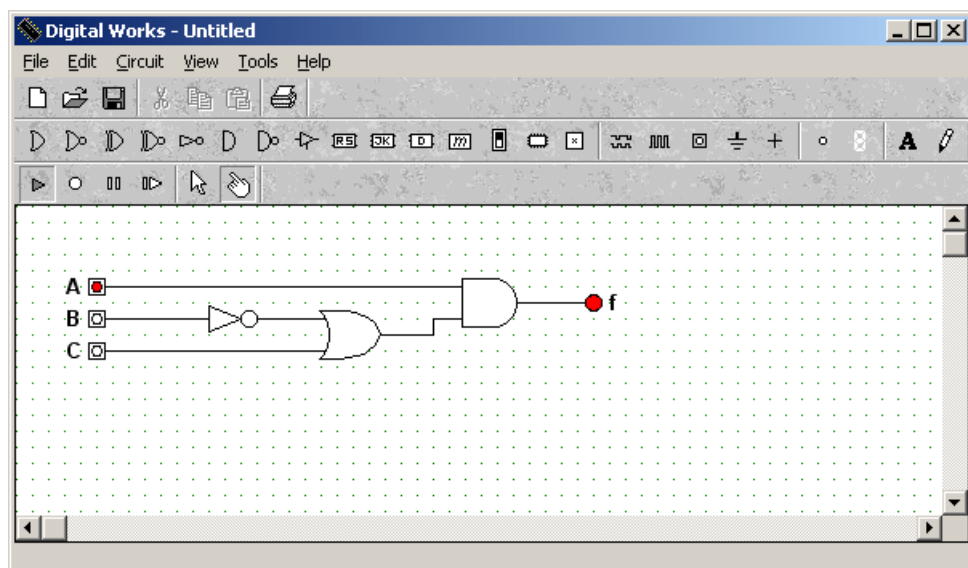
Σχήμα 1.9: Xilinx XSA-50 FPGA Board



Σχήμα 1.10: *Xilinx XST 2.x FPGA Expansion Board*

1.2.2 Πλατφόρμες Λογισμικού

Οι πλατφόρμες λογισμικού, που χρησιμοποιούνται από τα ελληνικά ιδρύματα, περιορίζονται στις εξής: το πανεπιστήμιο Αιγαίου κάνει χρήση του Xilinx ISE σε συνδυασμό με το αντίστοιχο hardware της εταιρείας, όπως περιγράψαμε στην προηγούμενη παράγραφο, ενώ το Εθνικό Μετσόβιο Πολυτεχνείο δουλεύει σε κάποιο compiler για το PLD που χρησιμοποιεί. Άξιο ενδιαφέροντος είναι το Mecanique Digital Works που χρησιμοποιεί το Πανεπιστήμιο Πειραιά. Πρόκειται για προσομοιωτή hardware, ικανό να προσομοιώνει κυκλώματα σχεδιασμένα από τους χρήστες. Ακολουθούν ενδεικτική φωτογραφίες:



Σχήμα 1.11: *Mecanique Digital Works*

1.3 Στο Πολυτεχνείο Κρήτης

Στο τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών του Πολυτεχνείου Κρήτης, το μάθημα της Λογικής Σχεδίασης διδάσκεται κατά το πρώτο έτος των προπτυχιακών σπουδών και αποτελεί πρόδρομος των υπολοίπων μαθημάτων του τομέα Ηλεκτρονικής και Αρχιτεκτονικής Υπολογιστών και ειδικότερα του Εργαστηρίου Μικροεπεξεργαστών και Υλικού (EMY).

Οι εργαστηριακές ασκήσεις της Λογικής Σχεδίασης δίνουν στο φοιτητή τη δυνατότητα εφαρμογής της θεωρίας μέσα από απλές εφαρμογές χρησιμοποιώντας λογικές πύλες, καθώς επίσης και συσκευές PLD. Ο αριθμός των εργαστηριακών ασκήσεων είναι πέντε και εστιάζει στις εξής θεματικές ενότητες:

- **Εξοικείωση με την συνδυαστική λογική TTL:** Η άσκηση περιλαμβάνει τη γνωριμία με τα λογικά κυκλώματα TTL, τον πειραματισμό με πύλες και την κατασκευή της συνάρτησης XOR(με δύο μεταβλητές ως $\bar{x}y + x\bar{y}$).
- **Πίνακες Karnaugh, πολλαπλές Έξοδοι και απεικόνιση σε διαθέσιμες πύλες:** Η άσκηση περιλαμβάνει την κατασκευή κυκλώματος συνάρτησης κώδικα Gray τριών εισόδων και τριών εξόδων, κάνοντας χρήση πινάκων Karnaugh και άλγεβρας Boole για την ελαχιστοποίηση της συνάρτησης.
- **Λογική Σχεδίαση με επαναπρογραμματιζόμενα κυκλώματα PAL/GAL, γλώσσα περιγραφής υλικού (HDL):** Στην άσκηση αυτή, οι φοιτητές υλοποιούν την δεύτερη εργαστηριακή άσκηση σε γλώσσα περιγραφής υλικού κάνοντας χρήση ενός PLD και του compiler PLDShell της Altera.
- **Λογική Σχεδίαση με Flip-Flop, σύγχρονα ακολουθιακά κυκλώματα και μετρητές:** Σκοπός της άσκησης είναι η εξοικείωση με το D Flip-Flop και η κατασκευή ενός μετρητή.
- **Μηχανές Πεπερασμένων Καταστάσεων (FSM), σχεδιασμένες με λογική PAL/GAL:** Σκοπός της εργαστηριακής άσκησης είναι η σχεδίαση και κατασκευή μιας FSM, η οποία υλοποιεί το κύκλωμα ελέγχου των φώτων κατεύθυνσης και κινδύνου ενός αυτοκινήτου.

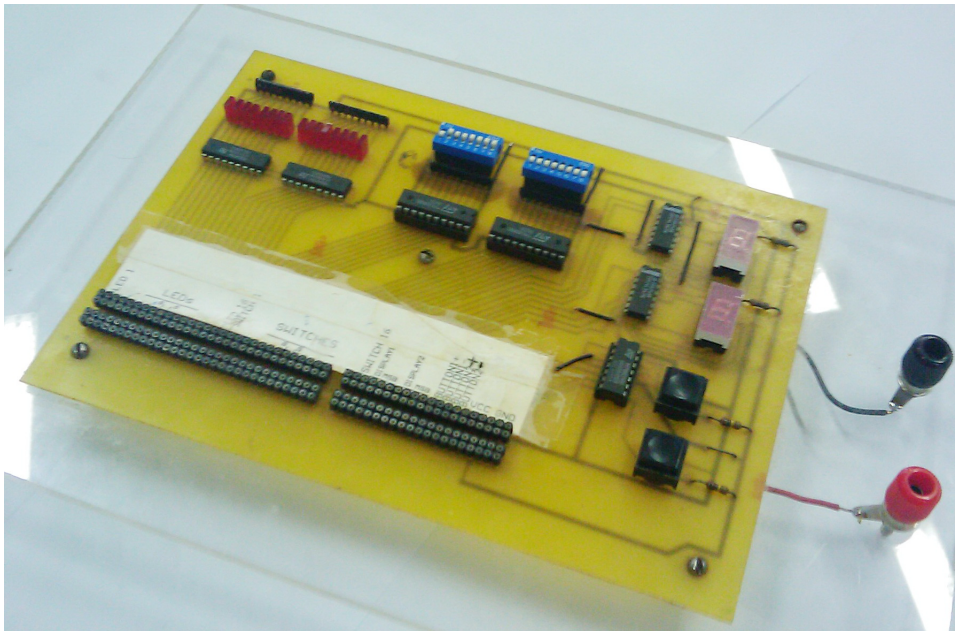
Το περιεχόμενο των εργαστηριακών ασκήσεων δεν διαφοροποιείται από αυτό των αντίστοιχων ιδρυμάτων του εσωτερικού. Ενδιαφέρον παρουσιάζει η πέμπτη εργαστηριακή άσκηση. Η μηχανή πεπερασμένων καταστάσεων, που ζητείται, υλοποιεί το κύκλωμα ελέγχου των φώτων κατεύθυνσης και κινδύνου του αυτοκινήτου Mercury Cougar του '68. Τα φώτα του συγκεκριμένου αυτοκινήτου λειτουργούσαν όπως φαίνεται παρακάτω.

Δεξιά Στροφή	Αριστερή Στροφή	Alarm
000 000	000 000	000 000
000 X00	00X 000	00X X00
000 XX0	0XX 000	0XX XX0
000 XXX	XXX 000	XXX XXX
000 000	000 000	000 000
κ.λ.π.	κ.λ.π.	κ.λ.π.

Το 0 δείχνει φώτα που είναι σβηστά και το X δείχνει φώτα αναμμένα. Το κάθε set από φώτα αποτελείται από τρεις λάμπες, ενώ κάθε γραμμή δείχνει έναν κύκλο του ρολογιού.

1.3.1 Πλατφόρμα Υλικού

Το Εργαστήριο Μικροεπεξεργαστών και Υλικού χρησιμοποιεί στις εργαστηριακές ασκήσεις της Λογικής Σχεδίασης breadboards, ολοκληρωμένα κυκλώματα πυλών, καθώς επίσης και την αναπτυξιακή πλακέτα που φαίνεται παρακάτω.



Σχήμα 1.12: Αναπτυξιακή πλακέτα εργαστηρίου E.M.Y.

Η πλακέτα διαθέτει:

- 16 LEDs.
- 2 Πιεστικούς Διακόπτες.

- 16 DIP Διακόπτες.
- 2 Ψηφία 7 στοιχείων(7 segment).

Τέλος, γίνεται χρήση του PLD GAL22V10.

1.3.2 Πλατφόρμα Λογισμικού

Στις εργαστηριακές ασκήσεις που γίνεται χρήση του PLD, οι φοιτητές συγγράφουν τον κώδικά τους σε γλώσσα περιγραφής υλικού και κάνουν χρήση του compiler PLDShell v5.0 της Altera, ο οποίος φαίνεται παρακάτω:



Σχήμα 1.13: Altera PLDShell v5.0 compiler in DOS

Κεφάλαιο 2

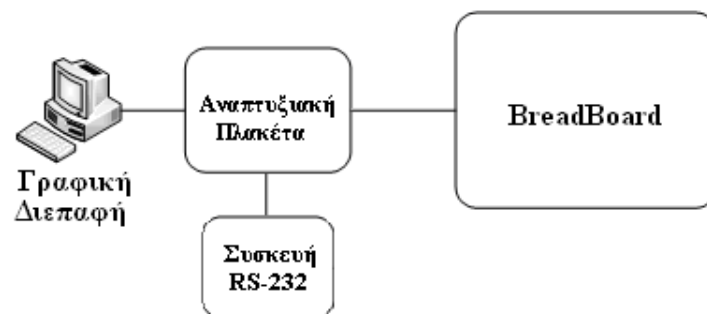
Η νέα πλατφόρμα

Η νέα πλατφόρμα που σχεδιάσαμε στα πλαίσια αυτής της διπλωματικής αποτελεί μια αρκετά διαφορετική προσέγγιση συγκριτικά με τον υλικοτεχνικό εξοπλισμό που χρησιμοποιείται στο Πολυτεχνείο Κρήτης μέχρι σήμερα, αλλά και στα υπόλοιπα ιδρύματα του εσωτερικού και εξωτερικού, όπως ήδη παρουσιάσαμε στο προηγούμενο κεφάλαιο.

Σκοπός μας είναι να παρέχουμε στο μαθητευόμενο μηχανικό πληθώρα πόρων για την εκτέλεση των εργαστηριακών πειραμάτων. Με τον όρο "πόρους" αναφερόμαστε στην ποικιλία εισόδων και εξόδων, αλλά και στον αριθμό αυτών. Επίσης, έμφαση δόθηκε στην φορητότητα της πλατφόρμας. Επιδιώξαμε να εξασφαλίσουμε για το μαθητευόμενο μηχανικό τη δυνατότητα πειραματισμού εκτός των ορίων του εργαστηρίου. Τέλος, έγινε πρόβλεψη επικοινωνίας μεταξύ του H/Y και σειριακών συσκευών, όπως π.χ. FPGAs.

Η πλατφόρμα αποτελείται από δύο μέρη: μια συσκευή που ονομάζουμε "**MHL Development Board**" (υλικό) και μια γραφική διεπαφή (λογισμικό) που ονομάζουμε "**MHL Development Board Simulator**". Από τούδε και στο εξής, θα αναφερόμαστε στην πρώτη ως *Αναπτυξιακή Πλακέτα* ή *Φυσική Διεπαφή* και στη δεύτερη ως *Γραφική* ή *Λειτουργική Διεπαφή*.

Στο παρακάτω σχηματικό διάγραμμα φαίνονται οι δομικές μονάδες της πλατφόρμας και η διασύνδεσή τους.



Σχήμα 2.1: Οι δομικές μονάδες της νέας πλατφόρμας

Όπως μπορούμε να δούμε, ο χρήστης δύναται να υλοποιήσει στο breadboard οποιοδήποτε κύκλωμα επιθυμεί και να συνδέσει τις απαραίτητες, για το κύκλωμά του, εισόδους και εξόδους που του παρέχει η αναπτυξιακή πλακέτα. Έπειτα, μπορεί να τροποποιήσει τις εξόδους από το γραφικό περιβάλλον στον Η/Υ και να παρατηρήσει τα αποτελέσματα του κυκλώματος, πάλι, στην οθόνη του υπολογιστή.

Σε αυτό το σημείο, χρήσιμο είναι να δώσουμε τον ορισμό των Εισόδων και των Εξόδων. Ως **Είσοδο** χαρακτηρίζουμε οποιοδήποτε σήμα εισέρχεται στην αναπτυξιακή πλακέτα και κατ'επέκταση στη γραφική διεπαφή ως έξοδος του κυκλώματος στο breadboard. Αντίστοιχα, **Έξοδοι** θεωρούνται τα σήματα που παράγονται από τη γραφική διεπαφή ή την πλακέτα και μέσω της τελευταίας καταλήγουν στο κύκλωμα του χρήστη ως εισοδοί. Όπως γίνεται αντιληπτό, η ορολογία που χρησιμοποιούμε έχει ως σημείο αναφοράς την φυσική ή τη γραφική διεπαφή. Μια έξοδος της αναπτυξιακής πλακέτας αποτελεί είσοδο ενός κυκλώματος για τον χρήστη και αντίστροφα.

Στις ενότητες που ακολουθούν θα γίνει εκτενής ανάλυση της δομής και αρχιτεκτονικής των μονάδων της πλατφόρμας, δηλαδή της φυσικής και της λειτουργικής διεπαφής. Σημειώνουμε πως στις ακόλουθες ενότητες δεν γίνεται αναφορά σε τεχνικές λεπτομέρειες (ο αναγνώστης μπορεί να ανατρέξει στο Κεφάλαιο 3 και 4 για περαιτέρω τεχνικές λεπτομέρειες).

2.1 Η Φυσική Διεπαφή με το Χρήστη στο Breadboard

Η Φυσική Διεπαφή αποτελεί το υλικό μέρος της πλατφόρμας. Σχεδιάζοντας τις προδιαγραφές της εργασίας καταλήξαμε σε κάποιους κεντρικούς άξονες. Η αναπτυξιακή πλακέτα, λοιπόν, θα έπρεπε:

- να διαθέτει μεγάλο αριθμό Εισόδων και Εξόδων.
- το μέγεθός της να μην είναι δεσμευτικό, ώστε η μεταφορά της να είναι εύκολη.

- να συνοδεύεται από λογισμικό ως συνέπεια των δύο προηγούμενων.
- να είναι συμβατή με τη νέα γενιά υπολογιστών που δεν διαθέτουν σειριακή θύρα για την επικοινωνία με το λογισμικό.

Κινούμενοι σε αυτό το πλαίσιο, οδηγηθήκαμε στην τελική μας υλοποίηση. Στις παραγράφους 2.1.1 και 2.2.1 γίνεται ανάλυση της δομής και αρχιτεκτονικής της αναπτυξιακής πλακέτας και της γραφικής διεπαφής, αντίστοιχα.

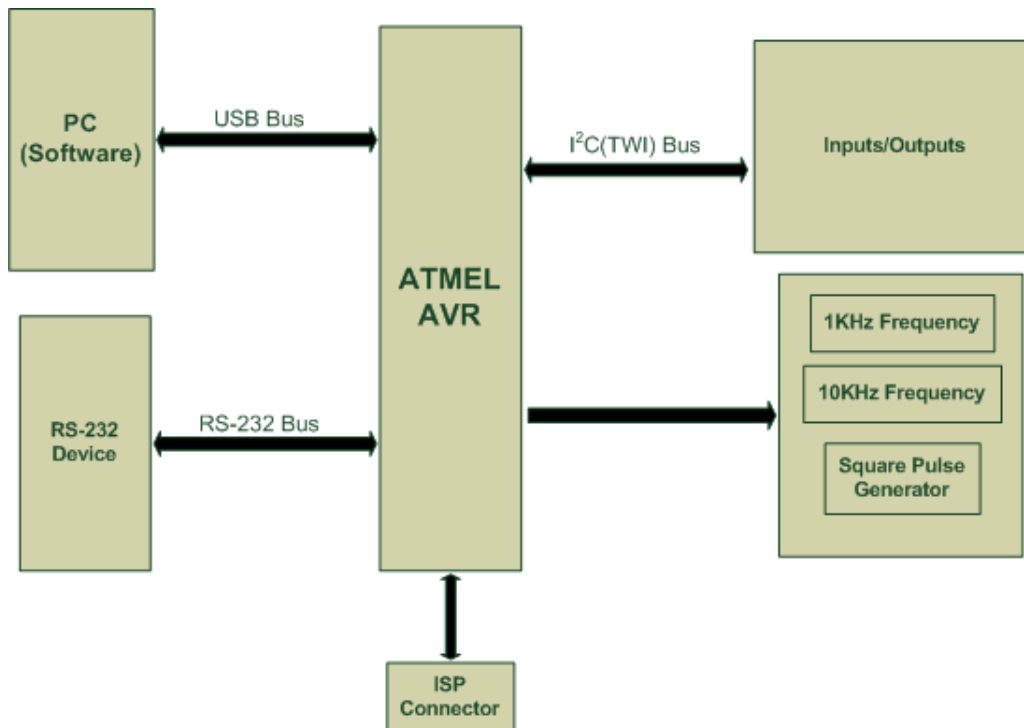
2.1.1 Δομή και Αρχιτεκτονική

Η φυσική διεπαφή υποστηρίζει μεγάλο και ποικίλο αριθμό εισόδων και εξόδων. Συνολικά, στην παρούσα έκδοση διαθέτει 132 ακροδέκτες εισόδων και εξόδων, 2 ακροδέκτες συχνοτήτων 1KHz και 10KHz, 1 ακροδέκτη τετραγωνικών παλμών, 4 ακροδέκτες για σύνδεση σε σειριακή θύρα, 3 ακροδέκτες τάσης 5 Volt και 3 ακροδέκτες γείωσης. Αναλυτικότερα, στη διάθεση του χρήστη βρίσκονται:

- 80 Είσοδοι εκ των οποίων:
 - 32 Φωτοδίοδοι(LEDs), στις οποίες αντιστοιχούν 32 ακροδέκτες.
 - 3 Οθόνες 7 στοιχείων(Seven Segments Displays) τεσσάρων ψηφίων έκαστη. Σε κάθε ψηφίο αντιστοιχούν 4 ακροδέκτες, οπότε συνολικά 3 οθόνες ·4 στοιχεία ·4 ακροδέκτες = 48 ακροδέκτες.
- 52 Έξοδοι εκ των οποίων:
 - 32 Διακόπτες S.P.D.T.(Single Pole, Double Throw), στους οποίους αντιστοιχούν 32 ακροδέκτες.
 - ένα Δεκαεξαδικό πληκτρολόγιο, στο οποίο αντιστοιχούν 4 ακροδέκτες.
 - ένας Μετρητής 8-bits, στο οποίο αντιστοιχούν 8 ακροδέκτες.
 - μία Γεννήτρια Ψευδοτυχαίων Αριθμών 8-bits (Galois Linear Feedback Shift Register), στο οποίο αντιστοιχούν 8 ακροδέκτες.
- 2 ακροδέκτες για τις γεννήτριες συχνοτήτων, μία του 1Hz και μία των 10KHz.
- 1 ακροδέκτης για την γεννήτρια τετραγωνικών παλμών της τάξης των 5KHz, η οποία πυροδοτείται με το πάτημα ενός πιεστικού διακόπτη.
- 4 ακροδέκτες για σύνδεση σε σειριακή θύρα.
- 3 ακροδέκτες τάσης 5 Volt.

- 3 ακροδέκτες γείωσης.

Όπως φαίνεται στο παρακάτω διάγραμμα, η φυσική διεπαφή στηρίζεται σε έναν μικροελεγκτή AVR. Η αρχιτεκτονική μας θέλει τον μικροελεγκτή υπεύθυνο για τη USB διασύνδεση με τον Η/Υ, την επικοινωνία με την σειριακή θύρα, την αποστολή και λήψη των εισόδων και εξόδων μέσω του διαύλου I^2C αλλά και για τη λειτουργία των γεννητριών συχνότητας.



Σχήμα 2.2: Το σχηματικό διάγραμμα του AVR

Πιο αναλυτικά, ο μικροελεγκτής αναλαμβάνει τη διαχείριση των εντολών που φτάνουν μέσω του διαύλου USB. Οι εντολές μπορεί να είναι δύο: αποστολή των εξόδων στο κύκλωμα του breadboard και αποστολή των δεδομένων εισόδου από το κύκλωμα προς τη γραφική διεπαφή. Τα δεδομένα εισόδων και εξόδων διατηρούνται στη μνήμη του μικροελεγκτή και ανανεώνονται με δειγματοληψία σε τακτά χρονικά διαστήματα. Οι συχνότητες των γεννητριών παράγονται από τον μικροελεγκτή με την χρήση εσωτερικής τεχνολογίας(χρήση μετρητών). Τέλος, πρόβλεψη έχει γίνει για την επικοινωνία του μικροελεγκτή διαμέσου της σειριακής RS-232 θύρας με συσκευές, όπου, κατά απαίτηση της λειτουργικής διεπαφής, ο χρήστης θα μπορεί να αποστείλει και να λάβει δεδομένα ακολουθώντας τη διαδρομή " $H/Y \rightarrow USB \rightarrow AVR \rightarrow RS - 232 \rightarrow Συσκευή$ " και αντίστροφα. Η παρούσα λειτουργία δεν έχει υλοποιηθεί σε αυτή την έκδοση του λογισμικού του AVR.

2.1.2 Λογική Διάταξη

Στην ενότητα αυτή θα περιγράψουμε αναλυτικά την δομή και αρχιτεκτονική του διαύλου USB, του διαύλου I²C(TWI), του λογισμικού(firmware) του μικροελεγκτή AVR και των υπολοίπων υποσυστημάτων της αναπτυξιακής πλακέτας.

Ο δίαυλος USB

Όπως περιγράφεται και στις προδιαγραφές μας, θέλαμε η υλοποίησή μας να είναι συμβατή με τη νέα γενιά υπολογιστών που δεν διαθέτουν σειριακή θύρα για την επικοινωνία με τη Γραφική Διεπαφή. Για την διασύνδεσή της, λοιπόν, διαλέξαμε τον Ενιαίο Σειριακό Δίαυλο(USB).

Ο Ενιαίος Σειριακός Δίαυλος, γνωστός και σαν Universal Serial Bus ή απλά USB, είναι ένα σύστημα διαύλου, το οποίο χρησιμοποιείται για την επικοινωνία ενός υπολογιστή με περιφερειακά συστήματα. Ο USB είχε ως σκοπό να επιτρέψει στις περιφερειακές μονάδες να συνδέονται με τον υπολογιστή χρησιμοποιώντας μια ενιαία τυποποιημένη υποδοχή διεπαφών και να βελτιώσει τις έτοιμες προς χρήση ικανότητες των συσκευών για σύνδεση ή αποσύνδεσή τους με το σύστημα χωρίς να χρειάζεται επανεκκίνηση. Άλλα χαρακτηριστικά γνωρίσματα περιλαμβάνουν την παροχή ηλεκτρικού ρεύματος στις συσκευές χαμηλής κατανάλωσης, χωρίς την ανάγκη εξωτερικής παροχής ηλεκτρικού ρεύματος και την δυνατότητα χρήσης πολλών συσκευών USB, δίχως την απαίτηση ρυθμίσεων ή μεμονωμένων προγραμμάτων οδήγησης (drivers) από τους κατασκευαστές. Ο USB λειτουργεί στις ακόλουθες ταχύτητες:

- USB 1.0 : 1.5 Mbit/s (χαμηλή-low ταχύτητα),
- USB 1.1 : 12 Mbit/s (πλήρης-full ταχύτητα) ή
- USB 2.0 : 480 Mbit/s (υψηλή-high ταχύτητα).

Πρόσφατα, παρουσιάστηκε και η νέα έκδοση του διαύλου, USB 3.0, η οποία πετυχαίνει ταχύτητες διαμεταγωγής έως και 5Mbit/s.

Για να υλοποιήσουμε τη διεπαφή με τον δίαυλο στην κατασκευή μας, διαθέταμε τις εξής επιλογές:

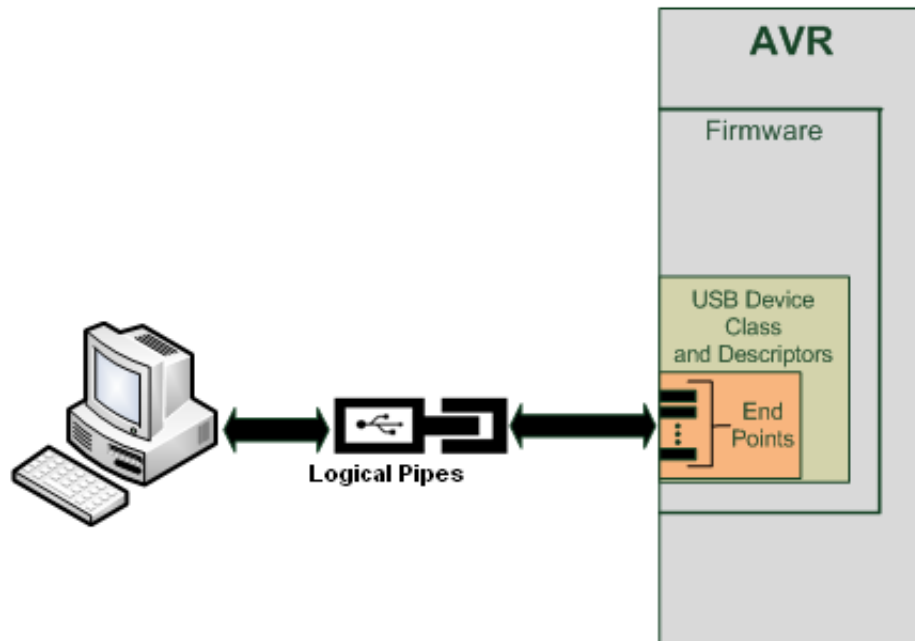
1. χρήση μικροελεγκτή με ενσωματωμένο USB ελεγκτή(USB controller).
2. χρήση μετατροπέα USB(USB to Serial converter, USB to I²C converter, κ.λ.π.).
3. χρήση απλού μικροελεγκτή, υλοποιώντας τον ελεγκτή USB σε επίπεδο λογισμικού ως κομμάτι του firmware του.

