



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ ΒΙΟΪΑΤΡΙΚΗ

Σχεδίαση και Ανάπτυξη
Αυτόνομου Κυκλώματος Αισθητήρων

Μυριδάκης Πάυλος

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
Επιβλέπων
Αναπληρωτής Καθηγητής Κακαρούντας Αθανάσιος

Λαμία, 2020



UNIVERSITY OF THESSALY

SCHOOL OF SCIENCE

INFORMATICS AND COMPUTATIONAL BIOMEDICINE

**Designing and Development
Autonomous Sensor Circuit**

Myridakis Paul

Master Thesis

**Supervisor
Associate Professor Kakarountas Athanasios**

Lamia

2020



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΔΙΑΤΜΗΜΑΤΙΚΟ ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΗ
ΒΙΟΪΑΤΡΙΚΗ
ΚΑΤΕΥΘΥΝΣΗ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΩΝ ΚΑΙ
ΕΠΙΚΟΙΝΩΝΙΩΝ (Τ.Π.Ε.) ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ**

**«ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑ ΠΛΗΡΟΦΟΡΙΩΝ ΚΑΙ
ΕΠΙΚΟΙΝΩΝΙΩΝ (Τ.Π.Ε.) ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ»**

**Σχεδίαση και Ανάπτυξη
Αυτόνομου Κυκλώματος Αισθητήρων**

Μυριδάκης Πάυλος

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
Επιβλέπων
Αναπληρωτής Καθηγητής Κακαρούντας Αθανάσιος**

Λαμία, 2020

«Υπεύθυνη Δήλωση μη λογοκλοπής και ανάληψης προσωπικής ευθύνης»

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων και γνωρίζοντας τις συνέπειες της λογοκλοπής, δηλώνω υπεύθυνα και ενυπογράφως ότι η παρούσα εργασία με τίτλο «Σχεδίαση και Ανάπτυξη Αυτόνομου Κυκλώματος Αισθητήρων» αποτελεί προϊόν αυστηρά προσωπικής εργασίας και όλες οι πηγές από τις οποίες χρησιμοποίησα δεδομένα, ιδέες, φράσεις, προτάσεις ή λέξεις, είτε επακριβώς (όπως υπάρχουν στο πρωτότυπο ή μεταφρασμένες) είτε με παράφραση, έχουν δηλωθεί κατάλληλα και ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο ΔΗΛΩΝ
Μυριδάκης Παύλος

Σεπτέμβριος 2020

Υπογραφή



**Σχεδίαση και Ανάπτυξη
Αυτόνομου Κυκλώματος Αισθητήρων**

Μυριδάκης Πάυλος

Τριμελής Επιτροπή:

Αναπληρωτής Καθηγητής Κακαρούντας Αθανάσιος

Επίκουρος Καθηγητής Λουκόπουλος Αθανάσιος

Επίκουρη Καθηγήτρια Κοζύρη Μαρία

Επιστημονικός Σύμβουλος:

Αναπληρωτής Καθηγητής Κακαρούντας Αθανάσιος

ΕΥΧΑΡΙΣΤΙΕΣ

Η ολοκλήρωση της διπλωματικής εργασίας υλοποιήθηκε με την υποστήριξη ενός συνόλου ανθρώπων, στους οποίους θα ήθελα να εκφράσω τις θερμότερες ευχαριστίες μου. Πρώτα από όλους θα ήθελα να ευχαριστήσω τον καθηγητή και επιβλέποντά μου, Αναπληρωτή Καθηγητή του τμήματος Πληροφορικής με Εφαρμογές στη Βιοϊατρική του Πανεπιστημίου Θεσσαλίας κ. Κακαρούντα Αθανάσιο, για τις γενικές υποδείξεις, τη συνεχή παρακολούθηση, επίβλεψη, άμεση υποστήριξη και διόρθωση, ώστε να καταστεί εφικτή η παρούσα διπλωματική εργασία. Νιώθω επίσης την ανάγκη να επισημάνω ότι η αγαστή συνεργασία και ο εμπνεόμενος σεβασμός αποτέλεσαν τα βασικά χαρακτηριστικά αυτής της επίβλεψης.

Επίσης θα ήθελα να ευχαριστήσω και να συγχαρώ το προσωπικό και τους υπόλοιπους καθηγητές του Πανεπιστημίου Θεσσαλίας, για τις πολύτιμες γνώσεις και εφόδια που μου προσέφεραν κατά τη διάρκεια της φοίτησής μου.