Η λύση που διαλέξαμε ήταν η τελευταία. Η επιλογή μας αυτή κράτησε χαμηλά το κόστος κατασκευής της αναπτυξιακής πλακέτας, παρείχε μια αρκετά αξιόπιστη υλοποίηση του διαύλου και δεν απαιτούσε την εύρεση και χρήση ιδιαίτερων υλικών κατασκευής, που θα δυσχέραινε την φάση της συναρμολόγησης της πλακέτας. Παράλληλα, μπορέσαμε να έρθουμε σε επαφή και να πειραματιστούμε με μια διαφορετική προσέγγιση της υλοποίησης και χρήσης του διαύλου USB, δίνοντας μας τη δυνατότητα να σχεδιάσουμε και να υλοποιήσουμε τις δικές μας δομές περιγραφής και κλάσης(USB Descriptors, USC Device Class) της συσκευής.

Η σχεδίασή μας στηρίχτηκε στην υλοποίηση V-USB(Virtual USB) του Christian Starkjohann, η οποία αποτελεί μια υλοποίηση USB οδηγού βασισμένη σε "λογισμικό μόνο"(software-only implementation)[5]. Η υλοποίηση αυτή ακολουθεί όλες τις προδιαγραφές της έκδοσης 1.1 του διαύλου, προσφέροντας τα εξής χαρακτηριστικά:

- Υλοποίηση χαμηλής ταχύτητας(Low-speed) συσκευής, πλήρως συμβατή με την έκδοση 1.1 του USB, παρέχοντας εξαιρετική διαχείριση σφαλμάτων επικοινωνίας και ηλεκτρικών προδιαγραφών.
- Υποστήριξη πολλαπλών endpoints: 1 control endpoint, 2 interrupt/bulk-in endpoints και μέχρι 7 interrupt/bulk-out endpoints.
- Υποστήριξη μετάδοσης πακέτων δεδομένων μέχρι και 254 bytes ανά μετάδοση και δυνατότητα διαχείρισης πακέτων μεγαλύτερου μεγέθους.
- Δωρεάν USB identifiers (Vendor-ID and Product-ID pairs).
- Συμβατό με κάθε μικροελεγκτή AVR ο οποίος διαθέτει περισσότερα των 2kB Flash memory, 128 Bytes RAM και ρολόι τουλάχιστον 12 MHz.
- Δεν είναι απαραίτητη η χρήση εξειδικευμένου hardware, όπως UART, timers, input capture unit, εκτός ενός ακμοπυροδότητου interrupt.
- Δυνατότητα χρονισμού του μικροελεγκτή σε συχνότητα 12Mhz, 15MHz, 16MHz, ή 20 MHz με χρήση ταλαντωτή κρυστάλλου ή 12.8 MHz και 16.5 MHz με χρήση του εσωτερικού RC ταλαντωτή.
- Μέγεθος πηγαίου κώδικα 1150 έως 1400 Bytes, γραμμένος σε γλώσσα C.
- Δυνατότητα χρήσης οδηγών(drivers) συμβατών με WindowsTM, Linux και Mac OSTM.
- Ελεύθερη ή εμπορική άδεια χρήσης.

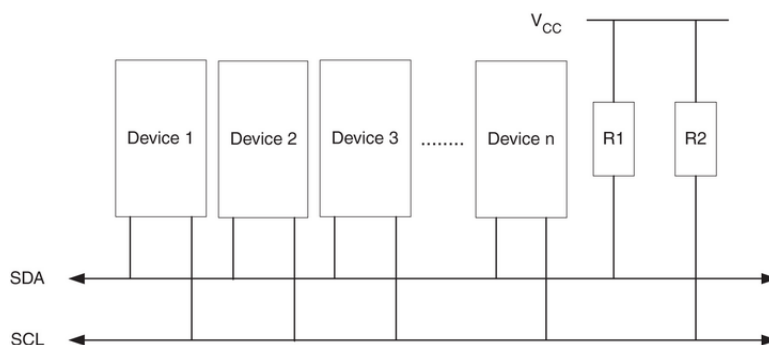
Έτσι, με βάση την παραπάνω υλοποίηση, δομήσαμε το firmware μας ορίζοντας την δική μας κλάση της συσκευής(USB Device Class) και τους δικούς μας Descriptors. Το παρακάτω διάγραμμα παρουσιάζει τη δομή της υλοποίησής μας.



Σχήμα 2.3: Η υλοποίηση USB στον AVR

Ο διάυλος I²C

Το πρωτόκολλο I²C(Inter-Integrated Circuit)[25] είναι ένα σειριακό πρωτόκολλο που αναπτύχθηκε από την PhilipsTM και αργότερα υιοθετήθηκε από πολλές εταιρείες του κλάδου των ηλεκτρονικών. Το σημαντικότερο χαρακτηριστικό του είναι ότι επιτρέπει τη σύνδεση μέχρι και 128 συσκευών χρησιμοποιώντας μόνο δύο αμφίδρομα καλώδια, ένα για ρολόι(SCL) και ένα για δεδομένα(SDA), γεγονός που το καθιστά ιδανικό σε εφαρμογές που εμπλέκονται μικροελεγκτές. Ακόμα, όλες οι συσκευές που συνδέονται στο δίαυλο διαθέτουν μοναδική διεύθυνση και μηχανισμούς για την προτεραιότητα της κάθε συσκευής όσον αφορά τη χρήση του διαύλου. Το παρακάτω σχήμα περιγράφει τη συνδεσμολογία των συσκευών:



Σχήμα 2.4: Οι συσκευές στο δίαυλο I^2C

Όπως φαίνεται παραπάνω, οι δύο γραμμές του διαύλου συνδέονται μέσω pull-up αντιστάσεων στη θετική παροχή τάσης. Αυτό είναι απαραίτητο, διότι οι γραμμές είναι open-drain ή open-collector σε όλες τις συμβατές με το πρωτόκολλο συσκευές.

Οι συσκευές που διαλέξαμε να χρησιμοποιήσουμε είναι τέσσερα ολοκληρωμένα κυκλώματα τα οποία διαθέτουν την τεχνολογία I^2C και παρέχουν 40 ακροδέκτες προγραμματιζόμενων Εισόδων/Εξόδων Γενικής Χρήσης (I^2C to GPIO).

Όπως αναφέραμε, κάθε συσκευή που συνδέεται στο δίαυλο διαθέτει μοναδική διεύθυνση, ώστε, σύμφωνα με την αρχιτεκτονική "Αφέντη - Υπηρετών" (*Master - Slaves*), ο "Αφέντης", που στην περίπτωση μας είναι ο μικροελεγκτής, να "διατάζει" τους "Υπηρέτες", δηλαδή τα ολοκληρωμένα Εισόδων/Εξόδων γενικής χρήσης, να διανέμουν κατάλληλα τα δεδομένα εισόδων/εξόδων. Η χρήση τεσσάρων ολοκληρωμένων I^2C to GPIO μας διαθέτει 160 ακροδέκτες, εκ των οποίων στην παρούσα έκδοση του firmware, χρησιμοποιούνται οι 132. Τα υπόλοιπα δύνανται να χρησιμοποιηθούν μελλοντικά. Σημειώνουμε πως η υλοποίησή μας μπορεί να δεχτεί έως και πέντε ολοκληρωμένα I^2C to GPIO, δηλαδή μπορεί να υποστηρίξει έως 200 Εισόδους και Εξόδους. Οι διευθύνσεις των ολοκληρωμένων είναι διαδοχικές με αρχική την "000" και τελική την "100". Ακόμα, πρέπει να αναφέρουμε πως τα πρώτα δύο σε σειρά ολοκληρωμένα I^2C to GPIO είναι υπεύθυνα για τις Εισόδους της αναπτυξιακής πλακέτας, ενώ τα δύο επόμενα για τις Εξόδους.

Γεννήτριες Συχνοτήτων

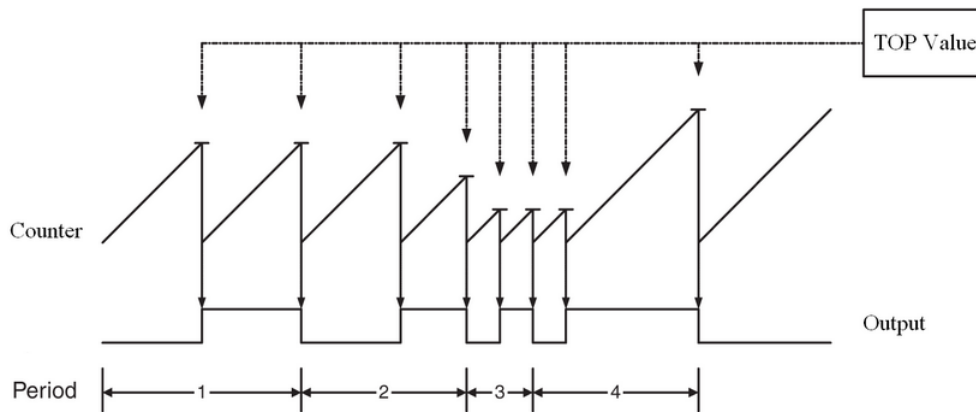
Για την υλοποίηση των Γεννητριών συχνοτήτων χρησιμοποιήσαμε μια αρκετά έξυπνη τεχνολογία που διαθέτει ο μικροελεγκτής AVR. Η τεχνολογία αυτή ονομάζεται *Clear Timer on Compare Match (CTC)*. Η λειτουργία της τεχνολογίας βασίζεται σε έναν απλό μετρητή και μια "Ανώτερη Τιμή" (TOP Value). Ο μετρητής αρχίζει από το 0 και αυξάνει σε κάθε κύκλο ρολογιού διαδοχικά μέχρι τη μέγιστη τιμή του. Θέτοντας την Ανώτερη Τιμή αναγκάζουμε τον μετρητή να συγκρίνει σε κάθε κύκλο την τιμή του με την Ανώτερη τιμή και όταν αυτές ταυτίζονται, να

μηδενίσει και να αρχίσει να μετράει από την αρχή. Κάθε φορά που η σύγκριση των δύο τιμών είναι αληθής ένα δευτερεύον σήμα μεταβάλλει το δυναμικό του από χαμηλό σε υψηλό και αντίστροφα, με σκοπό να ενημερώσει τον επεξεργαστή ότι ο μετρητής έφτασε την Ανώτερη τιμή. Αυτό το σήμα αποτελεί το ζητούμενο σήμα εξόδου. Μεταβάλλοντας κατάλληλα την Ανώτερη τιμή και τον διαιρέτη της συχνότητας του επεξεργαστή, σύμφωνα με τη σχέση

$$f = \frac{f_{CPU}}{2 \times N \times (1 + TOPValue)}$$

όπου N ο διαιρέτης της συχνότητας του επεξεργαστή (1, 8, 64, 256, 1024) και $TOPValue$ η Ανώτερη τιμή.

Το παρακάτω διάγραμμα παρουσιάζει τη λειτουργία CTC.

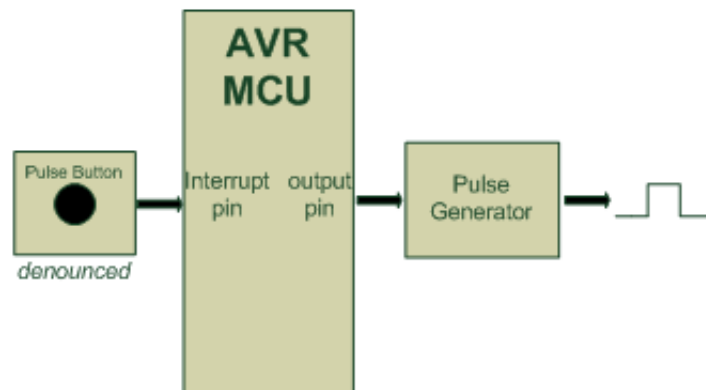


Σχήμα 2.5: Διάγραμμα χρονισμού της τεχνολογίας CTC

Γεννήτρια Τετραγωνικών Ωρολογιακών Παλμών

Από την αναπτυξιακή πλακέτα μας δεν θα μπορούσε να λείπει μια Γεννήτρια Τετραγωνικών Ωρολογιακών Παλμών. Θεωρήσαμε χρήσιμο να έχουμε τη δυνατότητα σε σύγχρονα κυκλώματα που θα σχεδιάζονται, όπως για παράδειγμα κυκλώματα με flip-flops, να μεταβάλλουμε την κατάσταση του κυκλώματος χρησιμοποιώντας απλούς ωρολογιακούς παλμούς και όχι τις γεννήτριες συχνοτήτων, ώστε να παρακολουθούμε τις εξόδους του κυκλώματος σε κάθε αλλαγή κατάστασης των σύγχρονων στοιχείων του.

Η Γεννήτρια Τετραγωνικών Ωρολογιακών Παλμών υλοποιήθηκε με αρκετά απλό τρόπο. Ο παλμός δημιουργείται κάθε φορά που ένας πιεστικός διακόπτης κλείνει (Pulse Button). Ο πιεστικός διακόπτης συνδέθηκε σε έναν ακροδέκτη του μικροελεγκτή, ο οποίος προκαλεί κάποιο "Interrupt" στον επεξεργαστή αναγκάζοντας τον να καλέσει μια ρουτίνα, η οποία με τη σειρά της οδηγεί έναν άλλο ακροδέκτη σε υψηλό δυναμικό για συγκεκριμένο χρόνο. Το παρακάτω σχήμα περιγράφει τη συνδεσμολογία:

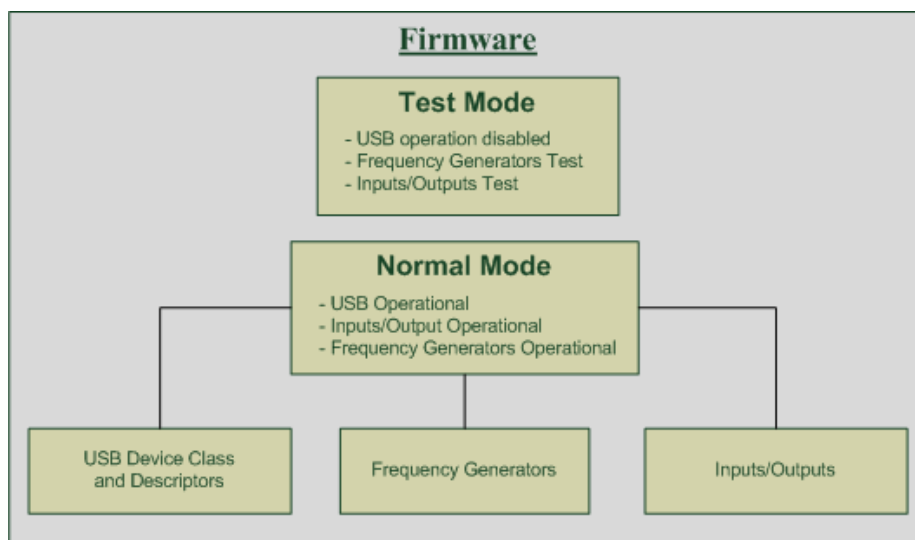


Σχήμα 2.6: Σχηματικό διάγραμμα της γεννήτριας παλμών

Σημειώνουμε πως ο πιεστικός διακόπτης παλμών είναι απαλλαγμένος από αναπηδήσεις (debounced button).

Το λογισμικό(firmware) του μικροελεγκτή AVR

Το λογισμικό(firmware) του μικροελεγκτή διαθέτει δύο καταστάσεις λειτουργίας: την δοκιμαστική λειτουργία και την κανονική.



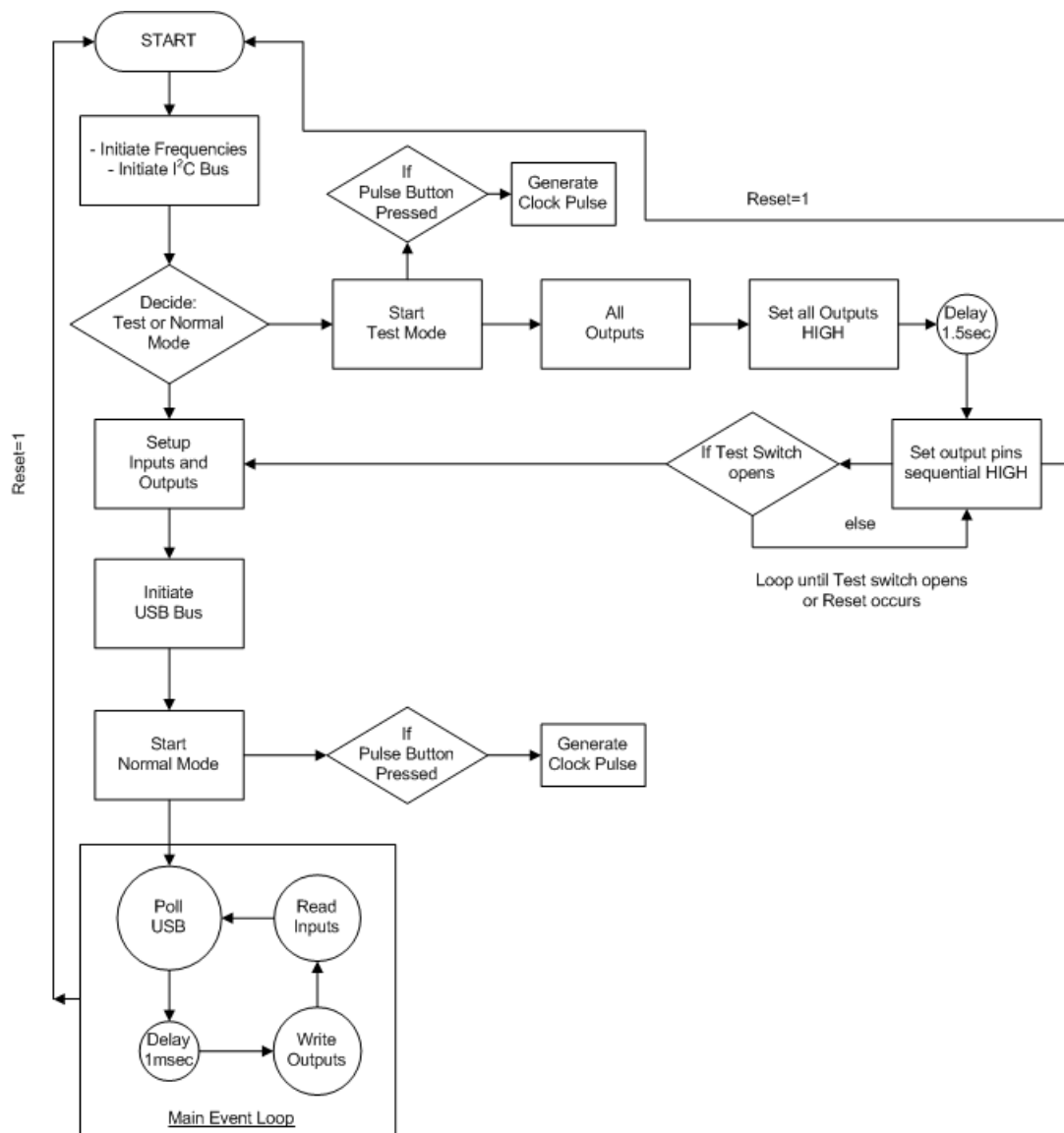
Σχήμα 2.7: Το λογισμικό(firmware) του AVR

Η δοκιμαστική λειτουργία στοχεύει στην εξακρίβωση της ακεραιότητας του κυκλώματος της αναπτυξιακής πλακέτας και στον έλεγχο μίας προς μία των εισόδων και εξόδων. Ενεργοποιείται στην περίπτωση που ένα εξωτερικός διακόπτης είναι κλειστός, τότε τροφοδοτώντας την πλακέτα με ρεύμα η δοκιμαστική λειτουργία ξεκινάει. Στη λειτουργία αυτή δεν είναι διαθέσιμη η διασύνδεση USB, όλοι οι

ακροδέκτες(ακόμα και αυτοί των εισόδων) γίνονται έξοδοι και παίρνουν διάφορες τιμές για να επιβεβαιώσουμε την ορθή λειτουργία τους και, τέλος, οι γεννήτριες συχνοτήτων και η γεννήτρια παλμών λειτουργούν κανονικά ώστε να μπορούν να ελεγχθούν και αυτές με τη σειρά τους. Για τον έλεγχο του κυκλώματος μπορούν να χρησιμοποιηθούν εργαλεία, όπως πολύμετρο και παλμογράφος. Ωστόσο, για να γίνει πιο εύκολη η επιβεβαίωση της ορθής λειτουργίας του κυκλώματος λόγω των πολλών ακροδεκτών, σχεδιάσαμε μια δοκιμαστική πλακέτα γι' αυτόν ειδικά το σκοπό, η οποία μας βοήθησε καθ' όλη τη διάρκεια της υλοποίησής μας στην αποσφαλμάτωση του firmware. Η δοκιμαστική πλακέτα θα αναλυθεί παρακάτω. Σημειώνουμε ότι αν ο διακόπτης ενεργοποίησης της δοκιμαστικής λειτουργίας κλείσει κατά την διάρκεια της κανονικής λειτουργίας, αρκεί ένα πάτημα του διακόπτη επανεκκίνησης (reset) για να ξεκινήσει η δοκιμαστική λειτουργία.

Κατά την κανονική λειτουργία όλες οι επιμέρους δυνατότητες της αναπτυξιακής πλακέτας είναι διαθέσιμες. Για να ξεκινήσει η κανονική λειτουργία αρκεί να τροφοδοτήσουμε την πλακέτα με ρεύμα και να είναι ανοικτός ο διακόπτης δοκιμαστικής λειτουργίας.

Το παρακάτω διάγραμμα ροής εξηγεί τη λειτουργία του μικροελεγκτή, όταν τροφοδοτείται με ρεύμα.



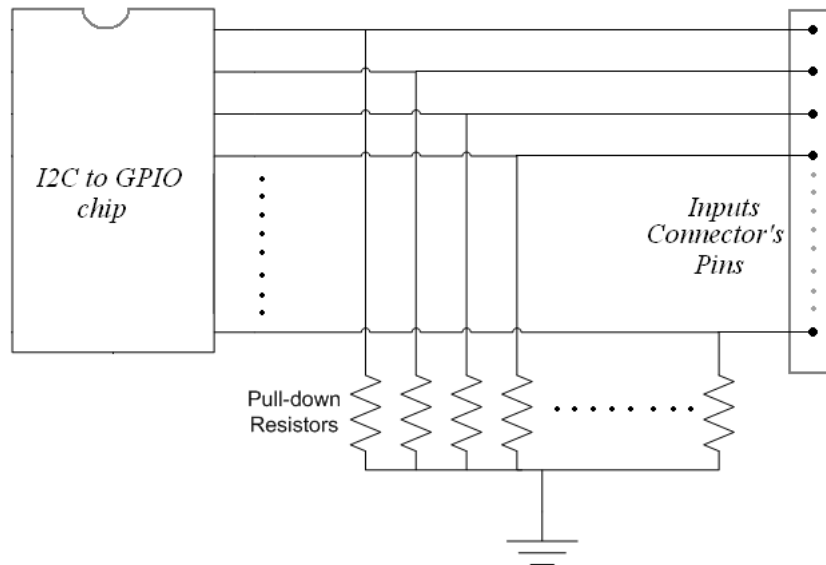
Σχήμα 2.8: Διάγραμμα ροής του firmware

Οι ακροδέκτες Εισόδων και Εξόδων

Επειδή τα κυκλώματα των εργαστηριακών ασκήσεων λειτουργούν και δοκιμάζονται σε πραγματικό χρόνο, η αναπτυξιακή πλακέτα μας σχεδιάστηκε έτσι, ώστε οι είσοδοι και οι έξοδοι να μεταβάλλονται σε πραγματικό-εκτελέσιμο χρόνο σύμφωνα με την τεχνολογία Live Insertion, γνωστή και ως Hot Swap[24].

Οι ακροδέκτες Εισόδων και Εξόδων συνδέονται στους ακροδέκτες των ολοκληρωμένων I²C to GPIO ένας προς έναν. Στους ακροδέκτες Εισόδων έχουμε

τοποθετήσει pull-down αντιστάσεις, ώστε, όταν κάποιος ακροδέκτης είναι αποσυνδεδεμένος, το δυναμικό του να παραμένει χαμηλό και η τιμή της εισόδου να είναι το λογικό "0". Όταν κάποια είσοδος συνδεθεί στον ακροδέκτη με υψηλό δυναμικό, τότε η τιμή του ακροδέκτη εισόδου γίνεται το λογικό "1". Το παρακάτω σχήμα περιγράφει τη συνδεσμολογία.

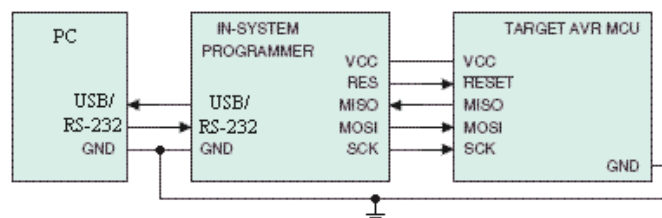


Σχήμα 2.9: Οι συνδέσεις των ακροδεκτών Εισόδων/Εξόδων

Σημειώνουμε ότι η κατασκευή μας υποστηρίζει τοποθέτηση pull-down αντιστάσεων και στους 200 ακροδέκτες Εισόδων/Εξόδων. Οι αντιστάσεις πρέπει να τοποθετούνται **μόνο** στους ακροδέκτες που προορίζονται για χρήση εισόδων στην αναπτυξιακή πλακέτα.

Προγραμματισμός Μικροελεγκτή

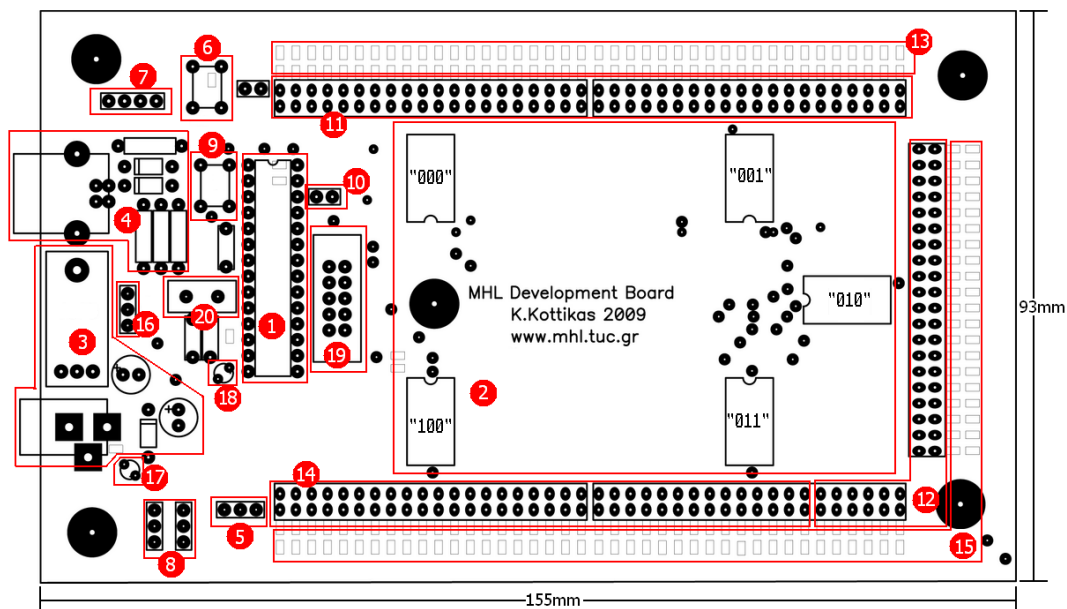
Για να είναι εύκολη η διαδικασία προγραμματισμού του μικροελεγκτή φροντίσαμε να συμπεριλάβουμε στη σχεδιάσή μας μια θύρα προγραμματισμού 10 ακροδεκτών σύμφωνα με της προδιαγραφές της Atmel για την τεχνολογία In-System Programming[2]. Η συνδεσμολογία φαίνεται στο παρακάτω σχήμα.



Σχήμα 2.10: Σχηματικό διάγραμμα ISP προγραμματισμού

2.1.3 Φυσική Διάταξη

Η φυσική διάταξη των δομικών μονάδων της αναπτυξιακής πλακέτας ακολουθεί μια αρκετά απλή αλλά αποδοτική φιλοσοφία. Όπως φαίνεται και στην εικόνα παρακάτω, οι ακροδέκτες εισόδων και εξόδων είναι τοποθετημένοι περιμετρικά της πλακέτας παρέχοντας άνεση στο χρήστη και συνάμα αποτρέπει την περαιτέρω επαφή με τα πιο ευαίσθητα στοιχεία του κυκλώματος. Η θύρα USB και η υποδοχή εξωτερικής τροφοδοσίας βρίσκονται στην ίδια πλευρά της πλακέτας έτσι ώστε τα καλώδια σύνδεσης να μην εμποδίζουν τη συνδεσμολογία των καλωδίων του κυκλώματος στο breadboard. Ο πιεστικός διακόπτης της γεννήτριας των παλμών, καθώς και οι ακροδέκτες των συχνοτήτων και της σειριακής θύρας, βρίσκονται και αυτοί στην περίμετρο παρέχοντας εύκολη πρόσβαση κατά την χρήση τους.



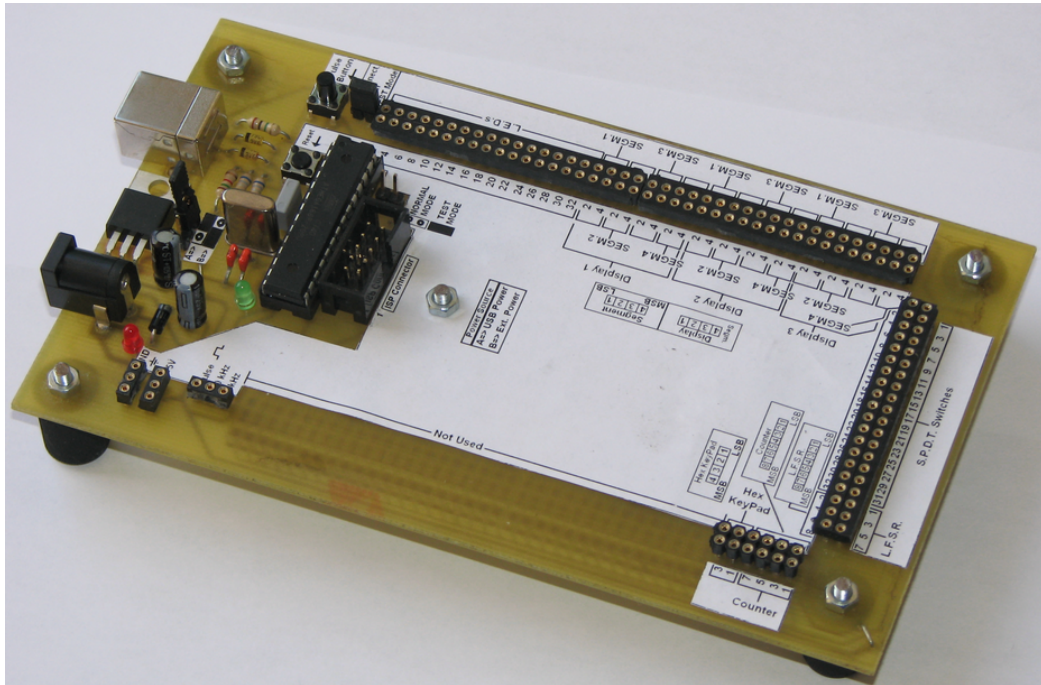
Σχήμα 2.11: Φυσική Διάταξη της αναπτυξιακής πλακέτας

Πιο αναλυτικά, στην παραπάνω εικόνα διακρίνουμε:

1. Τον μικροελεγκτή AVR.
2. Τα ολοκληρωμένα κυκλώματα I²C to GPIO.
3. Το κύκλωμα εξωτερικής τροφοδοσίας.
4. Την υποδοχή και το κύκλωμα της θύρας USB.

5. Τους ακροδέκτες συχνοτήτων και τετραγωνικού παλμού.
6. Τον πιεστικό διακόπτη της γεννήτριας παλμών.
7. Τους ακροδέκτες σύνδεσης της σειριακής(RS-232) συσκευής.
8. Τους ακροδέκτες τάσης +5 Volt και γείωσης.
9. Τον πιεστικό διακόπτη επανεκκίνησης(Reset).
10. Τον διακόπτη μετάβασης στη Δοκιμαστική Λειτουργία.
11. Τους ακροδέκτες Εισόδων.
12. Τους ακροδέκτες Εξόδων.
13. Τις pull-down αντιστάσεις των εισόδων.
14. Τους ακροδέκτες για την μελλοντική επέκταση των μονάδων της πλακέτας.
15. Τις θήκες pull-down αντιστάσεων για τις μελλοντικές εισόδους.
16. Τον διακόπτη επιλογής τροφοδοσίας (Εξωτερική ή USB).
17. Το ενδεικτικό LED λειτουργίας.
18. Το ενδεικτικό LED δραστηριότητας του διαύλου USB.
19. Την υποδοχή ISP για τον προγραμματισμό του AVR.
20. Τον κρυσταλλικό ταλαντωτή του AVR.

Στην φωτογραφία που ακολουθεί βλέπουμε την αναπτυξιακή πλακέτα στην τελική της μορφή:



Σχήμα 2.12: Η τελική έκδοση της αναπτυξιακής πλακέτας

Αναλυτικότερες τεχνικές λεπτομέρειες για την φυσική διεπαφή περιλαμβάνονται στο Κεφάλαιο 3.

2.2 Η Λειτουργική Διεπαφή με το Χρήστη στον Η/Υ

Το δεύτερο μέρος της αναπτυξιακής πλατφόρμας που σχεδιάσαμε είναι η Γραφική ή Λειτουργική Διεπαφή. Το μέρος αυτό είναι εξίσου σημαντικό με την φυσική διεπαφή, διότι αποτελεί το υψηλότερο επίπεδο όσον αφορά την διαδραστικότητα μεταξύ χρήστη και πειραματικού κυκλώματος μετά την υλοποίηση του τελευταίου. Στο επίπεδο αυτό ο χρήστης παρακολουθεί όλα τα σήματα εισόδων και μεταβάλλει κατά βούληση τα σήματα εξόδων.

Η Λειτουργική Διεπαφή αναπτύχθηκε με γνώμονα την φιλικότητα προς τον χρήστη. Όπως φανερώνει το όνομα της εφαρμογής(MHL Development Board Simulator), προσπαθήσαμε να μεταφέρουμε στην οθόνη του χρήστη μία αναπτυξιακή πλακέτα, δηλαδή να προσομοιώσουμε μια φυσική αναπτυξιακή πλακέτα με λογισμικό. Ακόμα, θελήσαμε το περιβάλλον εργασίας του χρήστη να είναι ευχάριστο και συνάμα απλό και κατανοητό. Οι μονάδες που εμφανίζονται στη γραφική διεπαφή δεν είναι περίπλοκες και δεν απαιτούν ιδιαίτερες ικανότητες ή γνώσεις στη χρήση τους.

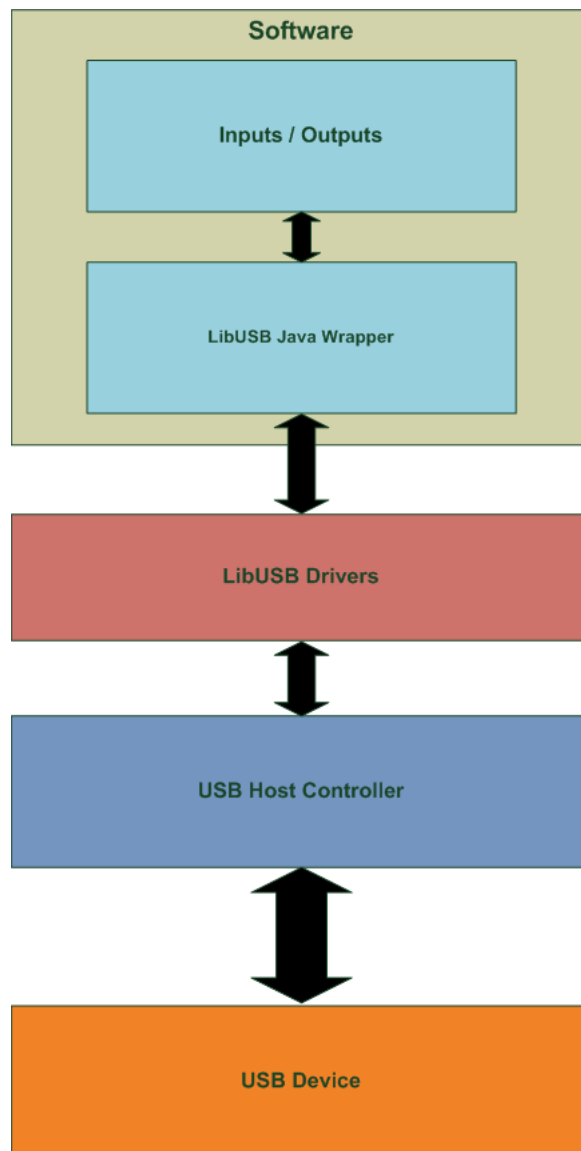
2.2.1 Δομή και Αρχιτεκτονική

Η αρχιτεκτονική του λογισμικού μας περιλαμβάνει δύο μέρη. Αρχικά θα περιγράψουμε τη δομή του λογισμικού μας σε συνεργασία με τα προγράμματα οδήγησης της συσκευής και στη συνέχεια την αρχιτεκτονική της γραφικής διεπαφής.

Η αναπτυξιακή πλακέτα επικοινωνεί μέσω του USB δρομολογητή του υπολογιστή(USB Controllor) χρησιμοποιώντας προγράμματα οδήγησης(Drivers) που περιγράφουν στο λειτουργικό σύστημα την USB συσκευή και τον τρόπο λειτουργίας της. Τα προγράμματα οδήγησης που χρησιμοποιήσαμε αποτελούν μια υλοποίηση προγραμμάτων οδήγησης σχεδιασμένα σύμφωνα με τις προδιαγραφές του διαύλου USB και είναι ικανά να χρησιμοποιηθούν σε διάφορες USB συσκευές. Το πακέτο των προγραμμάτων είναι λογισμικό ανοικτού κώδικα και ονομάζεται LibUSB[12]. Οι οδηγοί αυτοί έχουν αναπτυχθεί με τη χρήση της γλώσσας προγραμματισμού C και C++.

Στοχεύοντας στη δημιουργία λογισμικού ικανού να εκτελείται σε διάφορα λειτουργικά συστήματα(Cross-platform software), διαλέξαμε την γλώσσα προγραμματισμού JAVA για την δημιουργία και συγγραφή της λειτουργικής διεπαφής. Φυσική απόρροια της επιλογής μας ήταν η ασυμβατότητα μεταξύ γραφικής διεπαφής και χρήσης προγραμμάτων οδήγησης. Λύση στο πρόβλημά μας έδωσε η χρήση του "περιτυλίγματος" JAVA LibUSB Wrapper[15], ο οποίος αναλαμβάνει να δώσει στον σχεδιαστή την δυνατότητα χρήσης όλων των ρουτινών του πακέτου LibUSB στην πλατφόρμα σχεδίασης και συγγραφής της JAVA.

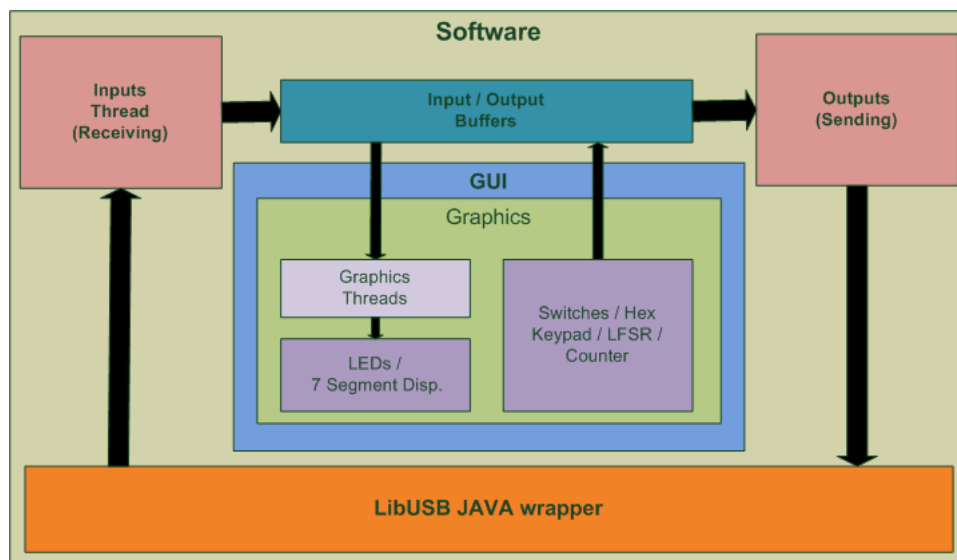
Το παρακάτω διάγραμμα εξηγεί τον τρόπο χρήσης του wrapper σε συνδυασμό με τους οδηγούς LibUSB.



Σχήμα 2.13: Σύνδεση του υπολογιστή με την αν. πλακέτα

Η φιλοσοφία της διεπαφής είναι αρκετά απλή, αλλά αποδοτική. Δύο καταχωρητές (Buffers) συγκρατούν τα δεδομένα εισόδων και εξόδων. Τα δεδομένα του καταχωρητή εισόδων (Inputs Buffer) ανανεώνονται από ένα νήμα (Main Thread), το οποίο εκτελείται επαναλαμβανόμενα μετά την επιτυχή σύνδεση της συσκευής και απενεργοποιείται κατά την αποσύνδεσή της. Το νήμα αυτό είναι υπεύθυνο για

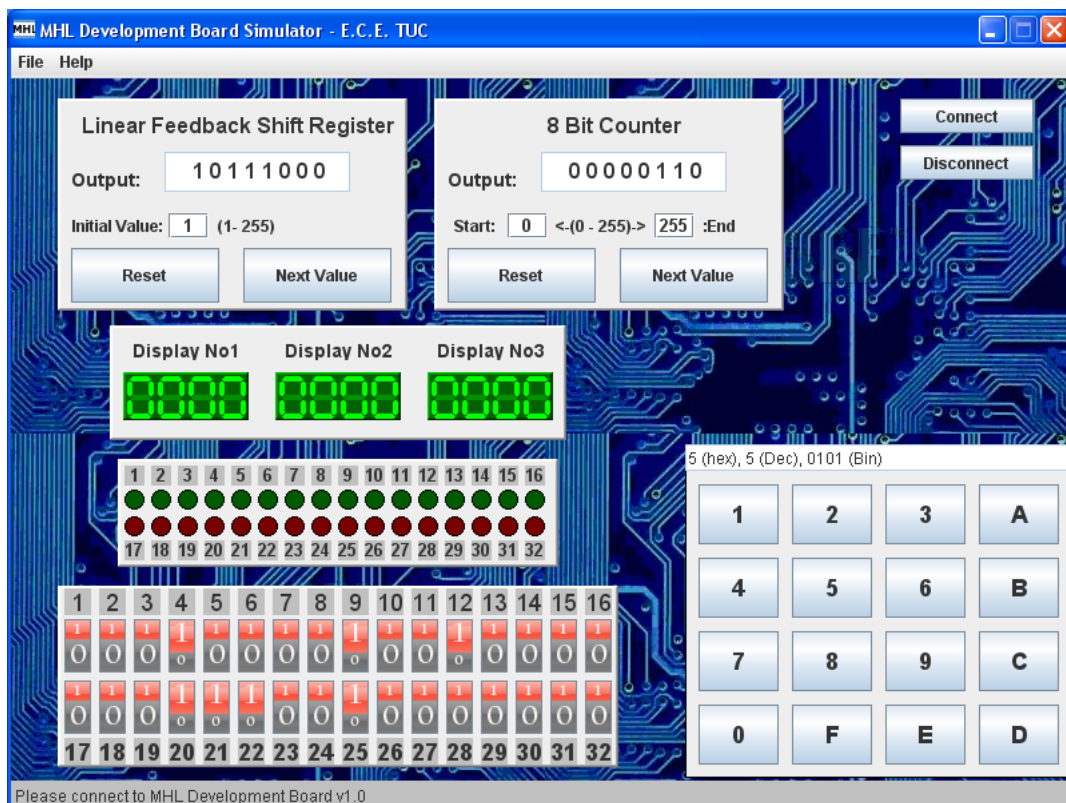
τη συνεχή ανάγνωση των δεδομένων εισόδου από την αναπτυξιακή πλακέτα στέλνοντας τις κατάλληλες εντολές προς το δίαυλο USB. Επιμέρους νήματα(threads) διαχειρίζονται τα γραφικά των εισόδων(LEDs, 7-Segment Displays), σύμφωνα με τα δεδομένα του αντίστοιχου καταχωρητή και ανανεώνουν τα γραφικά αμέσως μόλις πραγματοποιηθεί αλλαγή στις εισόδους της πλακέτας. Τα δεδομένα του καταχωρητή εξόδων ανανεώνονται μετά από οποιαδήποτε αλλαγή στην κατάσταση των αντίστοιχων δομικών μονάδων του λογισμικού(Διακόπτες, Δεκαεξαδικό Πληκτρολόγιο, LFSR, Counter) και αποστέλλονται προς την πλακέτα. Οποιαδήποτε αποστολή ή λήψη δεδομένων πραγματοποιείται διαμέσου του wrapper και κατ'επέκταση των οδηγών. Η αρχιτεκτονική του λογισμικού μας παρουσιάζεται σχηματικά παρακάτω:



Σχήμα 2.14: Αρχιτεκτονική της γραφικής διεπαφής

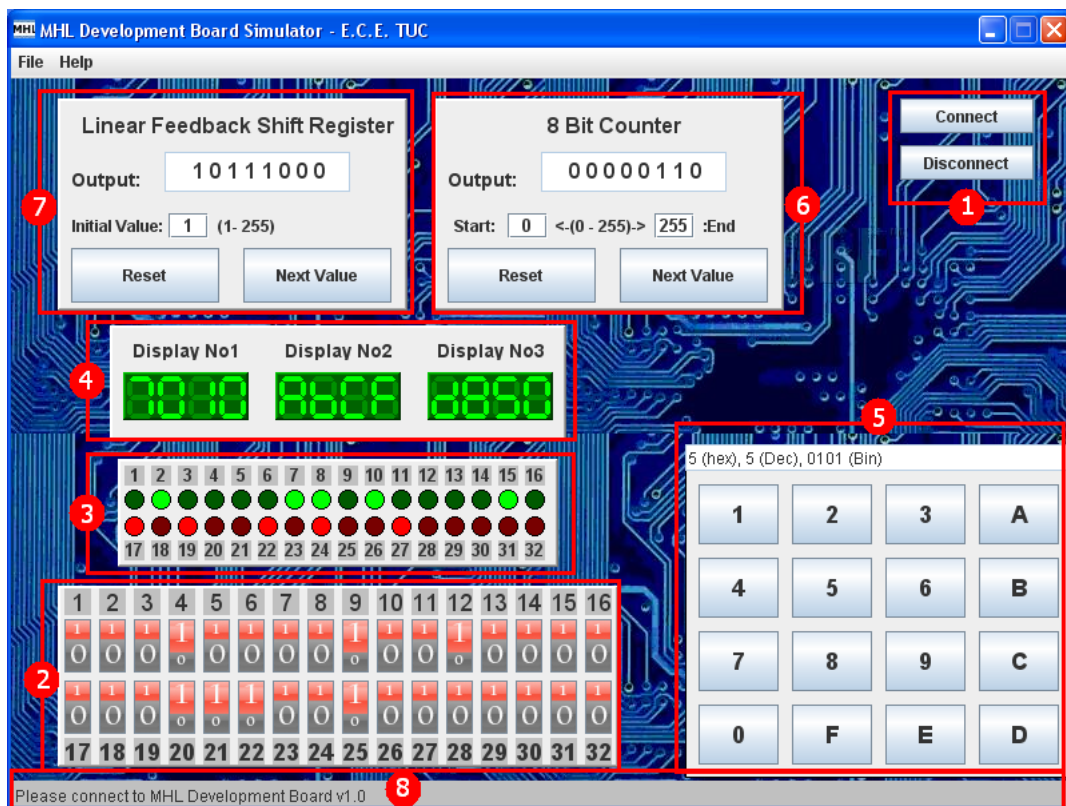
2.2.2 Φυσική Διάταξη

Στην παρακάτω εικόνα μπορούμε να δούμε τη γραφική διεπαφή. Όπως είναι φανερό, οι λειτουργικές μονάδες της γραφικής διεπαφής είναι αρκετά ευδιάκριτες και απλές ως προς τη χρήση τους.



Σχήμα 2.15: Η γραφική διεπαφή

Οι συστοιχίες των εισόδων(LEDs, 7-Segment Displays) τοποθετούνται στο κέντρο της εφαρμογής, ενώ η τροποποίηση των εξόδων γίνεται εύκολα με απλά πατήματα των κουμπιών στις διάφορες μονάδες(Διακόπτες, Δεκαεξαδικό Πληκτρολόγιο, LFSR, Counter). Πιο αναλυτικά, βλέπουμε παρακάτω:



Σχήμα 2.16: Φυσική διάταξη της γραφικής διεπαφής

1. τους διακόπτες Σύνδεσης και Αποσύνδεσης με την αναπτυξιακή πλακέτα.
2. τους S.P.D.T. Διακόπτες.
3. τις Φωτοδιόδους(LEDs).
4. τις Οθόνες 7 στοιχείων(7-Segment Displays).
5. το Δεκαεξαδικό Πληκτρολόγιο.
6. τον Μετρητή 8-bit(Counter).
7. τη Γεννήτρια Τυχαίων Αριθμών 8-bit(L.F.S.R.).
8. τη Μπάρα Ένδειξης Κατάστασης της συσκευής.

Κεφάλαιο 3

Η Φυσική Διεπαφή με το Χρήστη στο Breadboard

Το κεφάλαιο αυτό στοχεύει στην παρουσίαση και την ανάλυση των τεχνικών σχεδιαστικών επιλογών που κάναμε, σε σχέση με την φυσική διεπαφή ή αλλιώς αναπτυξιακή πλακέτα. Στις ενότητες που ακολουθούν θα παρουσιάσουμε την "καρδιά" της πλακέτας, τον μικροεπεξεργαστή AVR, τα ολοκληρωμένα I²C to GPIO, καθώς επίσης και τα υπόλοιπα περιφερειακά εξαρτήματα που πλαισιώνουν τα παραπάνω ολοκληρωμένα.

3.1 Ο μικροελεγκτής AVR

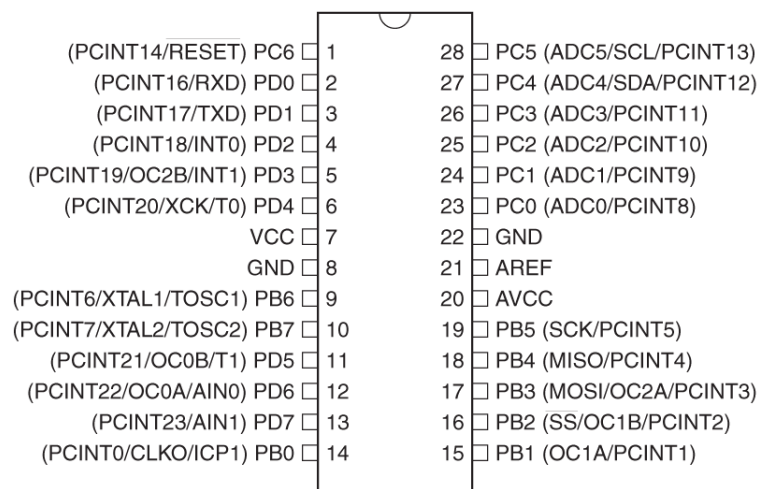
Όπως αναφέραμε στην αρχιτεκτονική της φυσικής διεπαφής, ο πυρήνας της πλακέτας είναι ένας μικροελεγκτής AtmelTM AVR. Ο μικροελεγκτής AVR που χρησιμοποιήσαμε δεν επιλέχτηκε τυχαία. Χρειαζόμασταν έναν μικροελεγκτή, ο οποίος θα διέθετε τουλάχιστον 2kBytes μνήμη Flash, θα παρείχε I²C(TWI) δίαυλο, θα προσέφερε Timers/Counters για τις γεννήτριες συχνοτήτων, θα λειτουργούσε συχνότητες από 12MHz έως 20MHz και, τέλος, θα παρείχε ικανοποιητικό αριθμό ακροδεκτών εισόδων και εξόδων για τη σύνδεση των υπόλοιπων περιφερειακών. Μετά από σχετική έρευνα, επιλέξαμε τον Atmel AVR ATMEGA168-20PU, ο οποίος διαθέτει:

- Αρχιτεκτονική Advanced RISC.
- Σύνολο εντολών: 131 Powerful Instructions – Most Single Clock Cycle Execution.
- Ενσωματωμένο πολλαπλασιαστή 2 κύκλων.
- 16K Bytes In-System Self-programmable μνήμη Flash.