Θα ήταν παράληψη μου να μην ευχαριστήσω τον υποψήφιο διδάκτορα, συνεργάτη και αδελφό μου Μυριδάκη Δημήτριο, για την αγαστή συνεργασία, επίβλεψη, άμεση υποστήριξη και συμπαράστασή του, υπό την καθοδήγηση του καθηγητή κ. Κακαρούντα Αθανασίου.

Τέλος, θα ήθελα να ευχαριστήσω και να αφιερώσω αυτή τη διπλωματική εργασία στους γονείς, τα αδέρφια και την οικογένειά μου, για την αμέριστη συμπαράσταση που μου εκφράσανε για την εκπόνηση της παρούσας.

Μυριδάκης Παύλος

Λαμία, Σεπτέμβριος 2020

ΠΕΡΙΛΗΨΗ

Στην παρούσα ερευνητική εργασία πραγματοποιήθηκε ένας ψηφιακός μετασχηματιστής τάσης και ένα αυτόνομο κύκλωμα αισθητήρων, μια υπολογιστική πλατφόρμα με ενσωματωμένο μικροελεγκτή, μία απλή πλακέτα ανοικτού κώδικα με τις ελάχιστες θύρες εισόδου / εξόδου για την επικοινωνία με τις εξωτερικές συσκευές. Με αυτόν τον τρόπο ελαχιστοποιήθηκαν οι πόροι, ο όγκος που καταλαμβάνει καθώς και το γενικό κόστος. Προγραμματίστηκε και μπορεί να επαναπρογραμματιστεί σε γλώσσα Wiring. Για τις ανάγκες της εργασίας επιλέχθηκαν από την Atmel Corporation, η οποία εξαγοράστηκε από την Microchip το 2016, οι μικροελεγκτές ATmega328P-PU, ATmega328P-AU και ATtiny85-20PU [1]. Η υπολογιστική πλατφόρμα μπορεί να συνδεθεί με υπολογιστή μέσω FTDI, για την παρακολούθηση της σειριακής θύρας, τον επαναπρογραμματισμό της και μέσω προγραμμάτων σε Processing να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων.

Η ερευνητική εργασία αναπτύχθηκε σε δύο κύρια τμήματα, το θεωρητικό και στο πρακτικό. Στο θεωρητικό τμήμα αναφέρονται ορισμένα ιστορικά στοιχεία για τα ολοκληρωμένα κυκλώματα καθώς και ορισμένες υπολογιστικές πλατφόρμες. Παραθέτονται τα χαρακτηριστικά από τους μικροελεγκτές όπου επιλέχθηκαν και το λογισμικό για τη σχεδίαση σε breadboard, ηλεκτρονικού σχεδίου καθώς και της σχεδίασης του PCB. Ακολουθώς αναφέρονται αναλυτικά το λογισμικό του προγραμματισμού και η μεθοδολογία που ακολουθήθηκε, τόσο για τη φόρτωση του μικροελεγκτή μέσω In System Programming (ISP) όσο και μέσω FTDI σειριακής θύρας.

Στη συνέχεια στο πρακτικό τμήμα της ερευνητικής εργασίας αναλύονται εκτενώς τα βήματα, για τη σύνθεση των υλικών σε breadboard, σε διάτρητη πλακέτα και σε PCB του ενός ή δύο επιπέδων, ανάλογα με τις ανάγκες, για την υλοποίηση ενός ψηφιακού μετασχηματιστή τάσης, μίας πλακέτας για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-PU, μίας πλακέτας για προγραμματισμό μέσω ISP για μικροελεγκτή ATtiny85-20PU, μίας πλακέτας για προγραμματισμό μέσω FTDI (σειριακής θύρας) για μικροελεγκτή ATmega328P-PU και μίας πλακέτας για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-AU, όπου χρειάστηκαν για την επίτευξη και ολοκλήρωση της εργασίας και τέλος της πλακέτας αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU και ATmega328P-AU. Πραγματοποιείται αναφορά στον πηγαίο κώδικα του ψηφιακού

μετασχηματιστή τάσης, καθώς και του προγράμματος, το οποίο επιλέχθηκε για την εκτέλεση μέσω της πλακέτας αυτόνομου κυκλώματος αισθητήρων, για την επίτευξη του αποτελέσματος.