- 512 Bytes μνήμη EEPROM.
- 1K Bytes ενσωματωμένη μνήμη SRAM.
- Προγραμματισμός με In-System Programming τεχνολογία.
- Λειτουργία Read-While-Write.
- Δύο 8-bit Timer/Counters με αποκλειστικό διαιρέτη συχνότητας και Compare Mode.
- Ένα 16-bit Timer/Counter με αποκλειστικό διαιρέτη συχνότητας, Compare Mode και Capture Mode.
- Real Time Counter με αποκλειστικό ταλαντωτή.
- Έξι PWM και CTC κανάλια.
- Προγραμματιζόμενη Serial USART.
- Master/Slave SPI σειριακή διεπαφή.
- Two-Wire σειριακό δίαυλο(TWI) Byte-oriented(Philips I²C compatible).
- Interrupt and Wake-up on Pin Change.
- DebugWIRE On-Chip Debug System.
- Power-on Reset and Programmable Brown-out Detection.
- Internal Calibrated Oscillator.
- Εσωτερικές και εξωτερικές πηγές Interrupt.
- Πέντε Sleep καταστάσεις: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby.
- 23 προγραμματιζόμενες I/O γραμμές.
- Τάση λειτουργίας: 2.7 - 5.5 Volts.
- Θερμοκρασία λειτουργίας: -40C έως 85C.
- Ταχύτητα λειτουργίας: 0 - 10 MHz στα 2.7 - 5.5 Volts, 0 - 20 MHz στα 4.5 - 5.5 Volts.
- Χαμηλή ενεργειακή κατανάλωση:

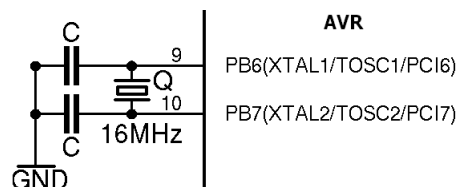
- Active Mode:
 - 250 μ A στο 1 MHz, 1.8V
 - 15 μ A στα 32 kHz, 1.8V (εσωτερικός ταλαντωτής)
- Power-down Mode:
 - 0.1 μ A at 1.8V

Παρακάτω μπορούμε να δούμε τους ακροδέκτες του μικροελεγκτή σε θήκη PDIP-28:



Σχήμα 3.1: Ο μικροελεγκτής AVR

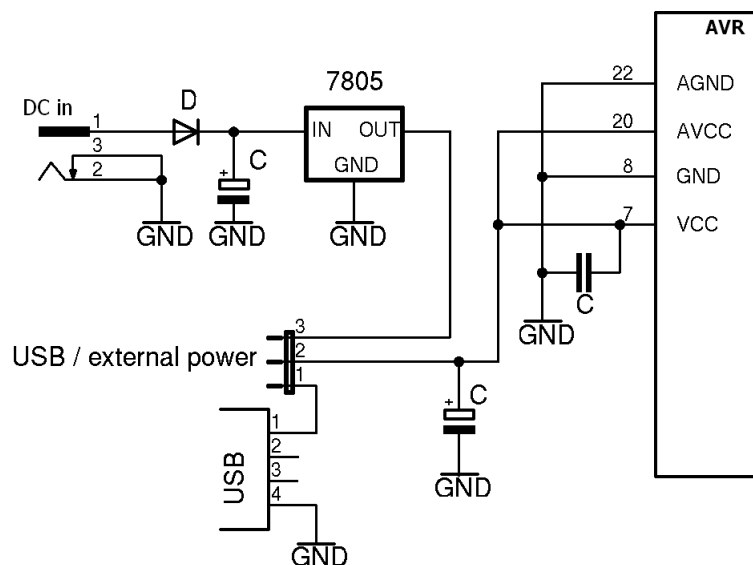
Η συχνότητα λειτουργίας του AVR επιλέχθηκε να είναι αυτή των 16MHz, χωρίς ωστόσο να είναι δεσμευτική, αφού με ελάχιστες αλλαγές στο firmware του μικροελεγκτή μπορεί να τροποποιηθεί στα 12Mhz, 15MHz, 16MHz, ή 20 MHz με χρήση ταλαντωτή κρυστάλλου ή 12.8 MHz και 16.5 MHz με χρήση του εσωτερικού RC ταλαντωτή. Τα 16MHz συχνότητας παράγονται από εξωτερικό κρυσταλλικό ταλαντωτή συνδεδεμένο στους ακροδέκτες 9 και 10 με δύο χωρητικούς πυκνωτές, όπως φαίνεται παρακάτω:



Σχήμα 3.2: Ο κρυσταλλικός ταλαντωτής

3.2 Τροφοδοσία Τάσης

Ο μικροελεγκτής AVR, καθώς και τα υπόλοιπα στοιχεία του κυκλώματος (I²C to GPIO chips), λειτουργούν με τάση τροφοδοσίας 5 Volt. Η θύρα USB είναι ικανή να τροφοδοτήσει το κύκλωμα με τάση 5 Volts, οπότε η χρήση της για τροφοδοσία του κυκλώματος θα ήταν μια αρκετά αξιόπιστη και εύκολη λύση. Ωστόσο, η περιορισμένη παροχή ρεύματος που προσφέρει (500mA από τις προδιαγραφές της θύρας) θα ήταν δεσμευτική στη χρήση της πλακέτας με ενεργοβόρα κυκλώματα υλοποιημένα στο breadboard, αν αναλογιστούμε ότι η πλακέτα σε κατάσταση αναμονής καταναλώνει περί τα 200mA έως 400mA περίπου. Γι' αυτό το λόγο, αποφασίσαμε να συμπεριλάβουμε ένα κύκλωμα παροχής εξωτερικής τροφοδοσίας στην πλακέτα κάνοντας χρήση ενός κλασικού σταθεροποιητή τάσης 7805 ικανού να παρέχει έως 1000mA. Τοποθετήσαμε, επίσης, ένα διακόπτη επιλογής της παροχής τάσης, έτσι ώστε ο χρήστης να μπορεί να τροφοδοτήσει την αναπτυξιακή πλακέτα από την θύρα USB για μικρά, μη ενεργοβόρα κυκλώματα. Σημειώνουμε ότι στην είσοδο του 7805 μπορεί να εφαρμοστεί τάση έως 18 Volts. Η συνδεσμολογία φαίνεται παρακάτω:

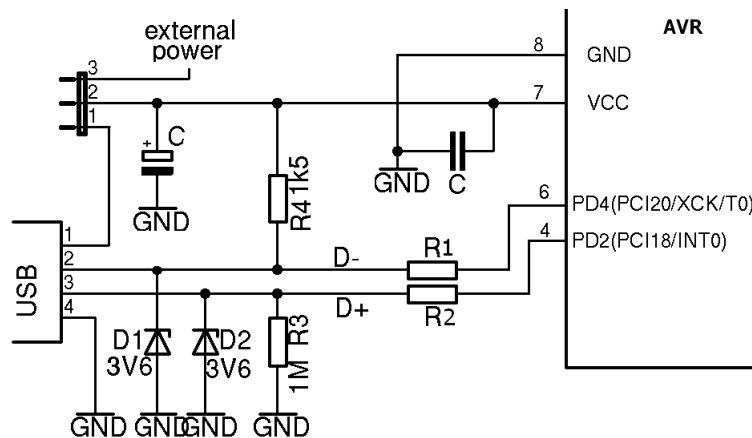


Σχήμα 3.3: Κύκλωμα τροφοδοσίας

3.3 Ο διάυλος USB

Η υλοποίηση του διαύλου USB, όπως αναφέραμε στο προηγούμενο κεφάλαιο, βασίστηκε στο λογισμικό του μικροελεγκτή και όχι σε κάποιον ελεγκτή. Χωρίς να αποκλίνουμε από τις προδιαγραφές του USB, χρειάστηκε να πλαισιώσουμε το

firmware με κάποια ακόμα ηλεκτρονικά στοιχεία για να επιτύχουμε τη ζεύξη της συσκευής με τον υπολογιστή. Όπως φαίνεται παρακάτω,



Σχήμα 3.4: Κύκλωμα USB

οι γραμμές D+ και D- της USB συνδέονται μέσω των δύο αντιστάσεων R1 και R2 στους ακροδέκτες του μικροελεγκτή που αντιστοιχούν στις -ορισμένες στο firmware- γραμμές όπου ο AVR προσομοιώνει τη λειτουργία της USB. Ακόμα, η αντίσταση pull-up R4(1.5kOhms) πληροφορεί τον υπολογιστή πως η συσκευή που συνδέεται είναι χαμηλής ταχύτητας(Low-speed) κατά τη σύνδεση.

Όπως τονίσαμε, θέλοντας να παραμείνουμε στις προδιαγραφές του USB χρειάστηκε να κάνουμε κάποιους συμβιβασμούς. Οι συσκευές που βασίζονται στην υλοποίηση V-USB[5] δεν μπορούν απόλυτα να πειθαρχήσουν στις ηλεκτρικές προδιαγραφές του USB, διότι τα ηλεκτρικά χαρακτηριστικά των I/O του AVR δεν ικανοποιούν τις απαιτήσεις της θύρας. Πιο αναλυτικά, κάθε πλευρά της ζεύξης(υπολογιστής και συσκευή) αναγνωρίζει διαφορετικά επίπεδα τάσης ως υψηλή ή χαμηλή λογική στάθμη. Έτσι, κάθε πλευρά πρέπει να παράγει τις υψηλές ή χαμηλές λογικές στάθμες σε δυναμικό τέτοιο, ώστε να γίνονται αντιληπτές από την άλλη πλευρά για να διατηρείται αξιόπιστο επίπεδο επικοινωνίας. Αφού, λοιπόν, δεν μπορούσαμε να παρέμβουμε στην πλευρά του υπολογιστή, έπρεπε με κάποιο τρόπο να προσαρμόσουμε τον AVR ώστε να γίνεται αντιληπτός από τον υπολογιστή όταν έστελνε δεδομένα ή εντολές.

Ο υπολογιστής στέλνει 0 Volts για την χαμηλή λογική στάθμη και 3.3 Volts για την υψηλή. Αυτά τα επίπεδα είναι αρκετά ακριβή, ακόμα κι αν οι προδιαγραφές παρέχουν ένα ποσοστό απόκλισης. Αυτή η κατεύθυνση των δεδομένων δεν αποτελεί συνήθως πρόβλημα, ακόμα και στην περίπτωση όπου ο AVR τροφοδοτείται με τάση 5 Volts, αφού μπορεί να ανιχνεύσει τα 3.3 Volts σαν λογικό "1".

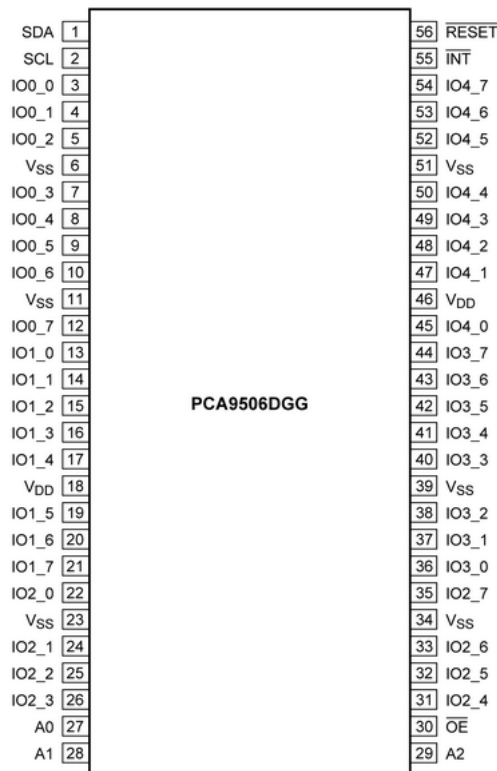
Η άλλη κατεύθυνση είναι πιο προβληματική. Ο υπολογιστής περιμένει 0 έως 0.8 Volts για το λογικό "0" και 2 έως 3.6 Volts για το λογικό "1". Κάποιοι υπολογι-

στές, βέβαια, επιτρέπουν έως 5 Volts για την υψηλή λογική στάθμη, αλλά δεν αποτελούν την πλειοψηφία. Έτσι, για να συμβαδίσουμε με αυτές τις προδιαγραφές, χρησιμοποιήσαμε ένα είδος μετατροπής των επιπέδων τάσης στις γραμμές επικοινωνίας. Για να πετύχουμε αυτή τη μετατροπή, τοποθετήσαμε δύο διόδους Zener 3.6 Volts χαμηλής κατανάλωσης(μικρή παρασιτική χωρητικότητα) στις γραμμές D+ και D-, όπως φαίνεται παραπάνω. Αυτό έχει σαν αποτέλεσμα η μέγιστη τάση στην έξοδο του AVR να περιορίζεται στα 3.6 Volts από τα 5 Volts που παράγει.

Τέλος, η επιλογή των ακροδεκτών του AVR για τις γραμμές D+ δεν D- είναι τυχαία. Η υλοποίηση του firmware θέλει την γραμμή D+ συνδεδεμένη στον Interrupt 0 του AVR και την D- σε ακροδέκτη της ίδια πόρτας του μικροελεγκτή. Και οι δύο γραμμές συνδέονται στην πόρτα D στους ακροδέκτες 2 και 4, αντίστοιχα.

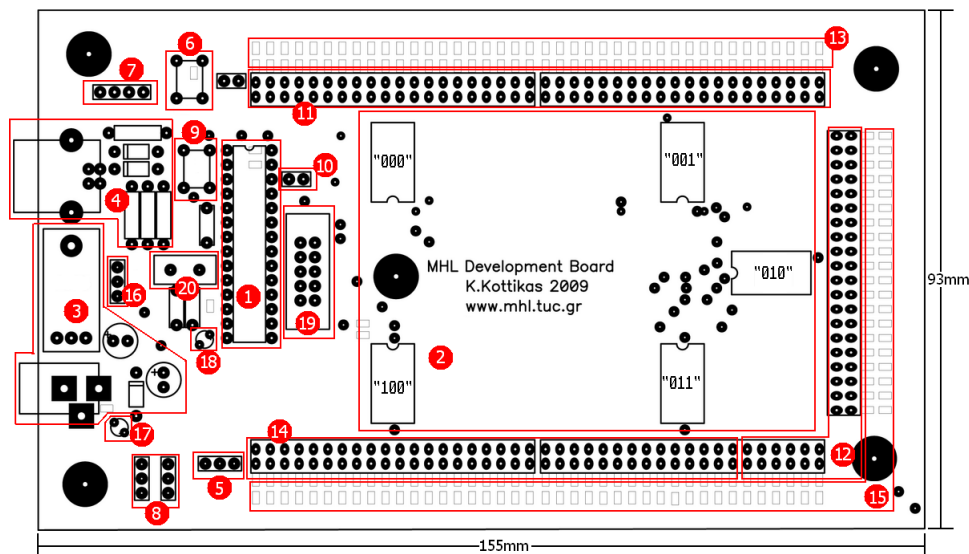
3.4 Τα ολοκληρωμένα I²C to GPIO

Όπως αναφέραμε στις προδιαγραφές της διπλωματικής εργασίας, η αναπτυξιακή πλακέτα μας έπρεπε να διαθέτει μεγάλο αριθμό εισόδων και εξόδων ώστε να παρέχει μεγάλο και ποικίλο αριθμό δομικών μονάδων για τον πειραματισμό του χρήστη. Αυτό κατέστη δυνατό με τη χρήση ολοκληρωμένων κυκλωμάτων, τα οποία χρησιμοποιούν δύο καλώδια, ένα για δεδομένα και ένα για ρολόι, και προσφέρουν 40 ακροδέκτες προγραμματιζόμενων εισόδων/εξόδων. Τα ολοκληρωμένα που χρησιμοποιήσαμε είναι τα PCA9506DGG της NXPTM και παρουσιάζονται παρακάτω:



Σχήμα 3.5: Το ολοκληρωμένο PCA9506

Το κάθε ολοκληρωμένο διαθέτει 5 πόρτες των 8 προγραμματιζόμενων εισόδων/εξόδων (IO_0 έως IO_4) έκαστη, δηλαδή 40 ακροδέκτες για εισόδους ή εξόδους. Επίσης, διαθέτει δύο ακροδέκτες, ένα για ρολόι (SCL) και ένα για τις εντολές και τα δεδομένα του διαύλου I²C (SDA). Τέλος, οι τρεις ακροδέκτες A0, A1 και A2 διευθυνσιοδοτούν τα ολοκληρωμένα στο δίαυλο. Τα ολοκληρωμένα παρατάσσονται περιμετρικά της πλακέτας και οι διευθύνσεις τους φαίνονται στην παρακάτω εικόνα στη θέση 2:



Σχήμα 3.6: Φυσική διάταξη της αν. πλακέτας

Όπως περιγράψαμε και στην ενότητα της αρχιτεκτονικής της αναπτυξιακής πλακέτας, οι δύο γραμμές του διαύλου I²C συνδέονται μέσω pull-up αντιστάσεων στη θετική παροχή τάσης. Αυτό είναι απαραίτητο, διότι οι γραμμές είναι open-drain ή open-collector σε όλες τις συμβατές με το πρωτόκολλο συσκευές. Οι αντιστάσεις που επιλέξαμε είναι 4.7 kOhms.

Η συνδεσμολογία των ολοκληρωμένων είναι αρκετά απλή. Οι ακροδέκτες SCL και SDA συνδέονται παράλληλα στους αντίστοιχους ακροδέκτες του AVR. Έτσι, ο AVR ως Master Device στον διάυλο απευθύνεται στα ολοκληρωμένα, δηλαδή τις Slave Devices, χρησιμοποιώντας την μοναδική διεύθυνση του καθενός.

Τέλος, σημειώνουμε ότι τα ολοκληρωμένα είναι επιφανειακής στήριξης(SMD) και έρχονται σε θήκη TSSOP56.

3.5 Οι ακροδέκτες Εισόδων

Οι ακροδέκτες Εισόδων συνδέονται στους ακροδέκτες των ολοκληρωμένων I²C to GPIO ένας προς έναν. Στους ακροδέκτες Εισόδων έχουμε τοποθετήσει pull-down αντιστάσεις ώστε όταν κάποιος ακροδέκτης είναι αποσυνδεδεμένος το δυναμικό του να παραμένει χαμηλό. Σε αντίθετη περίπτωση η τιμή εισόδου του ακροδέκτη θα ήταν απροσδιόριστη και τυχαία. Οι pull-down αντιστάσεις που χρησιμοποιήσαμε είναι 4.7 kOhms, ικανές να παρέχουν το χαμηλό δυναμικό "γρήγορα" κατά την αποσύνδεση του ακροδέκτη, αλλά και η τιμή τους είναι τέτοια ώστε η κατανάλωση ρεύματος να παραμένει σε χαμηλά επίπεδα. Κάνοντας χρήση αντιστάσεων 4.7 kOhms με παροχή τάσης στα 5 Volts, το ρεύμα που θα καταναλώνει η κάθε αντίσταση θα είναι:

$$\begin{aligned}
I = \frac{V}{R} &= \frac{5V}{4.7kOhms} \\
&= \frac{5V}{4.7 \times 10^3 Ohms} \\
&= 1.06 \times 10^{-3} Amperes \\
&\approx 1mA
\end{aligned}$$

δηλαδή στο σύνολο των 80 εισόδων, οι 80 pull-down αντιστάσεις θα καταναλώ-
νουν **80 mA**.

3.6 Γεννήτριες Συχνοτήτων

Σύμφωνα με την παρακάτω σχέση, που περιγράψαμε στην αρχιτεκτονική της αναπτυξιακής πλακέτας, όπου N είναι ο διαιρέτης της συχνότητας του επεξεργα-
στή και $TOPValue$ η "Ανώτερη τιμή",

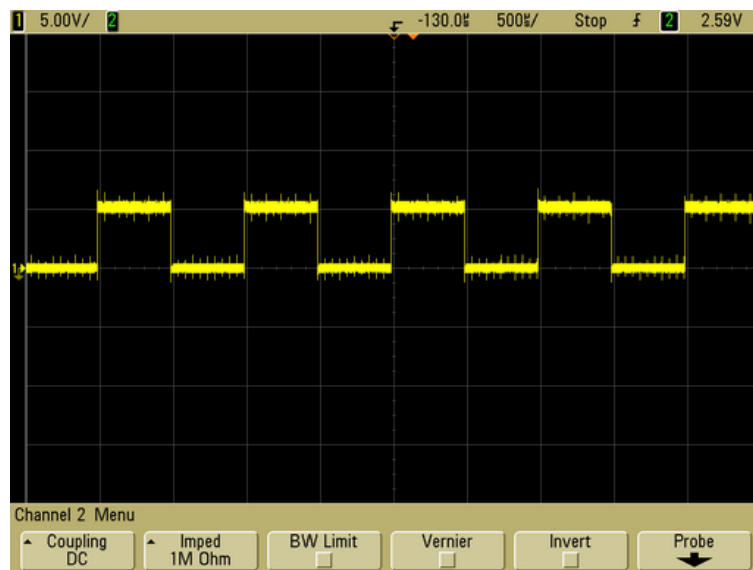
$$f = \frac{f_{CPU}}{2 \times N \times (1 + TOPValue)}$$

οι συχνότητες 1kHz και 10kHz παράγονται με τους εξής συνδυασμούς διαιρέτη
και Ανώτερης τιμής:

- Συχνότητα 1kHz στον Timer/Counter1, ακροδέκτης 15 του AVR
 - $N = 8$
 - $TOPValue = 99_{decimal}, 63_{hex}$
- Συχνότητα 10kHz στον Timer/Counter0, ακροδέκτης 12 του AVR
 - $N = 8$
 - $TOPValue = 999_{decimal}, 03E7_{hex}$

Σημειώνουμε ότι ο κάθε Counter μετράει από το **0** έως την μέγιστη τιμή του. Συνεπώς η τιμή $TOPValue = 99_{decimal}$ και $TOPValue = 999_{decimal}$ είναι η **100** και **1000** τιμή του Counter, αντίστοιχα. Επίσης, οι παραπάνω τιμές ισχύουν για ρολόι του μικροελεγκτή στα 16MHz. Αν το ρολόι αλλάξει, οι τιμές πρέπει να υπολογιστούν εκ νέου.

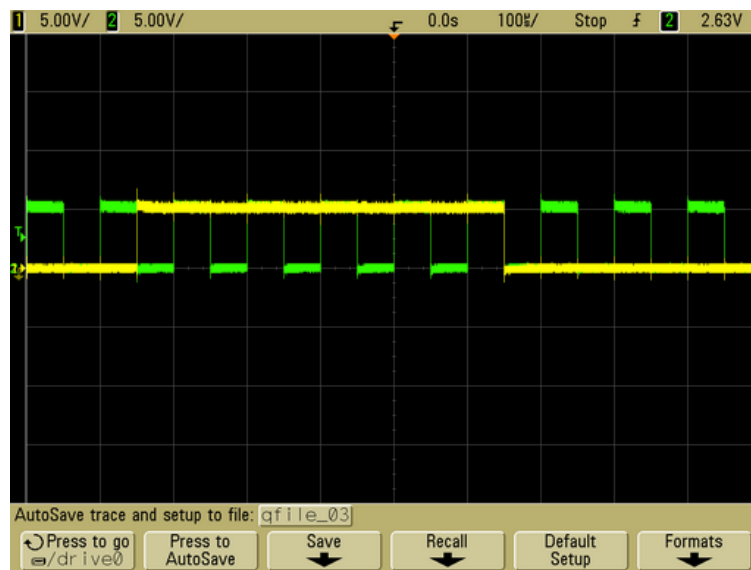
Στις εικόνες που ακολουθούν παρουσιάζονται οι μετρήσεις των συχνοτήτων στον παλμογράφο:



Σχήμα 3.7: Συχνότητα 1kHz



Σχήμα 3.8: Συχνότητα 10kHz



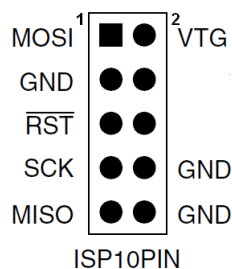
Σχήμα 3.9: Συχνότητα 1kHz και 10kHz

3.7 Γεννήτρια Τετραγωνικών Ωρολογιακών Παλμών

Ο τετραγωνικός ωρολογιακός παλμός δημιουργείται κάθε φορά που ένας πιεστικός διακόπτης κλείνει (Pulse Button). Ο πιεστικός διακόπτης συνδέεται στον ακροδέκτη 5 του AVR, όπου βρίσκεται και ο *Interrupt 1* του μικροελεγκτή και έχει την αμέσως μεγαλύτερη προτεραιότητα εκτέλεσης (τρίτη σε σειρά) μετά τις εντολές και τα δεδομένα που λαμβάνονται ή στέλνονται στη θύρα USB (θυμίζουμε ότι η γραμμή D- είναι συνδεδεμένη στον *Interrupt 0* και έχει την δεύτερη προτεραιότητα εκτέλεσης σε σειρά μετά το Reset). Ο *Interrupt 1* προκαλεί την εκτέλεση μιας ρουτίνας η οποία οδηγεί τον ακροδέκτη 14 του AVR σε υψηλό δυναμικό για χρόνο ίσο με 0.5msec (παλμός συχνότητας της τάξης των 5kHz).

3.8 Προγραμματισμός Μικροελεγκτή

Για τον εύκολο προγραμματισμό του μικροελεγκτή συμπεριλάβαμε στη σχεδίασή μας μια θύρα προγραμματισμού 10 ακροδεκτών σύμφωνα με της προδιαγραφές της Atmel™, όπως φαίνεται παρακάτω:



Σχήμα 3.10: Υποδοχή ISP

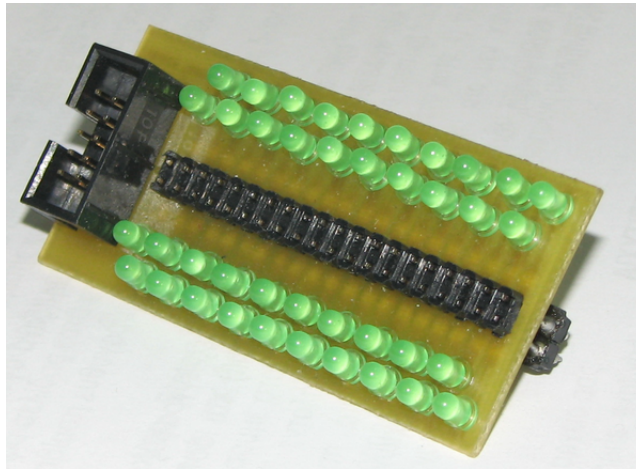
Ο πίνακας που ακολουθεί εξηγεί τον κάθε ακροδέκτη.

Σήμα	Ακροδέκτης	I/O	Περιγραφή
VTG	2	-	Τροφοδοσία
GND	8,10	-	Γείωση
MOSI	1	Έξοδος	Εντολές και Δεδομένα προς τον AVR
MISO	9	Είσοδος	Δεδομένα από τον AVR
SCK	7	Έξοδος	Σειριακό ρολόι
RESET	5	Έξοδος	Reset

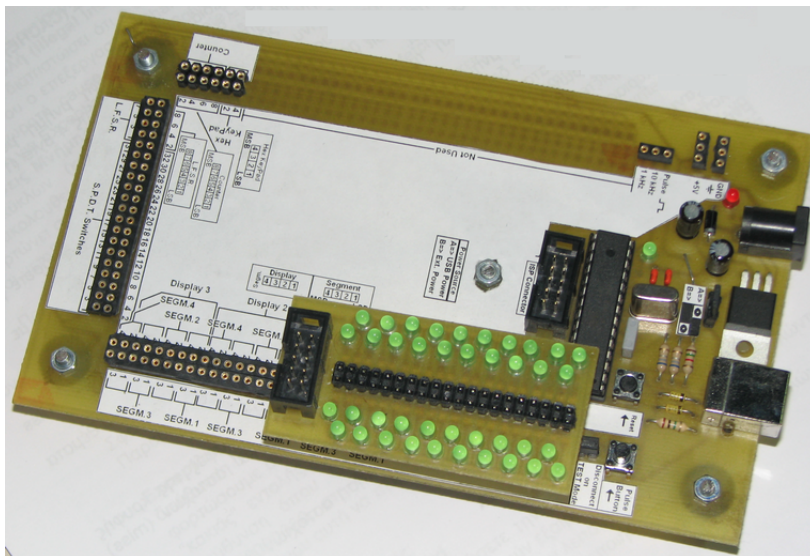
3.9 Δοκιμαστική Πλακέτα Ελέγχου

Όπως κάθε ηλεκτρονική συσκευή, έτσι και η φυσική διεπαφή που σχεδιάσαμε ενδέχεται να υποστεί βλάβες ή να καταστραφεί, εν μέρει ή εξολοκλήρου, κατά την χρήση της στην πειραματική διαδικασία. Για να γίνει πιο εύκολη η επιβεβαίωση της ορθής λειτουργίας της φυσικής διεπαφής λόγω των πολλών ακροδεκτών, σχεδιάσαμε μια δοκιμαστική πλακέτα γι' αυτόν ειδικά το σκοπό, η οποία μας βοήθησε καθ' όλη τη διάρκεια της υλοποίησης μας στη αποσφαλμάτωση του firmware. Η δοκιμαστική πλακέτα περιλαμβάνει 40 LEDs που καταλήγουν σε 40 ακροδέκτες τοποθετημένους με τέτοιο τρόπο ώστε να εφάπτονται απόλυτα στους ακροδέκτες της αναπτυξιακής πλακέτας.

Η χρήση της είναι αρκετά εύκολη και απλή. Αφού προσαρμόσουμε τη δοκιμαστική πλακέτα επάνω στην αναπτυξιακή, αρκεί να κλείσουμε το διακόπτη δοκιμαστικής λειτουργίας και να τροφοδοτήσουμε την αναπτυξιακή πλακέτα με ρεύμα. Κατά την δοκιμαστική λειτουργία όλοι οι ακροδέκτες εισόδων και εξόδων μετατρέπονται σε εξόδους. Αρχικά, και για μερικά δευτερόλεπτα, σε όλες τις εξόδους θα εφαρμοστεί υψηλό δυναμικό και τα LEDs της δοκιμαστικής πλακέτας θα φωτοβολήσουν. Στη συνέχεια, τα LEDs θα σβήσουν και θα αρχίσουν να αναβοσβήνουν διαδοχικά, έτσι ώστε να δοκιμαστούν οι ακροδέκτες και τα ολοκληρωμένα I²C to GPIO ένα προς ένα, εξονυχιστικά. Παρακάτω, ακολουθούν ενδεικτικές φωτογραφίες:



Σχήμα 3.11: Η δοκιμαστική πλακέτα



Σχήμα 3.12: Η δοκιμαστική πλακέτα τοποθετημένη στην αναπτυξιακή

3.10 Εργαλεία Ανάπτυξης Λογισμικού(firmware)

Η ανάπτυξη του λογισμικού του AVR έγινε στο λογισμικό WinAVR, έκδοση 20090313, το οποίο χρησιμοποιεί τον avr-gcc compiler.

3.11 Εργαλεία Σχεδίασης Κυκλώματος

Για την σχεδίαση των κυκλωμάτων χρησιμοποιήσαμε τα σχεδιαστικά προγράμματα Altium P-CAD 2006 και Cadence SPECCTRA. Αρχικά, δημιουργήσαμε τις απαραίτητες βιβλιοθήκες για τα διάφορα εξαρτήματα και έπειτα σχεδιάσαμε τα σχηματικά διαγράμματα(schematics). Ακολούθησε η κατασκευή των διαγραμμάτων των τυπωμένων κυκλωμάτων και, τελικά, κατασκευάσαμε τις πλακέτες με τον εξοπλισμό του εργαστηρίου.

3.12 Εκτύπωση και Συναρμολόγηση Τυπωμένου Κυκλώματος

Μετά την σχεδίαση των τυπωμένων κυκλωμάτων κατασκευάσαμε τις πλακέτες στο εργαστήριο Μικροεπεξεργαστών και Υλικού, χρησιμοποιώντας φωτοευαίσθητες πλακέτες και σχετικό εξοπλισμό. Μετά την εκτύπωση των πλακετών ακολούθησε η διάτρησή τους και, και τελικά, η συναρμολόγησή τους.

Κεφάλαιο 4

Η Λειτουργική Διεπαφή με το Χρήστη στον Η/Υ

Αναπόσπαστο κομμάτι της αναπτυξιακής πλατφόρμας που σχεδιάσαμε αποτελεί η Λειτουργική Διεπαφή. Το μέρος αυτό είναι εξίσου σημαντικό με την φυσική διεπαφή, διότι αποτελεί το υψηλότερο επίπεδο όσον αφορά την διαδραστικότητα μεταξύ χρήστη και πειραματικού κυκλώματος μετά την υλοποίηση του τελευταίου στο breadboard. Στο επίπεδο αυτό ο χρήστης παρακολουθεί όλα τα σήματα εισόδων και μεταβάλλει κατά βούληση τα σήματα εξόδων.

Η γραφική διεπαφή σχεδιάστηκε και υλοποιήθηκε με τη γλώσσα προγραμματισμού JAVA. Η επιλογή της γλώσσας δεν έγινε τυχαία. Επιθυμούσαμε το λογισμικό, που θα συνόδευε την φυσική διεπαφή, να είναι ικανό να εκτελείται σε διάφορα λειτουργικά συστήματα(Cross-platform software).

Το λογισμικό δομείται γύρω από μια κύρια κλάση. Η κλάση αυτή αποτελεί τη γραφική εφαρμογή. Κάθε μία λειτουργική μονάδα υλοποιείται σε δικιά της κλάση ξεχωριστά, ενώ συνολικά οι κλάσεις αυτές διαχειρίζονται από την κύρια. Ακόμα, μια κλάση αναλαμβάνει να υλοποιήσει και να διαχειριστεί το νήμα(thread) που ευθύνεται για την συλλογή των εισόδων προς την γραφική διεπαφή, ενώ η αποστολή των δεδομένων γίνεται από ρουτίνες που διαθέτουν οι μονάδες εξόδου, οι οποίες έχουν υλοποιηθεί στην κάθε μια μονάδα ξεχωριστά.

Πιο αναλυτικά, τα δεδομένα εισόδων και εξόδων αποθηκεύονται σε δύο καταχωρητές(buffers). Ο buffer εισόδων ονομάζεται *inputsBuffer* και ο buffer εξόδων *outputsBuffer*. Το μέγεθος των καταχωρητών είναι 10 και 7 bytes αντίστοιχα και είναι αρκετό για να διατηρήσει τον όγκο των δεδομένων.

Το νήμα, που είναι υπεύθυνο για την ανάγνωση των εισόδων, αρχίζει να εκτελείται επαναλαμβανόμενα μετά την έναρξη της εφαρμογής και την επιτυχή σύνδεση της συσκευής. Μετά την έναρξη της εκτέλεσής του, μια ρουτίνα που καλείται κάθε 50msec "ζητάει" στέλνοντας την αντίστοιχη εντολή στη συσκευή μέσω του

διαύλου USB 10 bytes που αντιστοιχούν στα δεδομένα εισόδου. Αν η συσκευή δεν επιστρέψει το σωστό αριθμό bytes, τότε τα δεδομένα του καταχωρητή εισόδων δεν ανανεώνονται. Η αποστολή των δεδομένων εξόδων γίνεται διαφορετικά. Τα bytes των δεδομένων τροποποιούνται κάθε φορά που τροποποιείται κάποια έξοδος από τον χρήστη. Μόλις η τιμή της λειτουργικής μονάδας εξόδου μεταβληθεί, ο αντίστοιχος καταχωρητής ενημερώνεται και ακολουθεί η εκτέλεση της αρμόδιας ρουτίνας της μονάδας για την αποστολή των δεδομένων προς την συσκευή. Η αποστολή επαναλαμβάνεται έως ότου η συσκευή απαντήσει πως έλαβε τον σωστό αριθμό bytes.

Οι λειτουργικές μονάδες εισόδων(LEDs, οθόνες 7-segment) εξαρτώνται από τα δεδομένα του καταχωρητή εισόδων. Το κάθε στοιχείο των μονάδων αυτών, είτε είναι LED, είτε είναι segment, υλοποιείται με ένα νήμα(thread). Τα κάθε νήμα διαβάζει επαναλαμβανόμενα τα περιεχόμενα του buffer και ανανεώνει την γραφική του κατάσταση. Κάθε φορά που τα δεδομένα του καταχωρητή ανανεώνονται, ακολουθεί και η ανανέωση των γραφικών.

Τα δεδομένα εισόδων και εξόδων διέρχονται από διάφορα στάδια από και προς την γραφική διεπαφή. Η γραφική διεπαφή χρησιμοποιεί τα προγράμματα οδήγησης LibUSB για να επικοινωνήσει με την αναπτυξιακή πλακέτα. Η μεταξύ τους επικοινωνία δεν γίνεται άμεσα. Μεταξύ της γραφικής εφαρμογής και των οδηγών παρεμβάλλεται ένας wrapper, που ονομάζεται JAVA LibUSB wrapper[15].

4.1 Ο JAVA LibUSB Wrapper

Δουλειά του wrapper είναι να παρέχει στον σχεδιαστή του λογισμικού όλα τα εργαλεία που θα τον βοηθήσουν να χρησιμοποιήσει τους οδηγούς LibUSB στις διάφορες JAVA εφαρμογές. Η χρήση του δίνει τη δυνατότητα στο σχεδιαστή να έχει πρόσβαση σε USB συσκευές που συνδέονται στον υπολογιστή, ανεξαρτήτως του λειτουργικού συστήματος. Ο JAVA LibUSB Wrapper μπορεί να συνεργαστεί με οποιαδήποτε πλατφόρμα WindowsTM, Linux και Mac OSTM. Απαραίτητη προϋπόθεση για την λειτουργία του είναι η χρήση των προγραμμάτων οδήγησης LibUSB 0.1.xx.x.

4.2 Οι οδηγοί LibUSB

Οι οδηγοί LibUSB 0.1 στοχεύουν στη δημιουργία μιας βιβλιοθήκης, η οποία θα χρησιμοποιείται από τους σχεδιαστές λογισμικού, για την εύκολη πρόσβαση σε USB συσκευές, ανεξαρτήτως του λειτουργικού συστήματος. Οι οδηγοί LibUSB υποστηρίζουν τα λειτουργικά Linux, FreeBSD, NetBSD, OpenBSD, Darwin, Mac OS X και Windows.

4.3 Εργαλεία Ανάπτυξης

Το περιβάλλον ανάπτυξης της εφαρμογής είναι το Eclipse SDK, έκδοσης 3.4.2. Κατά τη σχεδίαση χρησιμοποιήθηκε και το γραφικό εργαλείο Visual Editor Eclipse Plug-In, έκδοσης 1.4.0. Προσοχή πρέπει να δοθεί στο API της JAVA που βασίστηκε η υλοποίηση. Η ορθή λειτουργία της εφαρμογής είναι γνωστό ότι επιτυγχάνεται με το πακέτο JRE 1.6 (ή ανώτερο) της SUNTM.

Κεφάλαιο 5

Επιβεβαίωση Λειτουργίας

Οι ενότητες που ακολουθούν περιγράφουν την διαδικασία πιστοποίησης της λειτουργίας της νέας πλατφόρμας.

5.1 Δοκιμές κατά την σχεδίαση και υλοποίηση

Απαραίτητο βήμα για την ολοκλήρωση της διπλωματικής εργασίας ήταν η επιβεβαίωση της λειτουργίας της πλατφόρμας. Η πιστοποίηση της ορθότητας λειτουργίας έγινε, τόσο τμηματικά, όσο και στο σύνολων των επιμέρους μονάδων.

Οι δοκιμές μας ξεκίνησαν από τα πρώτα κιόλας στάδια της υλοποίησης. Αφού συναρμολογήσαμε το κύκλωμα της USB στο breadboard, χρησιμοποιήσαμε μια πρώτη έκδοση του firmware και της γραφικής διεπαφής για να ανταλλάξουμε bytes δεδομένων μεταξύ της συσκευής και του υπολογιστή.

Μετά τις επιτυχημένες δοκιμές της USB διασύνδεσης, ακολούθησε ο έλεγχος λειτουργίας του διαύλου και των ολοκληρωμένων I²C. Η δεύτερη δοκιμαστική έκδοση του λογισμικού αντάλλαξε δεδομένα μεταξύ ολοκληρωμένων και AVR.

Στην τρίτη δοκιμαστική έκδοση συνδέσαμε όλες τις μονάδες μεταξύ τους. Έπειτα, κατασκευάσαμε την τελική πλακέτα και προχωρήσαμε στην ολοκλήρωση της πρώτης λειτουργικής έκδοσης του firmware και της γραφικής διεπαφής.

Μετά την περάτωση της γραφικής και φυσικής διεπαφής και της λειτουργικότητάς τους, δοκιμάσαμε την πλατφόρμα κάνοντας χρήση μιας μεγάλης ποικιλίας ολοκληρωμένων κυκλωμάτων TTL (NOT, NOR, NAND, Flip-Flop, κ.α.) και ορισμένων μικρών αναλογικών κυκλωμάτων.

5.2 Δοκιμές σε πραγματικές συνθήκες

Εκτός από τις δοκιμές που πραγματοποιήσαμε κατά την σχεδίαση και υλοποίηση του υλικού και του λογισμικού, θεωρήσαμε εύλογο να αξιολογήσουμε την πλατφόρμα μέσα από μια σειρά ενδεικτικών πειραμάτων υπό πραγματικές συνθήκες. Οι δοκιμές αυτές έλαβαν χώρα στο εργαστήριο Μικροεπεξεργαστών και Υλικού με την συμμετοχή μιας εθελοντικής ομάδας πρωτοετών φοιτητών. Από τους φοιτητές ζητήθηκε η σχεδίαση, υλοποίηση και επιβεβαίωση λειτουργίας των εργαστηριακών ασκήσεων που είχαν διδαχθεί με τη χρήση της νέας πλατφόρμας και, τελικά, η αξιολόγησή της.

Οι πειραματικές ασκήσεις που υλοποίησαν οι εθελοντές είναι οι εξής:

1. Η πρώτη άσκηση περιλάμβανε τον πειραματισμό με απλές πύλες NAND, NOR και NOT και την κατασκευή της συνάρτησης XOR(με δύο μεταβλητές ως $\bar{x}y + x\bar{y}$) με τη χρήση των παραπάνω πυλών.
2. Η δεύτερη άσκηση περιλάμβανε την κατασκευή κυκλώματος συνάρτησης κώδικα Gray τριών εισόδων και τριών εξόδων, κάνοντας χρήση πινάκων Karnaugh και άλγεβρας Boole για την ελαχιστοποίηση της συνάρτησης.
3. Η τρίτη άσκηση περιλάμβανε την επαλήθευση λειτουργίας ενός D Flip-Flop και την κατασκευή ενός σύγχρονου μετρητή.
4. Ως τέταρτη άσκηση, αφήσαμε ελεύθερους τους φοιτητές να πειραματιστούν με την νέα πλατφόρμα και να κατασκευάσουν κυκλώματα τις προτίμησής τους με πύλες και D Flip-Flops.

Μέσα από αυτή τη διαδικασία καταφέραμε να επιβεβαιώσουμε την ορθή λειτουργία της νέας πλατφόρμας από χρήστες χωρίς ιδιαίτερες τεχνολογικές γνώσεις ή εμπειρία με αντίστοιχη τεχνολογία και εξοπλισμό. Το γεγονός αυτό είναι πολύ σημαντικό, αφού η πλατφόρμα στοχεύει σε αυτή την κατηγορία χρηστών. Ουσιαστικά, η χρήση της πλακέτας θα βοηθήσει τους νέους μηχανικούς στην εξοικείωσή τους με τα κυκλώματα TTL και τις εφαρμογές τους.

Τα σχόλια που λάβαμε ήταν ενθαρρυντικά. Οι εθελοντές επικρότησαν την προσπάθειά μας, τόνισαν την φιλικότητα χρήσης της γραφικής διεπαφής και της αναπτυξιακής πλακέτας και δεν εμφάνισαν ιδιαίτερη δυσκολία στην κατανόηση χρήσης και φιλοσοφίας της σχεδίασης. Τέλος, εξέφρασαν ομόφωνα την προτίμησή τους ως προς τη νέα πλατφόρμα έναντι της παλαιάς, γεγονός που μας ικανοποίησε ιδιαίτερα.

Κεφάλαιο 6

Συμπεράσματα

6.1 Απολογισμός

Όπως φάνηκε απ'όσα αναπτύχθηκαν στα προηγούμενα κεφάλαια, οι στόχοι που τέθηκαν κατά την ανάληψη της διπλωματικής εργασίας επιτεύχθηκαν. Πράγματι, σχεδιάστηκε και υλοποιήθηκε μια αναπτυξιακή πλακέτα με πληθώρα λειτουργιών, καθώς και λογισμικό που την συνοδεύει. Η πλακέτα είναι δυνατόν να προγραμματίζεται με απλό τρόπο σε γλώσσα υψηλού επιπέδου χάρη στο λογισμικό που αναπτύχθηκε (firmware). Ακόμα, το μέγεθός της και η USB διεπαφή που προσφέρει παρέχουν φορητότητα και συμβατότητα με όλους τους υπολογιστές της νέας γενιάς, καθώς και με πληθώρα λειτουργικών συστημάτων. Τέλος, είναι κατάλληλη για εκπαιδευτικούς σκοπούς και παρέχει αρκετά βοηθητικά εφόδια στο νέο μηχανικό για την εξοικειώσή του με την τεχνολογία των TTL κυκλωμάτων και όχι μόνο.

Τέλος, σε προσωπικό επίπεδο, η διπλωματική εργασία μου προσέφερε την ικανοποίηση να ολοκληρώσω ένα έργο, ξεκινώντας από τη σύλληψη της ιδέας, περνώντας στον αρχικό σχεδιασμό, συνεχίζοντας με έρευνα αγοράς, κάνοντας σχεδιαστικούς συμβιβασμούς και υλοποιώντας, τελικά, το υλικό και το λογισμικό.

6.2 Μελλοντική Εργασία

Ως μελλοντική εργασία θα προτείναμε την επέκταση της αναπτυξιακής πλακέτας και της γραφικής διεπαφής με περισσότερες δυνατότητες προσθέτοντας κι άλλες λειτουργικές μονάδες, όπως για παράδειγμα περισσότερους πιστικούς διακόπτες για δημιουργία παλμών, αύξηση του μεγέθους του μετρητή και της γεννήτριας τυχαίων αριθμών, προσθήκη ενός ηχητικού βομβητή(buzzer), καθώς επίσης και ενός Logic Probe.

Ο σχεδιασμός και η υλοποίηση της πλακέτας επιτρέπει την αύξηση των εισόδων και εξόδων από 132 σε 200, καθώς επίσης και την υλοποίηση της διασύνδεσης της γραφικής διεπαφής με σειριακές RS-232 συσκευές, διαμέσου της αναπτυξιακής πλακέτας.

Παράρτημα Α

Εγχειρίδια

Εγχειρίδιο Χρήστη

Εγκατάσταση οδηγών(Drivers)

Για την εγκατάσταση της συσκευής στο Η/Υ σας θα χρειαστείτε τα εξής:

- Τη αναπτυξιακή πλακέτα "MHL Development Board".
- Ένα καλώδιο USB.
- Τους οδηγούς(drivers) για το λειτουργικό σας.

Σημείωση: Προτού αρχίσετε τη διαδικασία εγκατάστασης, σιγουρευτείτε πως έχετε δικαιώματα διαχειριστή(administrator).

Windows XP™(x86)

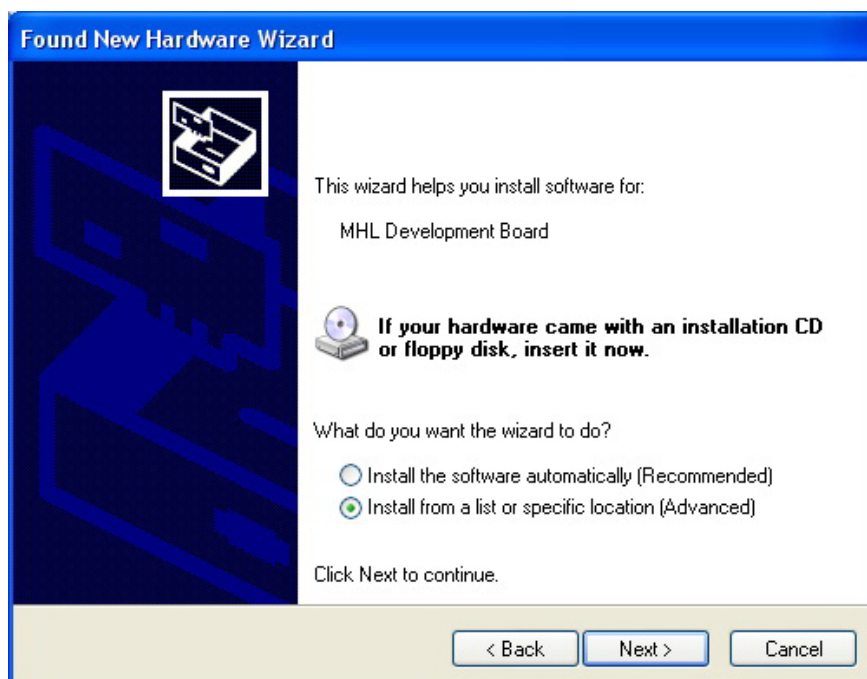
1. Συνδέστε τη συσκευή στο καλώδιο, και στη συνέχεια συνδέστε το καλώδιο σε μία κενή θύρα USB στον υπολόγιστή σας. Το ακόλουθο μήνυμα θα εμφανιστεί:



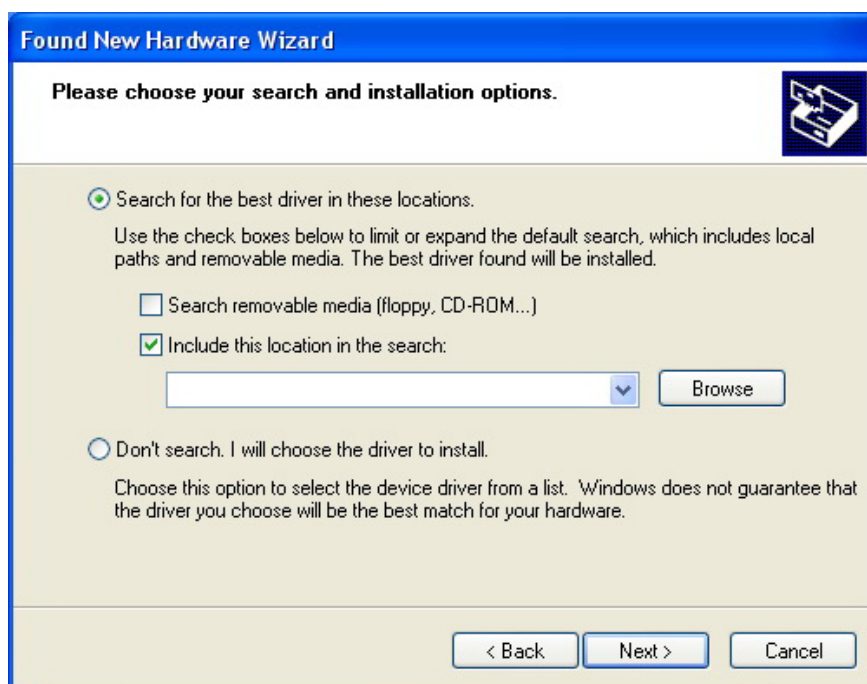
2. Αφού ξεκινήσει ο οδηγός εγκατάστασης νέου υλικού, επιλέξτε "Όχι, αυτή τη φορά" και πατήστε το κουμπί "Επόμενο".



3. Επιλέξτε "Εγκατάσταση από λίστα(Για προχωρημένους)" και πατήστε το κουμπί "Επόμενο".

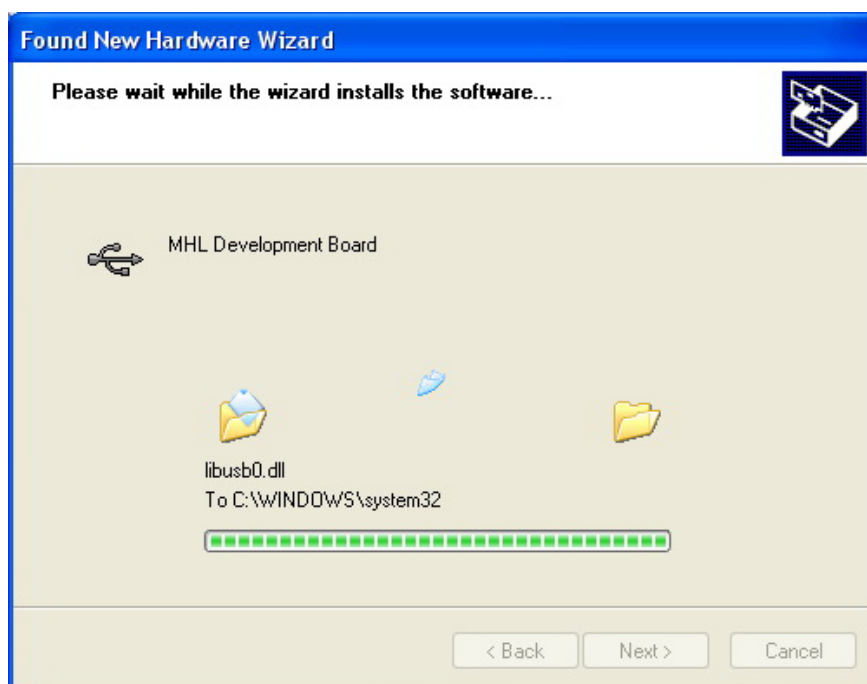


4. Επιλέξτε τα παρακάτω και πατήστε το κουμπί "Αναζήτηση".

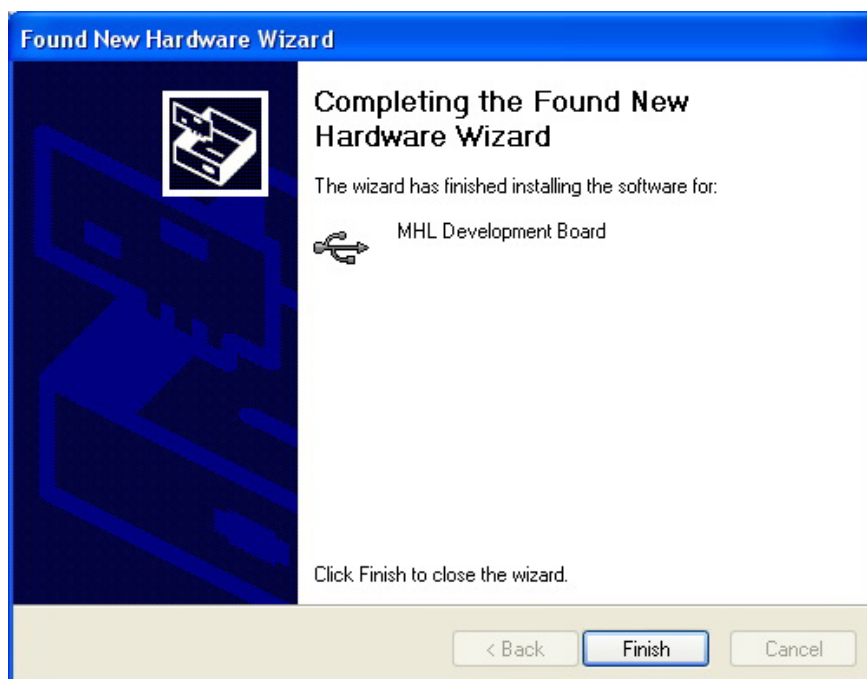


5. Επιλέξτε το φάκελο που έχετε αποσυμπίεσει τους οδηγούς, πατήστε "OK" και στη συνέχεια "Επόμενο".