Για την ολοκλήρωση της παρούσας εργασίας διενεργήθηκε βιβλιογραφική επισκόπηση, έρευνα τεχνολογίας μικροελεγκτών, σχεδιασμός, υλοποίηση και τέλος απόδοση των αποτελεσμάτων μέσω των διενεργηθέντων πειραμάτων. Στα αποτελέσματα της εργασίας παρουσιάζονται, η υλοποίηση ενός ψηφιακού μετασχηματιστή τάσης και ενός αυτόνομου κυκλώματος αισθητήρων. Έπειτα από έλεγχο της λειτουργικότητάς των κυκλωμάτων, κατόπιν μετρήσεων του πρώτου και ελέγχου του δεύτερου, μέσα από την εργασία «Intrusion Detection and Botnet Prevention Circuit for IoT Devices», τα αποτελέσματα της δοκιμής του κυκλώματος έδειξαν εξαιρετική απόδοση [2], με 100% επιτυχία των επιθυμητών στόχων ως προς τη λειτουργικότητα και την αποδοτικότητά τους. Η σχεδιάσή του, μπορεί να μεταβληθεί ως προς τις εξόδους του, με βάσει τις ανάγκες τόσο για το πλήθος, όσο και για τον τύπο τους. Με αυτό τον τρόπο μπορούμε να έχουμε ένα μικρό και εξειδικευμένο επαναπρογραμματιζόμενο αυτόνομο κύκλωμα αισθητήρων, με τους απαιτούμενους, για τις ανάγκες μας, ακροδέκτες.

Λέξεις Κλειδιά: Arduino IDE, FTDI FT232R Module, ISP (In System Programming), Αυτόνομο Κύκλωμα Αισθητήρων, Ηλεκτρονικό Σχέδιο, Μικροελεγκτής ATmega328P-AU, Μικροελεγκτής ATmega328P-PU, Μικροελεγκτής ATtiny85-20PU, Σχεδίαση PCB, Ψηφιακός Μετασχηματιστής Τάσης.

ABSTRACT

In the present research work, a digital voltage transformer and an autonomous sensor circuit, a computing platform with a built-in microcontroller, a simple open source board with minimal input / output ports for communication with external devices was realized. In this way, the resources are, the volume it occupies as well as the general cost is minimized. It is programmed and can be reprogrammed in the Wiring language. For the needs of the work, the ATmega328P-PU, ATmega328P-AU and ATtiny85-20PU microcontrollers were selected by Atmel Corporation, which was acquired by Microchip in 2016 [1]. The computing platform can be connected to a computer via FTDI, in order to monitor the serial port, reprogram it and through Programs to be used in order to develop independent interactive objects.

The research work was developed in two main sections, the theoretical and the practical. The theoretical part mentions some historical data about integrated circuits as well as some computing platforms. The characteristics of the microcontrollers which are selected, the software for the design on breadboard, electronic design as well as the design of the PCB are listed. The following is a detailed description of the programming software and the methodology which is followed, both for loading the microcontroller via In System Programming (ISP) and via FTDI serial hunting.

Then, in the practical part of the research work are analyzed in detail the steps, for the synthesis of the materials in breadboard, in perforated board and in PCB of one or two levels, depending on the needs, for the implementation of a digital voltage transformer, a board for programming via ISP for ATmega328P-PU microcontroller, one board for programming via ISP for ATtiny85-20PU microcontroller, one board for programming via FTDI (serial port) for ATmega328P-PU microcontroller and one board for programming via ATU28 to achieve and complete the work and end of the autonomous sensor circuit board with ATmega328P-PU and ATmega328P-AU microcontroller. Reference is made to the source code of the digital voltage transformer, as well as the program, which was selected to run via the stand-alone sensor circuit board, to achieve the result.

To complete the present work, a literature review, microcontroller technology research, design, implementation and finally a performance of the results were performed through the performed experiments. The results of the work are presented, the implementation of a digital

voltage transformer and an autonomous sensor circuit. After testing the functionality of the circuits, after measurements of the first and control of the second, through the work "Intrusion Detection and Botnet Prevention Circuit for IoT Devices", the test results of the circuit showed excellent performance [2], with 100% success of the desired objectives in terms of their functionality and efficiency. Its design can be changed in terms of its outputs, which are based on the needs of both the crowd and their type. In this way, we can have a small and specialized reprogrammable autonomous sensor circuit, with the required terminals for our needs.