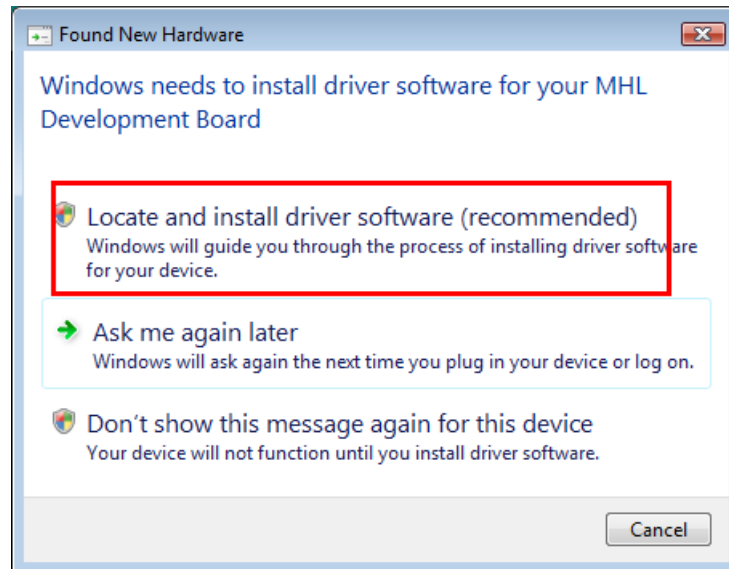
6. Το νέο υλικό, θα πρέπει να έχει εγκατασταθεί με επιτυχία.



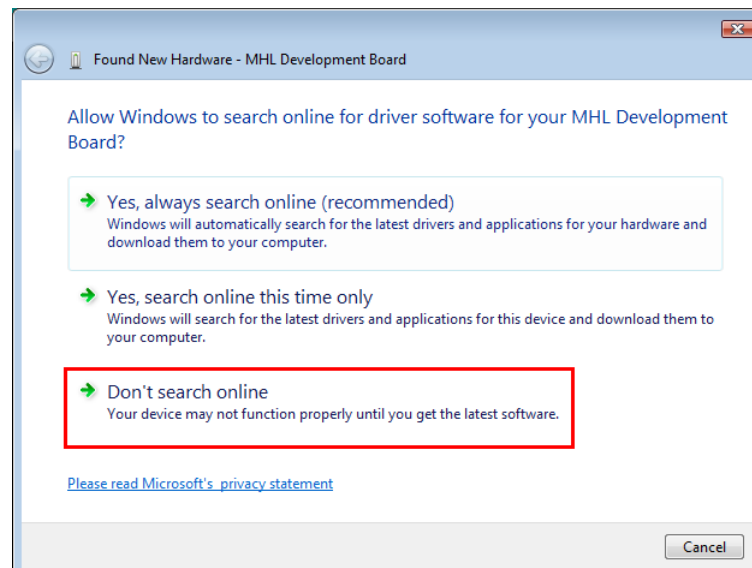
Μπορείτε να προχωρήσετε στην ενότητα "Λογισμικό".

Windows Vista™(x86)

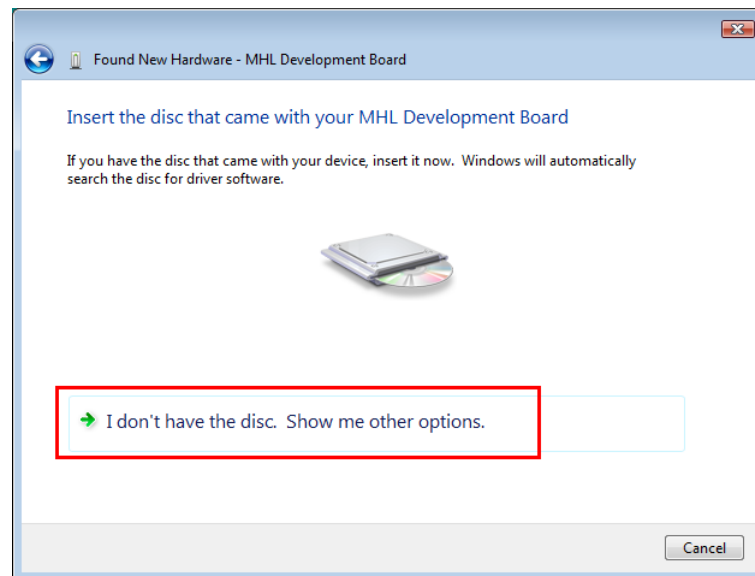
1. Συνδέστε τη συσκευή στο καλώδιο, και στη συνέχεια συνδέστε το καλώδιο σε μία κενή θύρα USB στον υπολόγιστή σας. Αφού ξεκινήσει ο οδηγός εγκατάστασης νέου υλικού, επιλέξτε "Εύρεση και εγκατάσταση οδηγών".



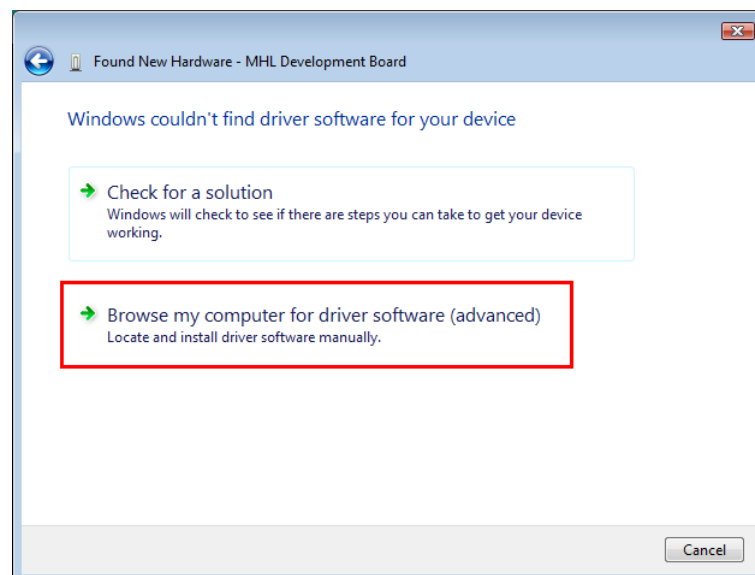
2. Στη συνέχεια, επιλέξτε "Να μην γίνει αναζήτηση".



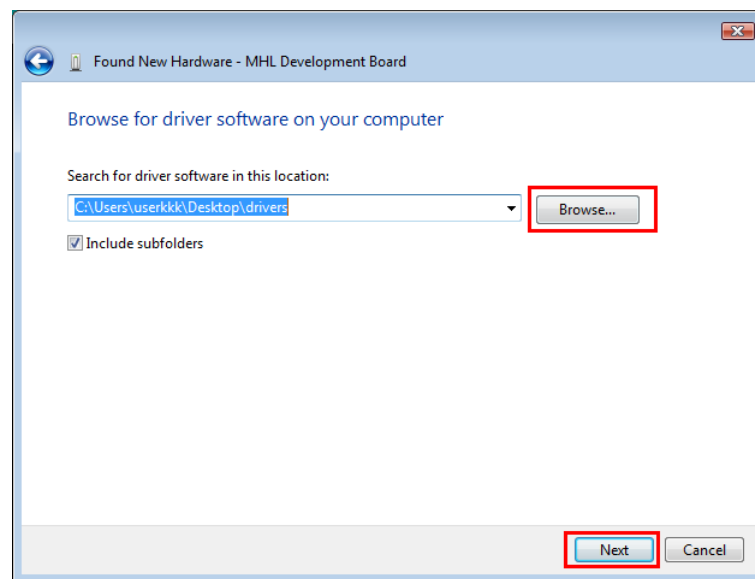
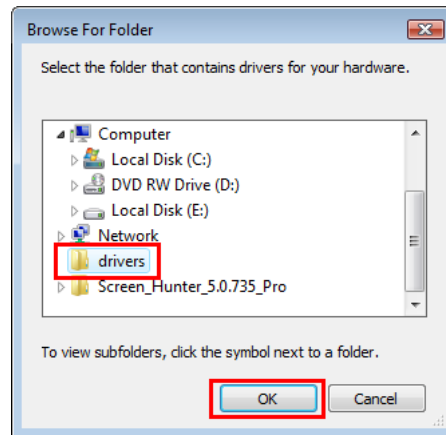
3. Στη συνέχεια, επιλέξτε "Δεν έχω δίσκο εγκατάστασης. Δείξε μου άλλες επιλογές".



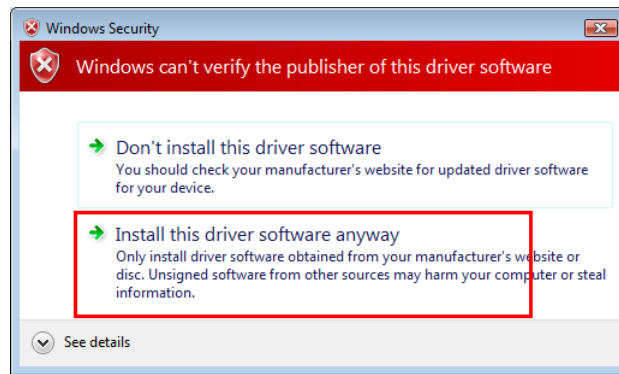
4. Στη συνέχεια, επιλέξτε "Αναζήτηση στον υπολογιστή μου για προγράμματα οδήγησης".



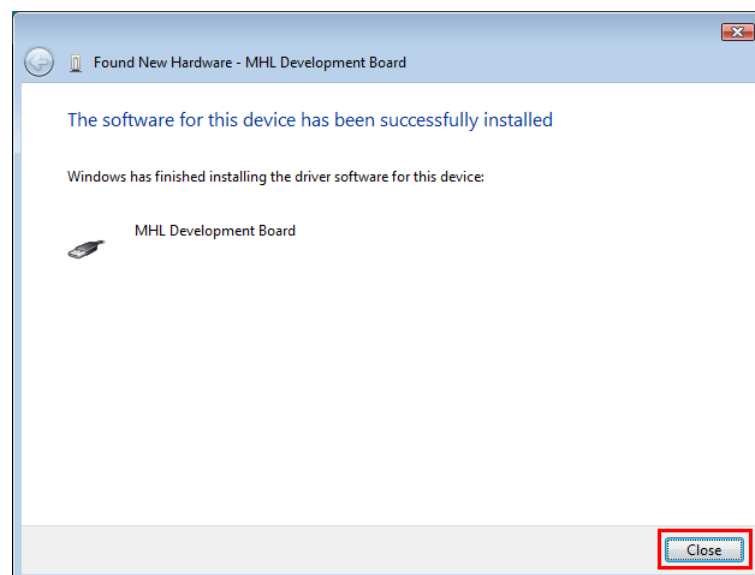
5. Επιλέξτε "Αναζήτηση" και περιηγηθείτε στο φάκελο που έχετε αποσυμπίσει τους οδηγούς, πατήστε "OK" και στη συνέχεια "Επόμενο".



6. Στο μήνυμα που θα εμφανιστεί επιλέξτε "Εγκατάσταση παρ'όλα αυτά".



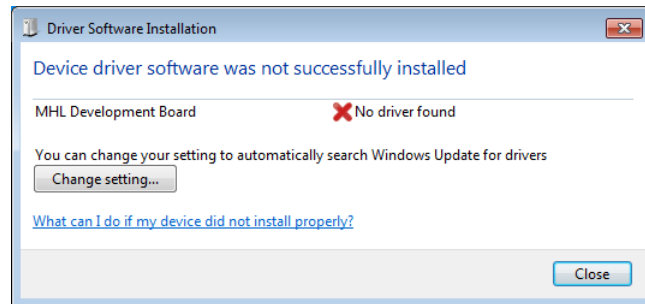
7. Το νέο υλικό, θα πρέπει να έχει εγκατασταθεί με επιτυχία.



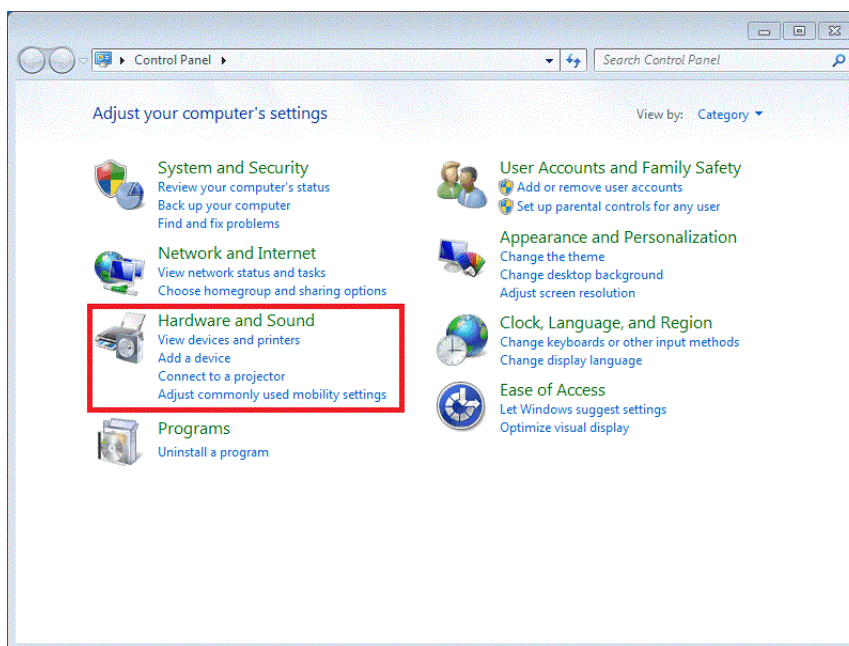
Μπορείτε να προχωρήσετε στην ενότητα "Λογισμικό".

Windows 7™(x86)

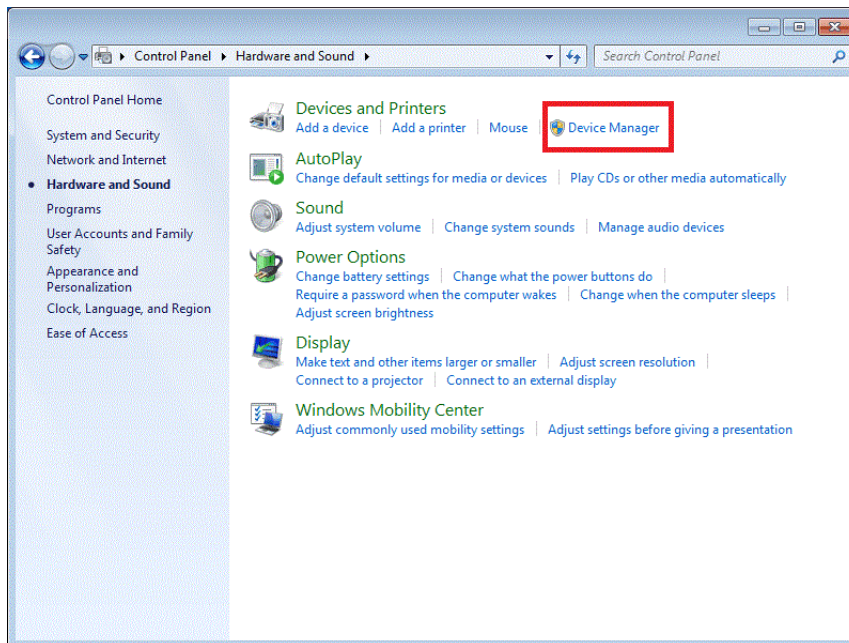
1. Συνδέστε τη συσκευή στο καλώδιο, και στη συνέχεια συνδέστε το καλώδιο σε μία κενή θύρα USB στον υπολογιστή σας. Το ακόλουθο μήνυμα θα εμφανιστεί. Πατήστε "Έξοδος".



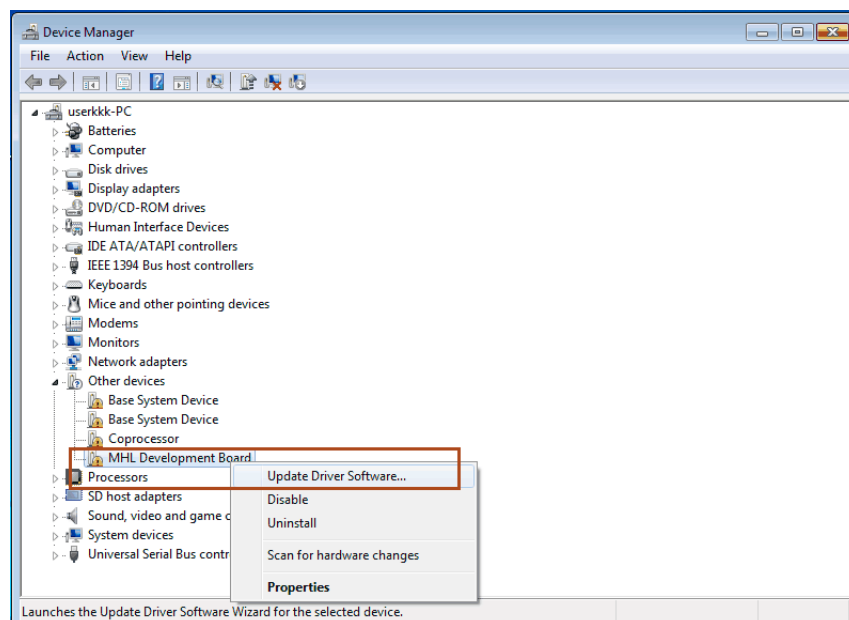
2. Πηγαίνετε στον "Πίνακα Ελέγχου" και επιλέξτε "Υλικό και Ήχος".



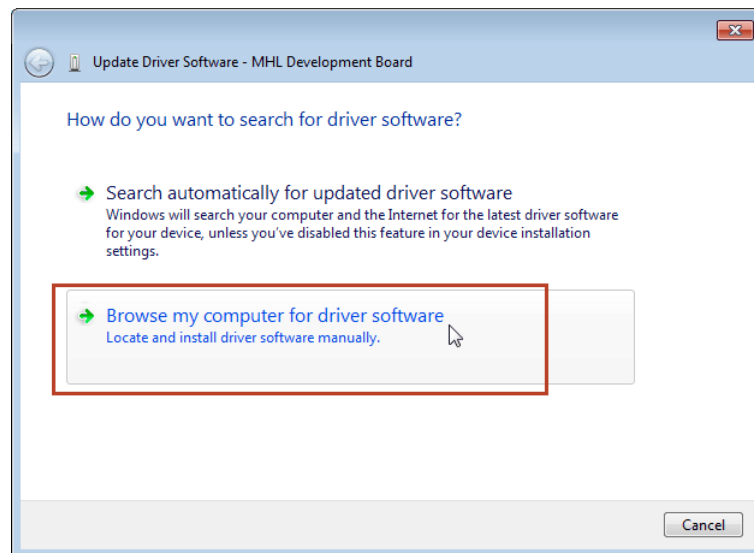
3. Στη συνέχεια, επιλέξτε ”Διαχείριση Συσκευών”.



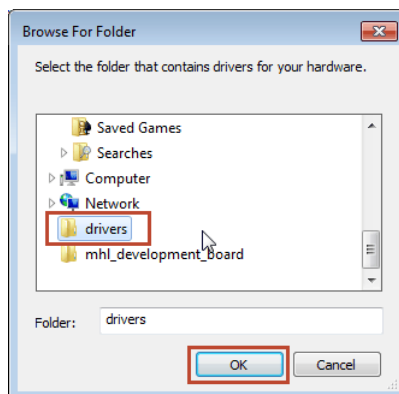
4. Βρείτε την συσκευή ”MHL Development Board”, κάντε δεξί κλικ και ”Ενημέρωση Προγραμμάτων Οδήγησης”.

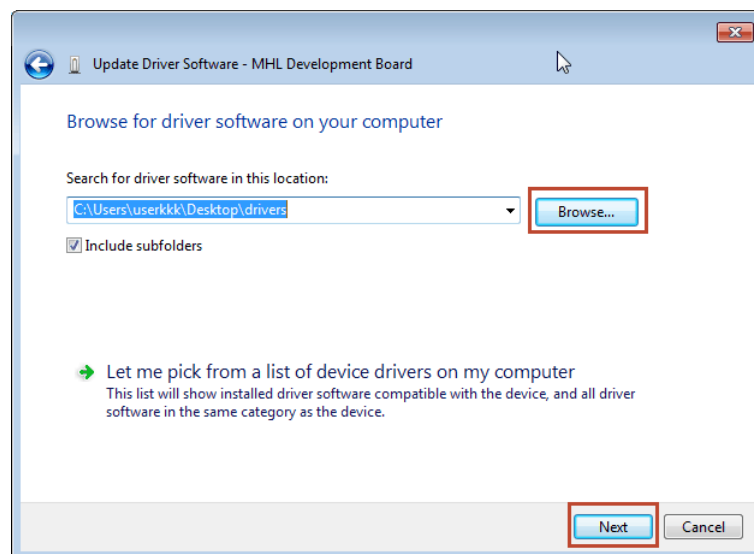


5. Στη συνέχεια, επιλέξτε "Αναζήτηση στον υπολογιστή μου για προγράμματα οδήγησης".

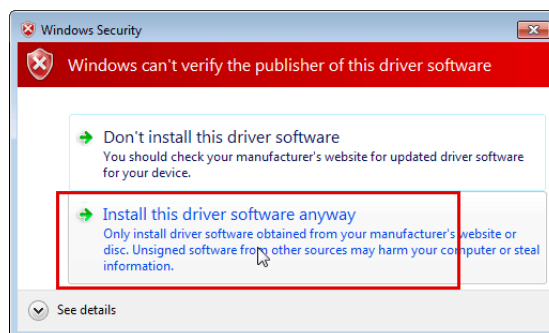


6. Επιλέξτε "Αναζήτηση" και περιηγηθείτε στο φάκελο που έχετε αποσυμπίε-σει τους οδηγούς, πατήστε "OK" και στη συνέχεια "Επόμενο".

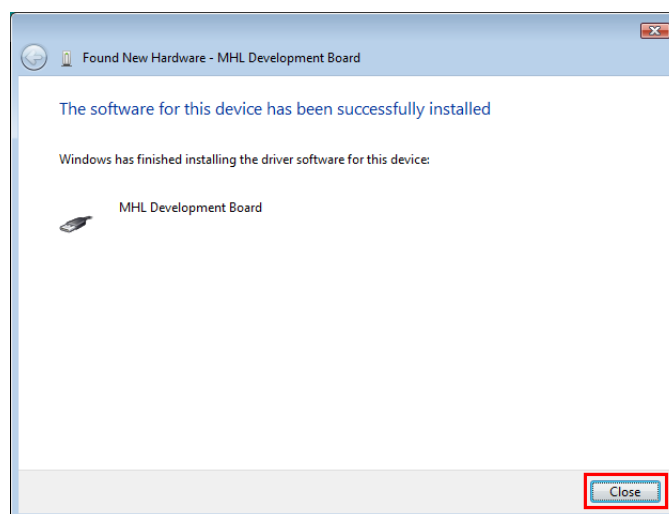




7. Στο μήνυμα που θα εμφανιστεί επιλέξτε "Εγκατάσταση παρ'όλα αυτά".



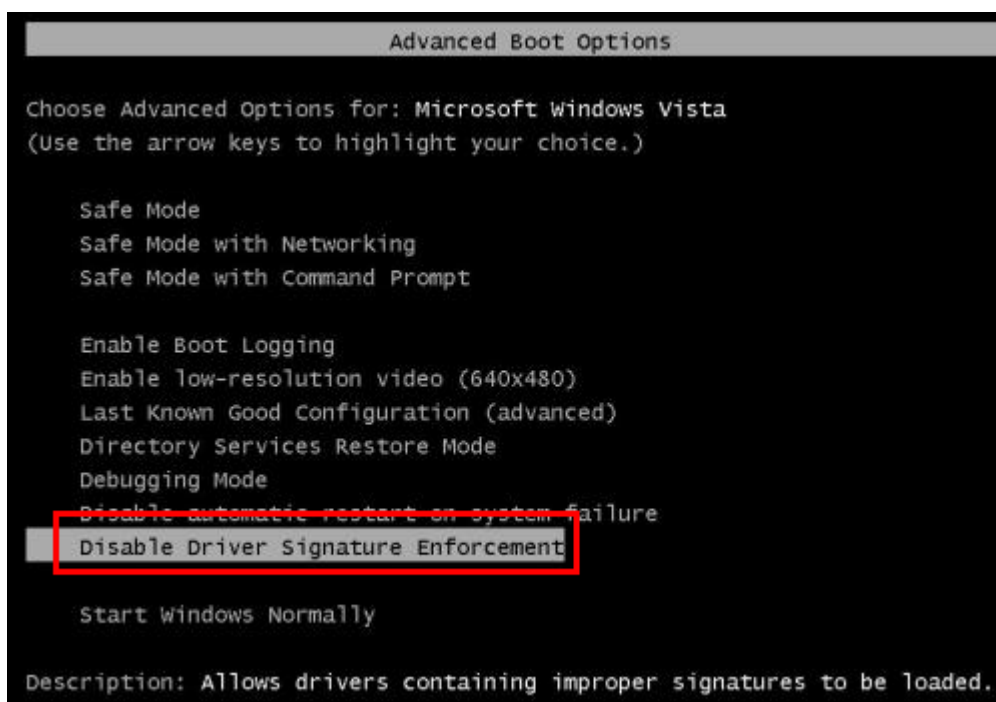
8. Το νέο υλικό, θα πρέπει να έχει εγκατασταθεί με επιτυχία.



Windows Vista™/7™(x64)

Για να εγκαταστήσετε επιτυχώς τα προγράμματα οδήγησης της συσκευής στα δύο 64-bit λειτουργικά θα πρέπει να ακολουθήσετε κατά γράμμα την παρακάτω διαδικασία.

Επανεκκινήστε τον υπολογιστή σας, πατώντας το πλήκτρο **"F8"** πριν ακριβώς εμφανιστεί το λογότυπο των Windows κατά την εκκίνηση του λειτουργικού. Τότε, η παρακάτω λίστα θα εμφανιστεί:



Επιλέξτε "Απενεργοποίηση ελέγχου της ψηφιακής υπογραφής των οδηγών" (ή "Disable driver signature enforcement"). Το λειτουργικό θα ξεκινήσει κανονικά. Τότε μπορείτε να εγκαταστήσετε την συσκευή όπως περιγράφεται στις ενότητες για την αντίστοιχη 32-bit έκδοση του λειτουργικού σας, παραπάνω.

Σημείωση: Αφού εγκαταστήσετε τους οδηγούς, για να χρησιμοποιείτε την συσκευή θα πρέπει πάντοτε να εκκινείτε το λειτουργικό με την επιλογή "Απενεργοποίηση ελέγχου της ψηφιακής υπογραφής των οδηγών" (ή "Disable driver signature enforcement") αλλιώς η συσκευή δεν θα αναγνωρίζεται από το λειτουργικό. Επίσης, **μην** προσπαθήσετε να χρησιμοποιήσετε άλλο software για να παρακάμψετε τον έλεγχο της ψηφιακής υπογραφής των drivers. Σε όλες μας τις δοκιμές με αντίστοιχα προγράμματα η συσκευή ή το λειτουργικό απέτυχαν να λειτουργήσουν σωστά.

Λογισμικό

Για την εκτέλεση της Γραφικής εφαρμογής χρειάζεστε τα εξής:

- Τη αναπτυξιακή πλακέτα "MHL Development Board", ορθά εγκατεστημένη σύμφωνα με το λειτουργικό σας σύστημα.
- Το Java Runtime Enviriment της Sun™, έκδοση 1.6 ή νεότερη (μπορείτε να το βρείτε εδώ: <http://www.java.com/en/download/manual.jsp>). Σημειώνουμε ότι για την λειτουργία της εφαρμογής απαιτείται η 32-bit έκδοση του JRE. Αν το λειτουργικό σας είναι αρχιτεκτονικής 64-bit και έχετε εγκατεστημένη την 64-bit έκδοση του JRE, θα πρέπει να εγκαταστήσετε και την 32-bit έκδοση.
- Το αρχείο εγκατάστασης της εφαρμογής "MHL Development Board Simulator".

Σημείωση: Προτού αρχίσετε τη διαδικασία εγκατάστασης, σιγουρευτείτε πως έχετε δικαιώματα διαχειριστή(administrator).

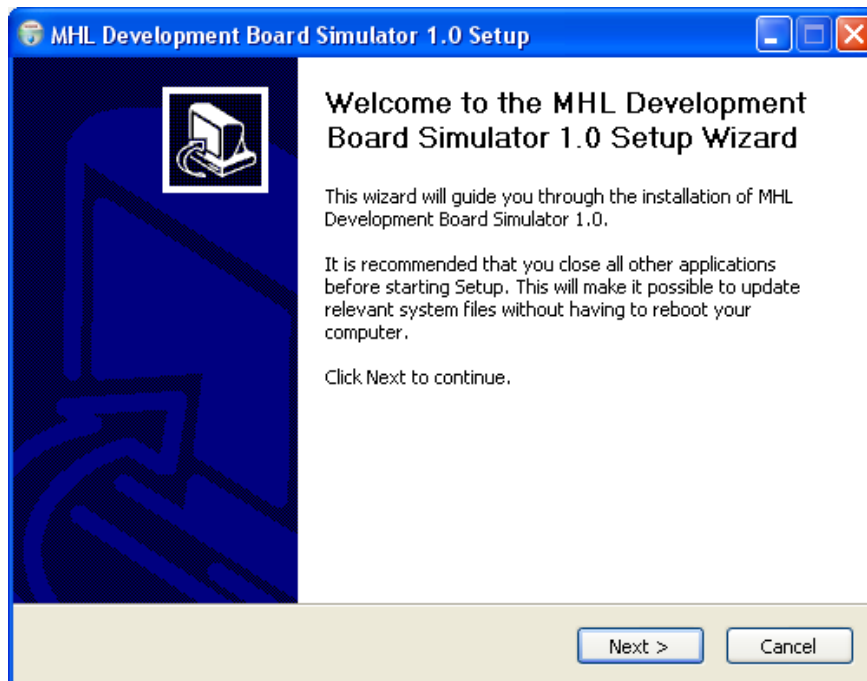
Εγκατάσταση λογισμικού

1. Αρχικά εγκαταστήστε το Java Runtime Enviriment της Sun™, ακολουθώντας τα βήματα της εγκατάστασης.
2. Στη συνέχεια κάντε διπλό κλικ στο αρχείο εγκατάστασης της εφαρμογής "MHL Development Board Simulator". Αν το λειτουργικό σας είναι Windows Vista/7 κάντε δεξί κλικ στο αρχείο, και "Εκτέλεση ως Διαχειριστής". Ο οδηγός εγκατάστασης θα ξεκινήσει. Ακολουθήστε τα βήματα παρακάτω.

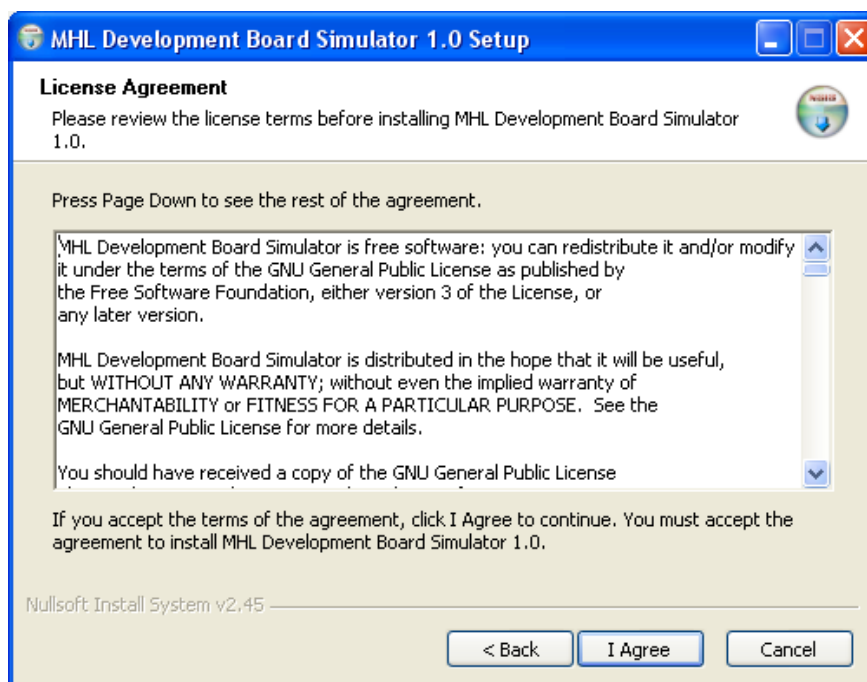
Επιλέξτε γλώσσα, και κάντε κλικ στο "OK".



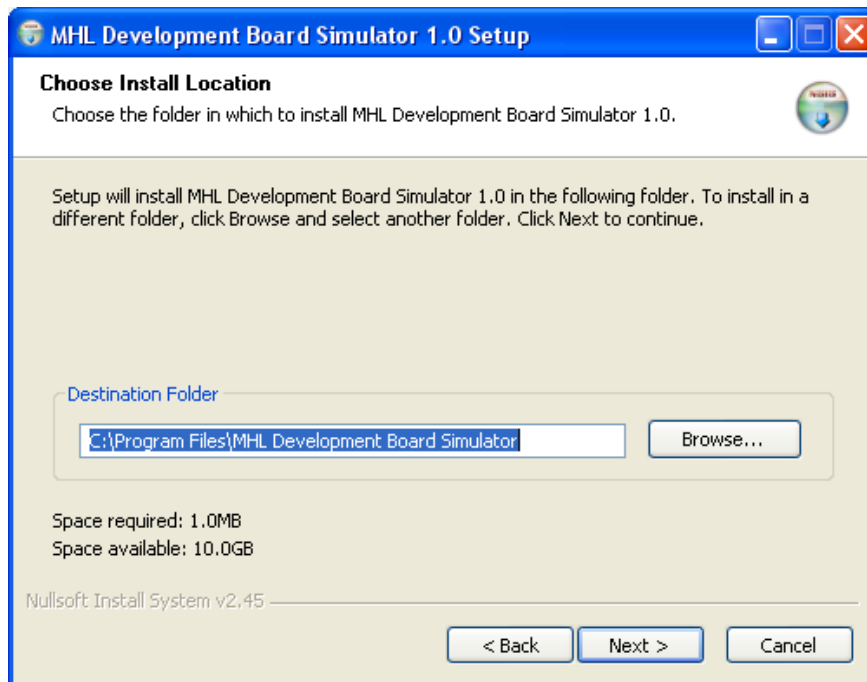
Κάντε κλικ στο "Επόμενο".



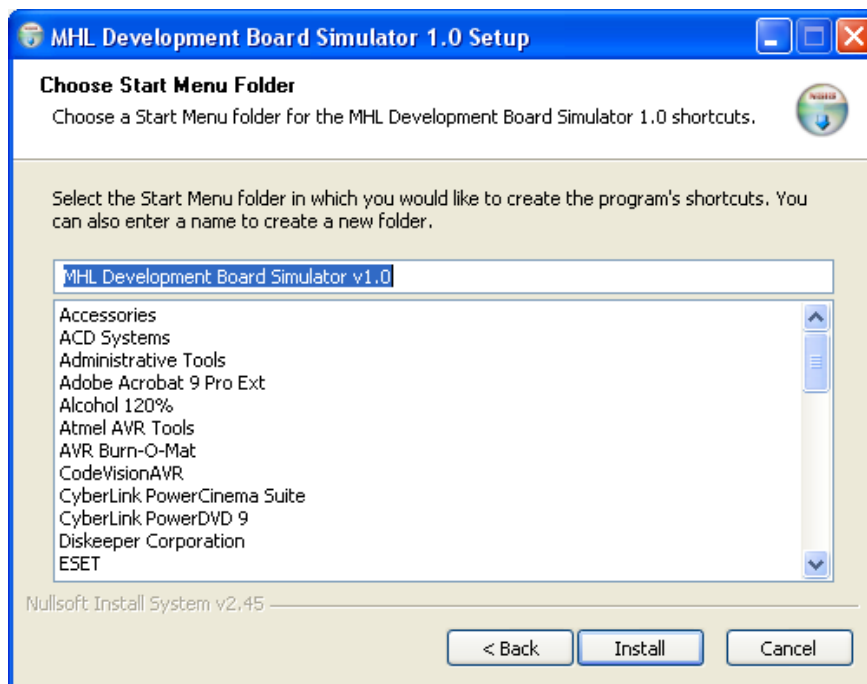
Επιλέξτε "Συμφωνώ".



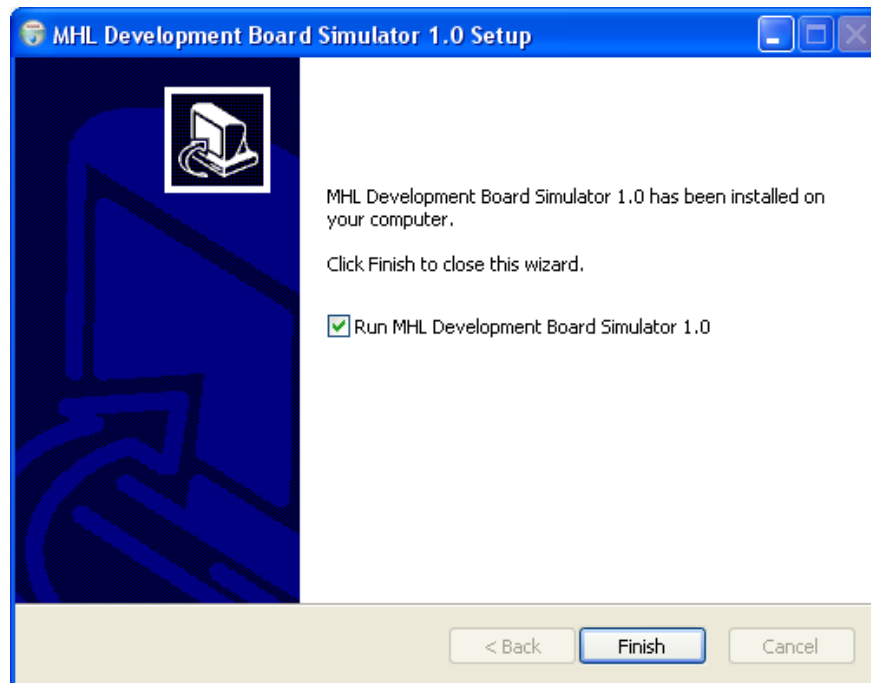
Διαλέξτε τη διαδρομή εγκατάστασης ή απλά αφήστε την εργοστασιακή και επιλέξτε "Επόμενο".



Επιλέξτε "Επόμενο".



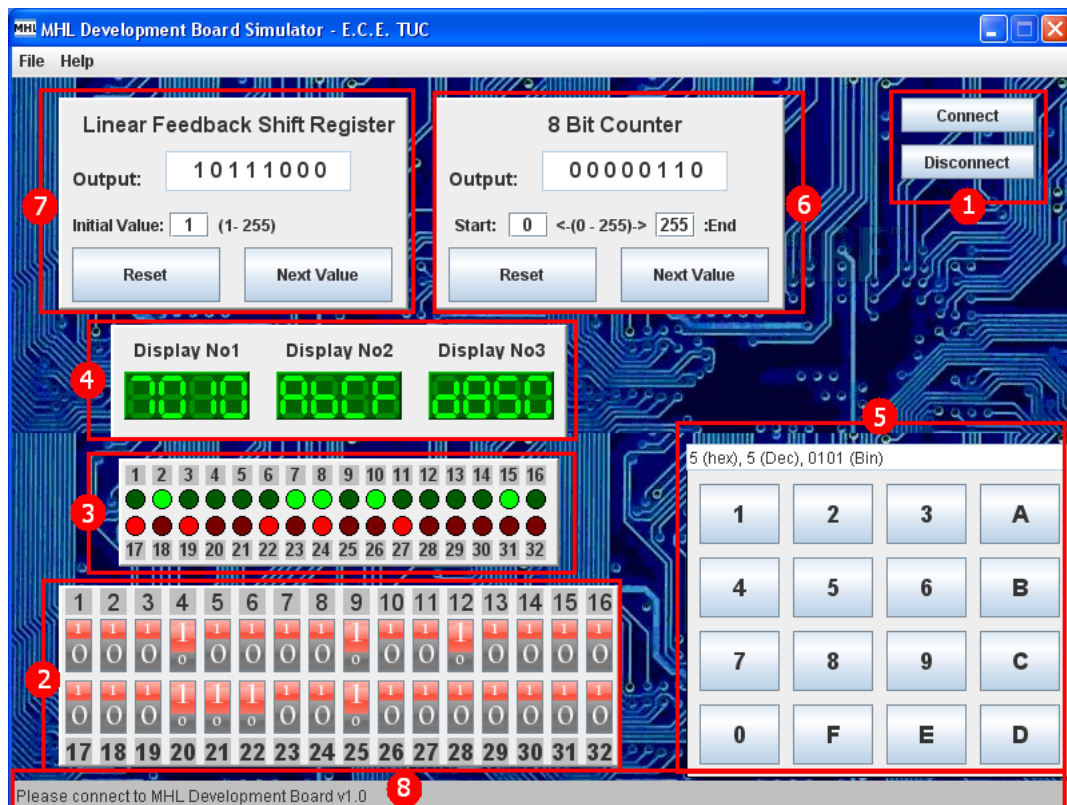
Επιλέξτε "Τέλος".



Το λογισμικό έχει εγκατασταθεί με επιτυχία και θα πρέπει να βλέπετε μια συντόμευση στην επιφάνεια εργασίας σας.

Χρήση λογισμικού

Στην παρακάτω εικόνα μπορούμε να δούμε τη γραφική διεπαφή. Πιο αναλυτικά, βλέπουμε:



1. τους διακόπτες Σύνδεσης και Αποσύνδεσης με την αναπτυξιακή πλακέτα.
2. τους S.P.D.T. Διακόπτες.
3. τις Φωτοδιόδους(LEDs).
4. τις Οθόνες 7 στοιχείων(7-Segment Displays).
5. το Δεκαεξαδικό Πληκτρολόγιο.
6. τον Μετρητή 8-bit(Counter).
7. τη Γεννήτρια Τυχαίων Αριθμών 8-bit(L.F.S.R.).

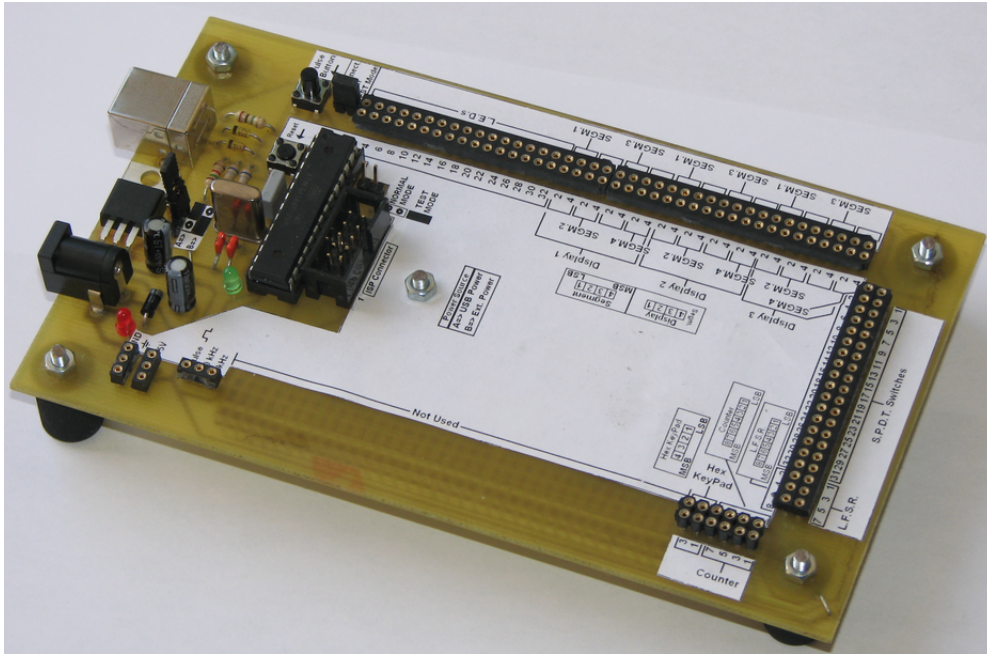
8. τη Μπάρα Ένδειξης Κατάστασης της συσκευής.

Η χρήση του λογισμικού είναι απλή. Αφού συνδέσετε την αναπτυξιακή πλακέτα στον υπολογιστή και εκκινήσετε το λογισμικό, για να συνδεθείτε με τη συσκευή, πατήστε το κουμπί σύνδεσης "Connect". Αντιστοίχως για να αποσυνδέσετε την συσκευή πατήστε το κουμπί "Disconnect". Καθ'όλη τη διάρκεια εργασίας στην γραφική εφαρμογή μπορούμε να παρακολουθούμε την κατάσταση σύνδεσης της συσκευής, στην μπάρα στο κάτω μέρος της οθόνης. Κάθε φορά που συνδέεστε στη συσκευή, όλοι οι είσοδοι και έξοδοι επανέρχονται στην αρχική τους κατάσταση(μηδενίζονται).

Για την αλλαγή των τιμών των Διακοπών, του Δεκαεξαδικού πληκτρολογίου, του Μετρητή και της Γεννήτριας τυχαίων αριθμών αρκεί να κάνετε κλικ στα αντίστοιχα κουμπιά. Για να μεταβάλετε τις αρχικές και τελικές τιμές του Μετρητή και της Γεννήτριας Τυχαίων Αριθμών πατήστε το κουμπί "Reset", συμπληρώστε την επιθυμητή νέα τιμή στα αντίστοιχα πεδία και την επόμενη φορά που θα πατήσετε το κουμπί "Next Value" ο Μετρητή ή η Γεννήτριας Τυχαίων Αριθμών θα δώσουν την αντίστοιχη τιμή.

Χρήση Αναπτυξιακής Πλακέτας

Η Αναπτυξιακή Πλακέτα φαίνεται παρακάτω:

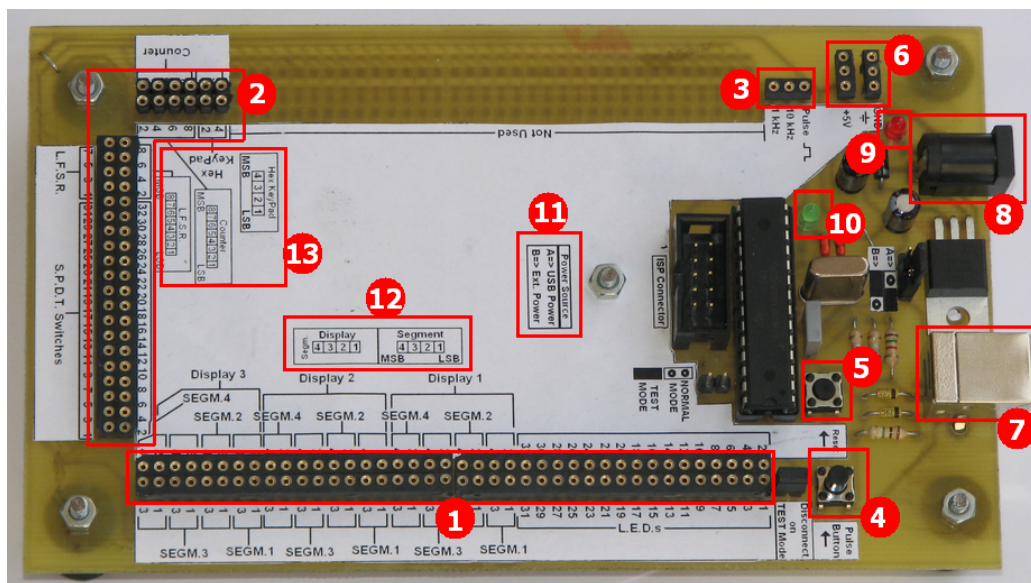


Αναλυτικότερα, στη διάθεση του χρήστη βρίσκονται:

- 80 Είσοδοι εκ των οποίων:
 - 32 Φωτοдиодοι(LEDs), στις οποίες αντιστοιχούν 32 ακροδέκτες.
 - 3 Οθόνες 7 στοιχείων(Seven Segments Displays) τεσσάρων ψηφίων έκαστη. Σε κάθε ψηφίο αντιστοιχούν 4 ακροδέκτες, οπότε συνολικά $3 \text{ οθόνες} \cdot 4 \text{ στοιχεία} \cdot 4 \text{ ακροδέκτες} = 48 \text{ ακροδέκτες}$.
- 52 Έξοδοι εκ των οποίων:
 - 32 Διακόπτες S.P.D.T.(Single Pole, Double Throw), στους οποίους αντιστοιχούν 32 ακροδέκτες.
 - ένα Δεκαεξαδικό πληκτρολόγιο, στο οποίο αντιστοιχούν 4 ακροδέκτες.
 - ένας Μετρητής 8-bits, στο οποίο αντιστοιχούν 8 ακροδέκτες.
 - μία Γεννήτρια Ψευδοτυχαίων Αριθμών 8-bits (Galois Linear Feedback Shift Register), στο οποίο αντιστοιχούν 8 ακροδέκτες.
- 2 ακροδέκτες για τις γεννήτριες συχνοτήτων, μία του 1Hz και μία των 10KHz.

- 1 ακροδέκτης για την γεννήτρια τετραγωνικών παλμών της τάξης των 5KHz, η οποία πυροδοτείται με το πάτημα ενός πιεστικού διακόπτη.
- 4 ακροδέκτες για σύνδεση σε σειριακή θύρα.
- 3 ακροδέκτες τάσης 5 Volt.
- 3 ακροδέκτες γείωσης.

Αφού είδαμε τις δυνατότητες της αναπτυξιακής πλακέτας, παρακάτω θα δούμε που βρίσκονται οι ακροδέκτες των παραπάνω μονάδων επάνω στην πλακέτα.

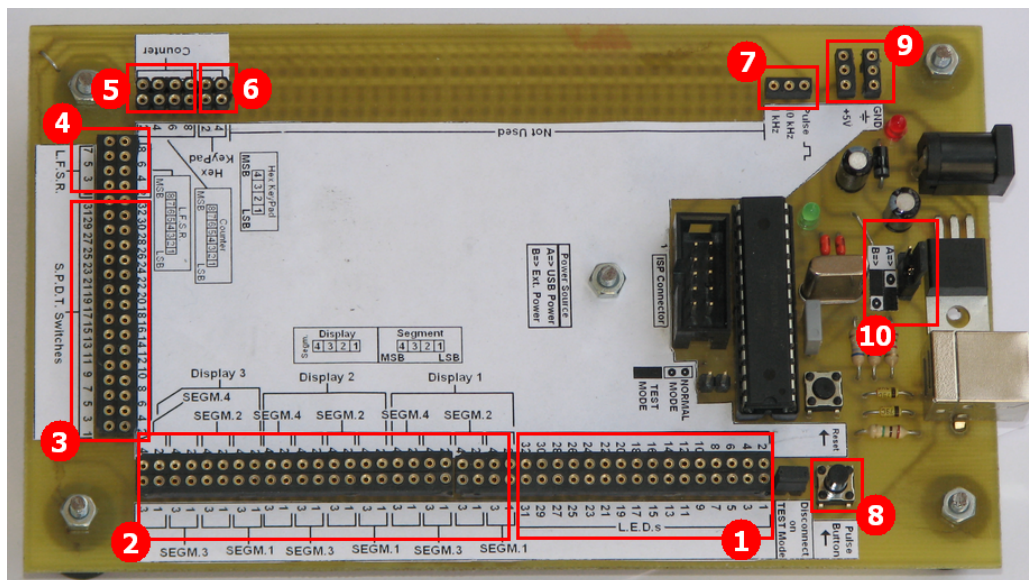


Έτσι, στην παραπάνω εικόνα βλέπουμε:

1. Τις εισόδους.
2. Τις εξόδους.
3. Τις γεννήτριες συχνотήτων και παλμών.
4. Τον πιεστικό διακόπτη της γεννήτριας παλμών.
5. Τον πιεστικό διακόπτη επανεκκίνησης(reset).
6. Τους ακροδέκτες τάσης +5 Volt και γείωσης.
7. Τη θύρα USB.
8. Τους ακροδέκτες τάσης +5 Volt και γείωσης.

9. Την υποδοχή εξωτερικής τροφοδοσίας.
10. Το ενδεικτικό LED τροφοδοσίας της πλακέτας.
11. Το ενδεικτικό LED USB δραστηριότητας.
12. Την ετικέτα πληροφοριών για την πηγή της τροφοδοσίας.
13. Την ετικέτα πληροφοριών για τις οθόνες και τα ψηφία τους.
14. Τις ετικέτες πληροφοριών για την γεννήτρια τυχαίων αριθμών, τον μετρητή και το δεκαεξαδικό πληκτρολόγιο,

Πιο αναλυτικά, βλέπουμε:



1. Τους ακροδέκτες εισόδων των LED.
2. Τους ακροδέκτες εισόδων των οθονών 7 στοιχείων.
3. Τους ακροδέκτες εξόδων των διακοπών.
4. Τους ακροδέκτες εξόδων της γεννήτρια τυχαίων αριθμών(LFSR).
5. Τους ακροδέκτες εξόδων του μετρητή.
6. Τους ακροδέκτες εξόδων του δεκαεξαδικού πληκτρολογίου.
7. Τους ακροδέκτες της γεννήτριας συχνότητας 1kHz, 10kHz και των παλμών(από αριστερά στα δεξιά)

8. Τον πιεστικό διακόπτη των τετραγωνικών παλμών.
9. Τους ακροδέκτες τάσης +5Volt και γείωσης GND.
10. Τον διακόπτη επιλογής πηγής τροφοδοσίας.

Πριν συνδέσετε την αναπτυξιακή πλακέτα στον υπολογιστή, καθορίστε τη θέση του διακόπτη τροφοδοσίας ώστε να τροφοδοτήσετε την πλακέτα σύμφωνα με την πηγή που επιθυμητέ(Εξωτερική τροφοδοσία, τροφοδοσία από την θύρα USB).