Keywords: Arduino IDE, FTDI FT232R Module, ISP (In System Programming), Autonomous Sensor Circuit, Electronic Design, Microcontroller ATmega328P-AU, Microcontroller ATmega328P-PU, Microcontroller ATtiny85-20PU, PCB Design, Digital Voltage Transformer.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ	i
ΠΕΡΙΛΗΨΗ	ii
ABSTRACT	iv
ΠΕΡΙΕΧΟΜΕΝΑ	vi
ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ ΚΑΙ ΣΧΗΜΑΤΩΝ	x
ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ	xiii
ΕΙΣΑΓΩΓΗ ΣΤΑ ΟΛΟΚΛΗΡΩΜΕΝΑ ΚΥΚΛΩΜΑΤΑ	1
1.1 Εισαγωγή	1
1.2 Ιστορική αναδρομή	1
1.3 Μοντέλα από υπολογιστικές πλατφόρμες	4
1.4 Μικροελεγκτές	5
1.5 Μικροελεγκτής Atmel ATmega328P-PU & AU	6
1.5.1 Γενικά Χαρακτηριστικά	6
1.5.1.1 Μικροελεγκτής ATmega328P-PU	6
1.5.1.2 Μικροελεγκτής ATmega328P-AU	7
1.5.2 Μνήμη και τροφοδοσία	7
1.5.3 Ακροδέκτες	7
1.5.3.1 Μικροελεγκτής ATmega328P-PU	7
1.5.3.2 Μικροελεγκτής ATmega328P-AU	9
1.6 Μικροελεγκτής Atmel ATtiny85-20PU	11
1.6.1 Γενικά Χαρακτηριστικά	11
1.6.2 Μνήμη και τροφοδοσία	11
1.6.3 Ακροδέκτες	11
ΛΟΓΙΣΜΙΚΟ ΣΧΕΔΙΑΣΗΣ	13
2.1 Εισαγωγή	13
2.2 Σχεδίαση για breadboard	15
2.2.1 Σχεδίαση ψηφιακού μετασχηματιστή τάσης	15
2.2.2 Σχεδίαση για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU	18
2.2.3 Σχεδίαση για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU	18
2.2.4 Σχεδίαση για προγραμματισμό μέσω FTDI FT232R Module (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU	19

2.2.5	Σχεδίαση αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU	20
2.2.6	Σχεδίαση για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU	21
2.2.7	Σχεδίαση αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU	22
2.3	Ηλεκτρονικό σχέδιο	22
2.4	Σχεδίαση PCB.....	26
ΛΟΓΙΣΜΙΚΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ		30
3.1	Εισαγωγή	30
3.2	Βασικές λειτουργίες λογισμικού	30
3.3	Δομή προγράμματος.....	32
3.4	Συναρτήσεις και εντολές.....	33
3.5	In System Programming (ISP).....	36
3.5.1	Φόρτωση μέσω ISP σε μικροελεγκτή ATmega328.....	37
3.5.1.1	Φόρτωση bootloader μέσω ISP σε μικροελεγκτή ATmega328.....	37
3.5.1.2	Φόρτωση προγράμματος μέσω ISP σε μικροελεγκτή ATmega328.....	39
3.5.2	Φόρτωση προγράμματος μέσω ISP σε μικροελεγκτή ATtiny85-20PU.....	40
3.6	Φόρτωση προγράμματος μέσω FTDI (σειριακής θύρας) σε μικροελεγκτή ATmega328	44
ΥΛΟΠΟΙΗΣΗ ΚΑΤΑΣΚΕΥΗΣ		47
4.1	Εισαγωγή	47
4.2	Ψηφιακός μετασχηματιστής τάσης	47
4.2.1	Σε breadboard	48
4.2.1.1	Υλικά κατασκευής.....	48
4.2.1.2	Υλοποίηση κατασκευής.....	49
4.2.2	Σε διάτρητη πλακέτα	50
4.2.2.1	Υλικά κατασκευής.....	50
4.2.2.2	Υλοποίηση κατασκευής.....	51
4.2.3	Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)	52
4.2.3.1	Υλικά κατασκευής.....	52
4.2.3.2	Υλοποίηση κατασκευής.....	53
4.3	Πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-PU	55
4.3.1	Σε breadboard	56