Αφού συνδέσετε την πλακέτα στον υπολογιστή, εκκινήστε το λογισμικό και πατήστε το κουμπί "Connect". Η πλακέτα θα πρέπει να έχει συνδεθεί στον υπολογιστή. Το LED στη θέση 10(δείτε στις εικόνες παραπάνω) θα πρέπει να αναβοσβήνει. Ωστόσο, μπορείτε να δείτε και τα μηνύματα στην μπάρα ένδειξης της κατάστασης της συσκευής. Σημειώνουμε ότι η συσκευή είναι έτοιμη για χρήση **μόνο** όταν το LED στη θέση 10 αναβοσβήνει και όχι όταν είναι είτε αναμμένο, είτε σβηστό.

Έπειτα, μπορείτε να συνδέσετε το κύκλωμά σας στις επιθυμητές μονάδες τις πλακέτας οποιαδήποτε στιγμή. Αυτό σημαίνει ότι δεν χρειάζεται να αποσυνδέσετε την πλακέτα από τον υπολογιστή, ούτε να διακόψετε την τροφοδοσία. Οι επεξηγηματικές ετικέτες επάνω στην πλακέτα θα σας βοηθήσουν να επιλέξετε τις εισόδους και εξόδους που επιθυμείτε να χρησιμοποιήσετε. Σημειώνουμε ότι οι ετικέτες είναι σχεδιασμένες σε αντιστοιχία με το λογισμικό στον υπολογιστή. Έτσι, για παράδειγμα, στην ετικέτα "Display", ο κάθε αριθμός "Segment" αναφέρετε στο αντίστοιχο ψηφίο της αντίστοιχης οθόνης στο λογισμικό.

Ακόμα, για να χρησιμοποιήσετε τετραγωνικούς ωρολογιακούς παλμούς στο κύκλωμά σας, συνδέστε τον ακροδέκτη παλμών στον αντίστοιχο ακροδέκτη του κυκλώματός σας και πατήστε διαδοχικά τον πιεστικό διακόπτη των παλμών. Με κάθε πάτημα ένα παλμός θα δημιουργείται στην έξοδο του ακροδέκτη.

Ιδιαίτερη **προσοχή** πρέπει να δοθεί στα καλώδια που χρησιμοποιείτε για τις συνδέσεις σας. Προσέξτε οι άκρες των καλωδίων που τοποθετούνται στους ακροδέκτες της πλακέτας να μην είναι τσακισμένες ή σπασμένες, γιατί αν κάποιο καλώδιο σπάσει μέσα στον ακροδέκτη τότε είναι σχεδόν αδύνατο να αφαιρεθεί με αποτέλεσμα την αχρήστευση του ακροδέκτη. Επίσης, κατά την τοποθέτηση των καλωδίων, **προσέξτε** να μην προκαλείτε βραχυκυκλώματα με άλλους ακροδέκτες ή με τα πιο ευαίσθητα εξαρτήματα της αναπτυξιακής πλακέτας.

Εγχειρίδιο Μηχανικού

Φυσική Διεπαφή

Η φυσική διεπαφή διαθέτει κάποια ευαίσθητα σημεία. Για να διασφαλίσουμε την πλήρη συμβατότητα της συσκευής με κάθε USB controller σε οποιονδήποτε υπολογιστή, μεγάλη σημασία και προσοχή πρέπει να δοθεί στις δύο διόδους Zener που χρησιμοποιούνται στις γραμμές D+ D- της USB. Οι διόδοι πρέπει να είναι 3.6V χαμηλής κατανάλωσης (Low power, 500mW) ώστε να μην προκαλούν μεγάλη παρασιτική χωρητικότητα και κατ'επέκταση μεγάλη παραμόρφωση στις γραμμές δεδομένων. Συνίσταται να χρησιμοποιούνται διόδοι τύπου 1N747A ή παρόμοιες με τα ίδια ηλεκτρικά χαρακτηριστικά. Ακόμα ένας κρίσιμος παράγοντας για την ορθή ζεύξη της συσκευής στον δίαυλο αποτελεί το ρολόι χρονισμού του AVR. Σε περίπτωση εσφαλμένης ζεύξης, αφού ελεγχθούν οι τάσεις στα άκρα των διόδων, καλή σχεδιαστική παρέμβαση αποτελεί η αντικατάσταση του κρυστάλλου των 16MHz με έναν των 12MHz. **Προσοχή:** σε αυτή την περίπτωση πρέπει να υπολογιστούν ξανά και να αλλαχθούν οι τιμές των καταχωρητών του μικροελεγκτή από τις οποίες εξαρτώνται οι γεννήτριες συχνοτήτων. Ακόμα, το firmware του AVR πρέπει να επαναδομηθεί (re-compile) σύμφωνα με το νέο ρολόι.

Πολύ σημαντικό παράγοντα για την ορθή λειτουργία του μικροελεγκτή στο κύκλωμα αποτελούν οι τιμές των "ασφαλειών" (Fuses) του AVR. Μετά τον προγραμματισμό του μικροελεγκτή **πρέπει** να ακολουθήσει και ο προγραμματισμός των Fuses.

Όσον αφορά τη συμβατότητα της συσκευής με τα προγράμματα οδήγησης του εκάστοτε λειτουργικού, αξίζει να αναφέρουμε ότι η αναπτυξιακή πλακέτα συνεργάζεται απόλυτα με το πακέτο οδηγών LibUSB έκδοσης 0.1.xx.x. Οι εκδόσεις του πακέτου 1.x ή μεταγενέστερες, κατά πάσα πιθανότητα, είναι συμβατές με την συσκευή, ωστόσο δεν υποστηρίζονται από το JAVA LibUSB wrapper της γραφικής διεπαφής.

Ακόμα, αξίζει να αναφέρουμε πως τα ολοκληρωμένα PCA9506DGG δύνανται να αντικατασταθούν με τα PCA9505DGG, ωστόσο στη περίπτωση αυτή θα πρέπει να γίνουν ριζικές αλλαγές στο firmware του AVR.

Τέλος, οποιαδήποτε τροποποίηση του τυπωμένου κυκλώματος θα πρέπει να γίνει με τα εργαλεία Altium P-CAD 2006 και Cadence SPECCTRA.

Λειτουργική Διεπαφή

Η λειτουργική διεπαφή δεν περικλείει, γενικά, κινδύνους ή ευαίσθητα σημεία. Αξίζει να αναφέρουμε ότι το περιβάλλον ανάπτυξης της εφαρμογής είναι το Eclipse SDK, έκδοσης 3.4.2. Ο μηχανικός μπορεί να βρει ιδιαίτερη βοήθεια,

κατά την επέκταση της γραφικής εφαρμογής, κάνοντας χρήση του γραφικού εργαλείου Visual Editor Eclipse Plug-In, έκδοσης 1.4.0. Προσοχή πρέπει να δοθεί στο API της JAVA που χρησιμοποιήθηκε. Η ορθή λειτουργία της εφαρμογής είναι γνωστό ότι επιτυγχάνεται με το πακέτο JRE 1.6(ή ανώτερο) της SUN™.

Ακόμα, το εργαλείο που χρησιμοποιήθηκε για την δημιουργία του εκτελέσιμου Jar αρχείου είναι το Fat Jar Eclipse Plug-In 0.0.31 και για την δημιουργία του Win-32 wrapper(mhl_dev_board_sim_v10.exe) είναι το JSmooth 0.9. Τέλος, το εκτελέσιμο αρχείο εγκατάστασης της εφαρμογής δημιουργήθηκε με το εργαλείο NSIS 2.45.

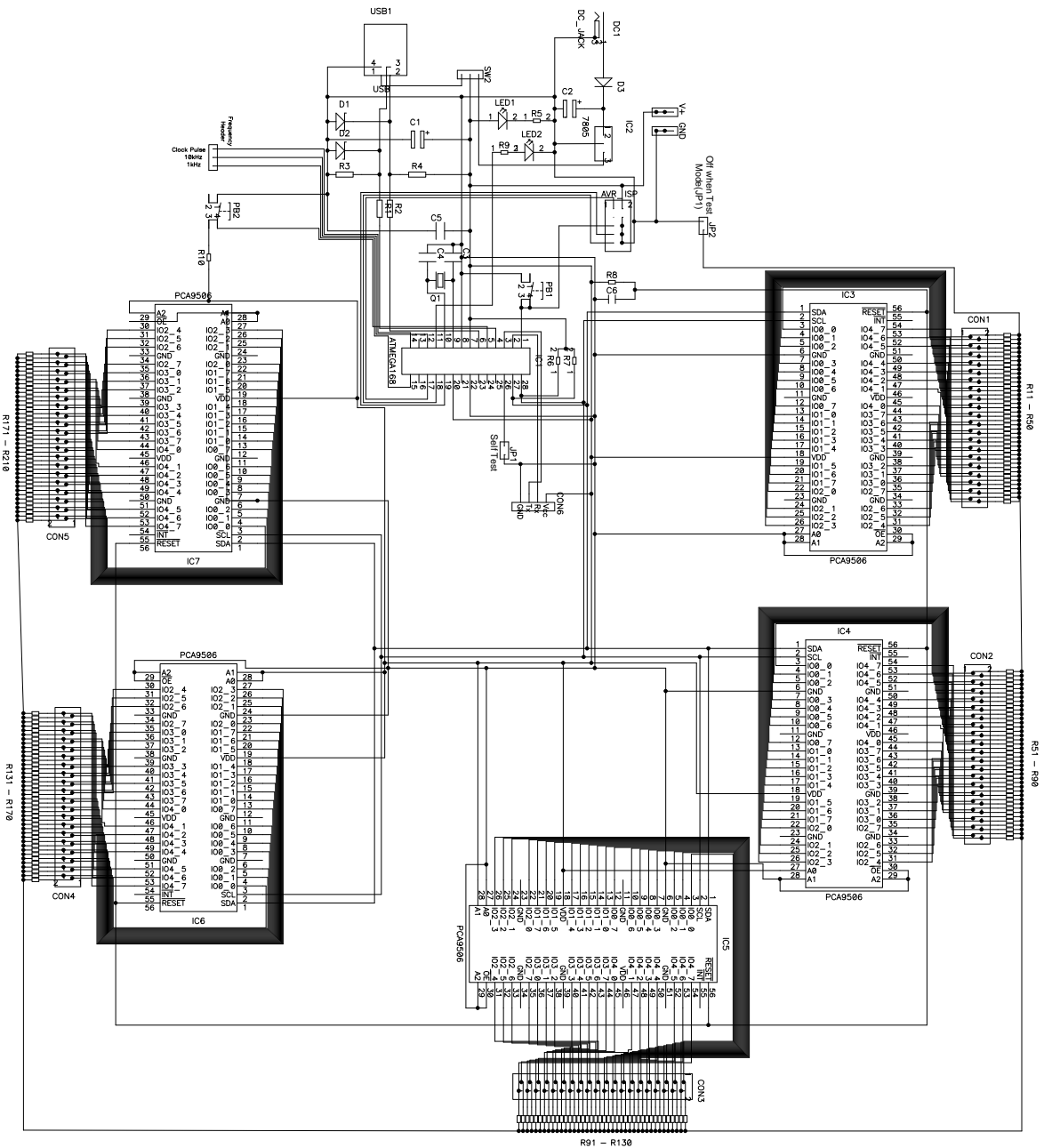
Δοκιμαστική Πλακέτα Ελέγχου

Η χρήση της Δοκιμαστικής Πλακέτας Ελέγχου είναι αρκετά εύκολη και απλή. Αφού προσαρμόσουμε τη δοκιμαστική πλακέτα επάνω στην αναπτυξιακή, αρκεί να κλείσουμε το διακόπτη δοκιμαστικής λειτουργίας και να τροφοδοτήσουμε την αναπτυξιακή πλακέτα με ρεύμα. Κατά την δοκιμαστική λειτουργία όλοι οι ακροδέκτες εισόδων και εξόδων μετατρέπονται σε εξόδους. Αρχικά, και για μερικά δευτερόλεπτα, σε όλες τις εξόδους θα εφαρμοστεί υψηλό δυναμικό και τα LEDs της δοκιμαστικής πλακέτας θα φωτοβολήσουν. Στη συνέχεια, τα LEDs θα σβήσουν και θα αρχίσουν να αναβοσβήνουν διαδοχικά, έτσι ώστε να δοκιμαστούν οι ακροδέκτες και τα ολοκληρωμένα I²C to GPIO ένα προς ένα, εξονυχιστικά.

Παράρτημα Β

Τεχνικά Αρχεία

Σχηματικό Διάγραμμα(Schematic)

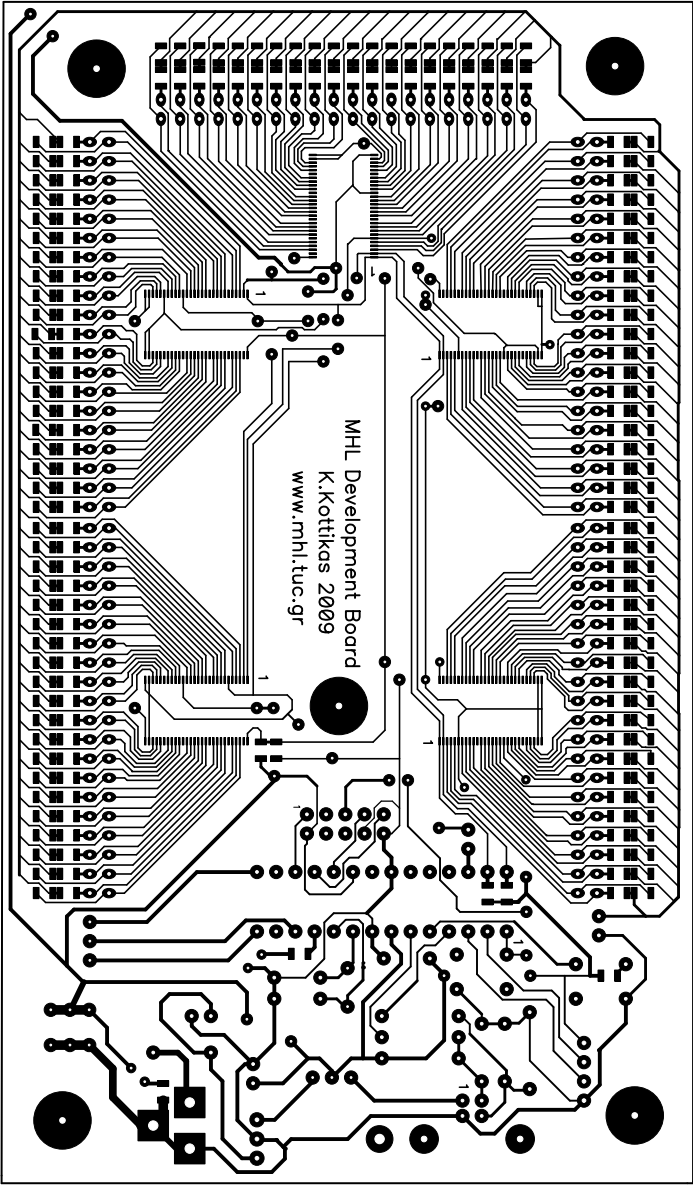


Title	
MHL Development Board	

Size	Number	Rev
A2	Kotikios Konstantinos	1.00

Date	Filename	Drawn by	Sheet	of
27/09/2009	mhl_dev_board.sch			

Τυπωμένο Κύκλωμα(PCB)



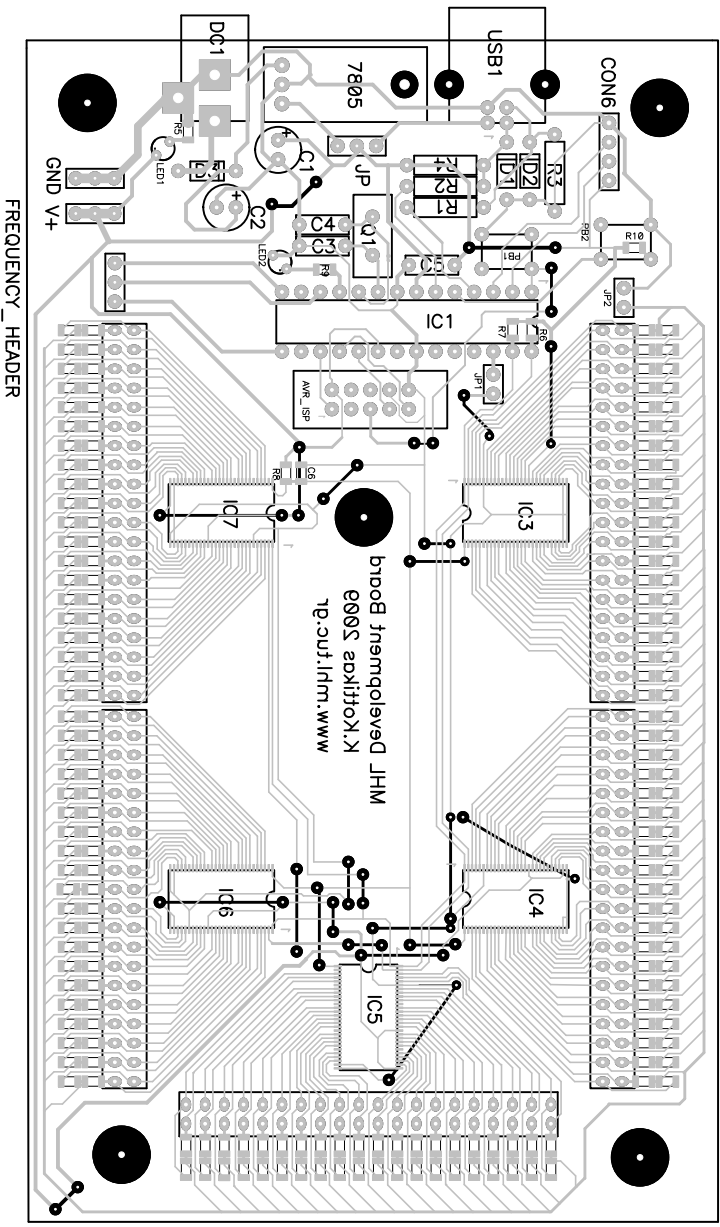
Λίστα εξαρτημάτων

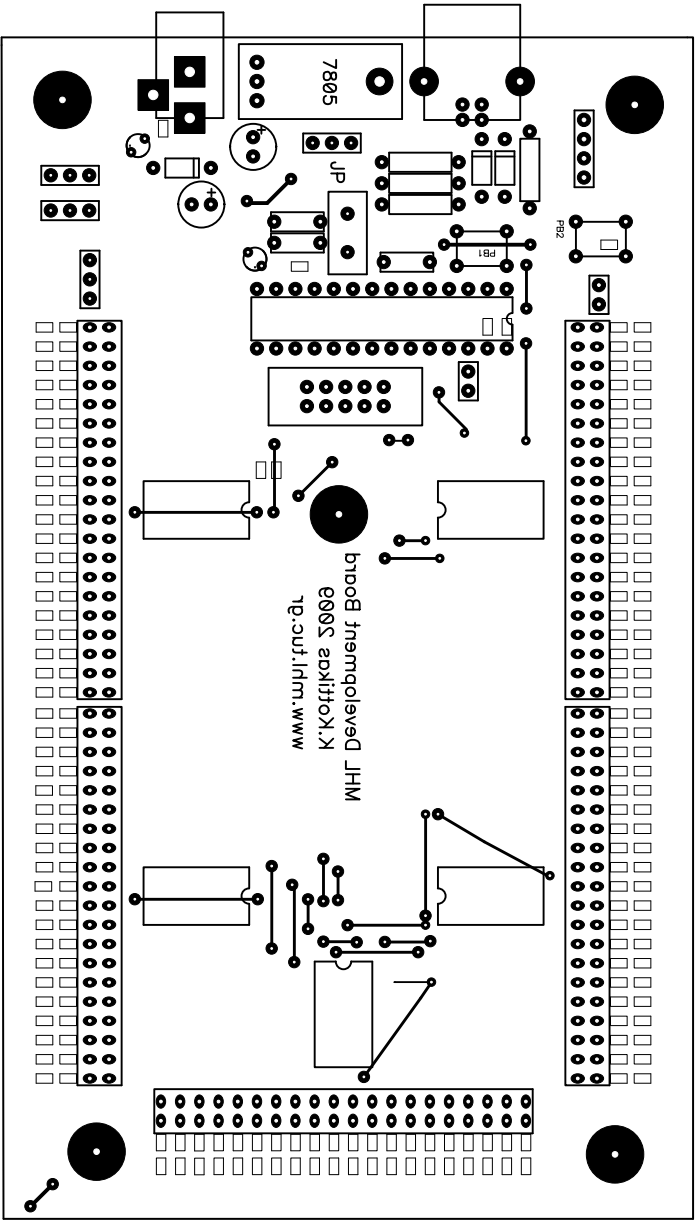
PART LIST

MHL Development Board

- DC1 = 1 x DC in Female 2.1mm
- USB1 = 1 x USB Female Type B
- 7805 = 1 x 7805 Regulator
- D3 = 1 x 1N4001 Diode
- LED2 = 1 x GREEN LED 3mm
- LED1 = 1 x RED LED 3mm
- Q1 = 1 x 16MHz Crystal
- AVR_ISP = 1 x 10 pin Male ISP Connector(Rectangular)
- C1,C2 = 2 x 10uF 16-25V Electrolitic Capacitor
- C3,C3 = 2 x 22pF Capacitor
- C5 = 1 x 100nF Capacitor
- D1,D2 = 2 x 3.6V Zenner(Low power,500mW,like 1N747A)
- R1,R2 = 2 x 68 Ohms Resistor
- R4 = 1 x 1.5 kOhms Resistor
- R3 = 1 x 1 MOhms Resistor
- R6,R7,R8,R10,INPUTS = 84 x 4.7 kOhms Resistor SMD 0805
- R5,R9 = 2 x 330 Ohms Resistor SMD 0805 (or 330 to 390)
- C6 = 1 x 88nF Capacitor SMD 0805(or 100nF)
- PB2 = 1 x Push Button(Long neck)
- PB1 = 1 x Push Button(short neck)
- IN/OUTputs = 170 x Header connector pins(Female,Golden)
- CON 6 = 4 x Header connectors Male(or other 4 pin Connector)
- JP,JP1,JP2 = 7 x Jumper pins Male
- 3 x Jumper cups
- 5 x Board holding screws
- 2 x 14 pin IC-DIP socket
- ICs-----
- IC1 = 1 x Atmel ATMEGA168-20PU(or AtMega48-20PU/88-20PU/48p-20PU/88p-20PU/168p-20PU)
- IC3,IC4,IC5,IC6(IC7 for expansion) = 4 x NXP PCA9506DCG, socket TSSOP56(PCA9505DCG are NOT compatible with the firmware)

Τοποθέτηση Εξαρτημάτων





Βιβλιογραφία

- [1] Arkansas Technical University. Digital Logic Design. Website. http://engr.atu.edu/Greco/Digital_Logic/diglogic.html.
- [2] Atmel. In-System Programming. Website. http://www.atmel.com/dyn/resources/prod_documents/DOC0943.PDF.
- [3] Norman Balabanian and Bradley Carlson. *Digital Logic Design Principles*. John Wiley and Sons Ltd, 2001.
- [4] Center for Reliable and High-Performance Computing. Logical Design. Website. <http://www.crhc.uiuc.edu/~nhv/462/index.html>.
- [5] Christian Starkjohann. V-USB. Website. <http://www.obdev.at/products/vusb/index.html>.
- [6] Engineering and Computer Science Program. Digital Logic Design. Website. <http://www.engracs.com/courses/engr250/engr250.php>.
- [7] David Hodges and Horace Jackson. *Analysis and Design of Digital Integrated Circuits*. McGraw-Hill Education – Europe, 1988.
- [8] Paul Horowitz and Winfield Hill. *The Art of Electronics*. Cambridge University Press, 1989.
- [9] Indian Institute of Technology. Computer Science and Engineering. Website. <http://www.cse.iitm.ac.in/moodle/course/view.php?id=44>.
- [10] Martin Kenneth. *Digital Integrated Circuit Design*. Oxford University Press, 1999.
- [11] Edward McCluskey. *Logic Design Principles*. Prentice-Hall Publishers, 1988.
- [12] Open Source. LibUSB. Website. <http://libusb.org/>.

- [13] Rose-Hulman Institute of Technology. Digital Design. Website. <http://www.rose-hulman.edu/class/ee/radu/ece330/>.
- [14] University of Alberta. Introduction to Digital Electronics. Website. <http://www.ece.ualberta.ca/~ee280/lab/>.
- [15] University of Applied Sciences of Technology NTB. JAVA LibUSB Wrapper. Website. <http://libusbjava.sourceforge.net/>.
- [16] University of Fatih. Digital Logic Design. Website. <http://www.fatih.edu.tr/~edoganay/Courses2.html>.
- [17] University of Kentucky. Logical Design. Website. <http://courses.engr.uky.edu/spring06/ECE/EE281/>, <http://courses.engr.uky.edu/spring07/ECE/ee281/>.
- [18] University of the South Pacific. Digital Electronics. Website. <http://classshares.student.usp.ac.fj/EE222/Old%20-%202006%20Files/Labs/index.htm>.
- [19] University of Wisconsin Madison. Digital Logic. Website. <http://eceserv0.ece.wisc.edu/~morrow/ECE351/>.
- [20] John Wakerly. *Digital Design Principles and Practices*. 2005.
- [21] Michael Wiesner and Vance Venable. *Digital Electronics Laboratory Manual 6th Edition*. Prentice Hall, 2002.
- [22] Wikipedia. Atmel AVR. Website. http://en.wikipedia.org/wiki/Atmel_AVR.
- [23] Wikipedia. General Purpose Input/Output. Website. http://en.wikipedia.org/wiki/General_Purpose_Input/Output.
- [24] Wikipedia. Hot Swap. Website. http://en.wikipedia.org/wiki/Hot_swapping.
- [25] Wikipedia. I^2C . Website. <http://en.wikipedia.org/wiki/I2C>.
- [26] Wikipedia. Linear Feedback Shift Register. Website. http://en.wikipedia.org/wiki/Linear_feedback_shift_register.
- [27] Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών Εθνικό Μετσόβιο Πολυτεχνείο. Εργαστήρια Ψηφιακών Συστημάτων. Website. <http://archimedes.cs.ntua.gr/askhseis.html>.
- [28] Τμήμα Πληροφοριακών και Επικοινωνιακών Συστημάτων Πανεπιστήμιο Αγαίου. Εργαστήρια Λογικής Σχεδίασης Ψηφιακών Συστημάτων. Website. http://www.icsd.aegean.gr/karybali/LD_Lab/LD-Lab.html.

- [29] Τμήμα Πληροφορικής και Υπολογιστών Πανεπιστήμιο Αθηνών. Εργαστήρια Λογικής Σχεδίασης. Website. <http://eclass.di.uoa.gr/D13/>.
- [30] Τμήμα Επιστήμης και Τεχνολογίας Τηλεπικοινωνιών Πανεπιστήμιο Πειραιά. Εργαστήρια Λογικής Σχεδίασης Ψηφιακών Συστημάτων. Website. <http://gunet.cs.unipi.gr/TMA110/>.
- [31] Τμήμα Τεχνολογίας και Υπολογιστών Πανεπιστήμιο Πελοποννήσου. Εργαστήρια Λογικής Σχεδίασης. Website. <http://eclass.uop.gr/courses/TST188/>.