4.3.1.1	Υλικά κατασκευής.....	56
4.3.1.2	Υλοποίηση κατασκευής.....	57
4.3.2	Σε διάτρητη πλακέτα	57
4.3.2.1	Υλικά κατασκευής.....	57
4.3.2.2	Υλοποίηση κατασκευής.....	58
4.3.3	Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)	59
4.3.3.1	Υλικά κατασκευής.....	59
4.3.3.2	Υλοποίηση κατασκευής.....	60
4.4	Πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-AU	62
4.4.1	Σε breadboard	62
4.4.2	Σε διάτρητη πλακέτα	63
4.4.3	Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)	63
4.4.3.1	Υλικά κατασκευής.....	63
4.4.3.2	Υλοποίηση κατασκευής.....	64
4.5	Πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATtiny85-20PU	66
4.5.1	Σε breadboard	66
4.5.1.1	Υλικά κατασκευής.....	66
4.5.1.2	Υλοποίηση κατασκευής.....	67
4.5.2	Σε διάτρητη πλακέτα	67
4.5.2.1	Υλικά κατασκευής.....	67
4.5.2.2	Υλοποίηση κατασκευής.....	68
4.5.3	Σε χάλκινη πλακέτα ενός επιπέδου (δημιουργία PCB).....	69
4.5.3.1	Υλικά κατασκευής.....	69
4.5.3.2	Υλοποίηση κατασκευής.....	70
4.6	Πλακέτα για προγραμματισμό μέσω FTDI (σειριακής θύρας) σε μικροελεγκτή ATmega328P-PU	71
4.6.1	Σε breadboard	72
4.6.1.1	Υλικά κατασκευής.....	72
4.6.1.2	Υλοποίηση κατασκευής.....	73
4.6.2	Σε διάτρητη και χάλκινη πλακέτα	73
4.7	Πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU	73
4.7.1	Σε breadboard	74
4.7.1.1	Υλικά κατασκευής.....	74
4.7.1.2	Υλοποίηση κατασκευής.....	75
4.7.2	Σε διάτρητη πλακέτα	75

4.7.2.1	Υλικά κατασκευής.....	75
4.7.2.2	Υλοποίηση κατασκευής.....	77
4.7.3	Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)	78
4.7.3.1	Υλικά κατασκευής.....	78
4.7.3.2	Υλοποίηση κατασκευής.....	79
4.8	Πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU	80
4.8.1	Σε breadboard	81
4.8.2	Σε διάτρητη πλακέτα	81
4.8.3	Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)	81
4.8.3.1	Υλικά κατασκευής.....	81
4.8.3.2	Υλοποίηση κατασκευής.....	82
ΥΛΟΠΟΙΗΣΗ ΛΟΓΙΣΜΙΚΟΥ		85
5.1	Εισαγωγή	85
5.2	Υλοποίηση προγραμματισμού.....	85
5.2.1	Προγραμματισμός μικροελεγκτή ATtiny85-20PU	85
5.2.2	Προγραμματισμός μικροελεγκτή ATmega328P-PU & AU	86
ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ		88
6.1	Εισαγωγή	88
6.2	Μετασηματιστής τάσης.....	88
6.3	Αυτόνομο κύκλωμα αισθητήρων με μικροελεγκτή ATmega328.....	89
ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ		91
ΒΙΒΛΙΟΓΡΑΦΙΑ		93
ΗΛΕΚΤΡΟΝΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ		94

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ ΚΑΙ ΣΧΗΜΑΤΩΝ

Εικόνα (1):	Σχέδιο στο πρόγραμμα Fritzing του μετασχηματιστή τάσης για το breadboard.	17
Εικόνα (2):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU για το breadboard.	18
Εικόνα (3):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU για το breadboard.	19
Εικόνα (4):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω FTDI Module (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU για το breadboard.	20
Εικόνα (5):	Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU για το breadboard.	20
Εικόνα (6):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU για το breadboard.	21
Εικόνα (7):	Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU για το breadboard.	22
Εικόνα (8):	Σχέδιο στο πρόγραμμα Multisim του μετασχηματιστή τάσης.	23
Εικόνα (9):	Σχέδιο στο πρόγραμμα Multisim για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.	23
Εικόνα (10):	Σχέδιο στο πρόγραμμα Multisim για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.	24
Εικόνα (11):	Σχέδιο στο πρόγραμμα Multisim για προγραμματισμό μέσω FTDI Module (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU.	24
Εικόνα (12):	Σχέδιο στο πρόγραμμα Multisim αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.	25
Εικόνα (13):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU.	25
Εικόνα (14):	Σχέδιο στο πρόγραμμα Eagle αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU.	26
Εικόνα (15α):	Σχέδιο στο πρόγραμμα Fritzing του μετασχηματιστή τάσης (επίπεδο 1).	26
Εικόνα (15β):	Σχέδιο στο πρόγραμμα Fritzing του μετασχηματιστή τάσης (επίπεδο 2).	27
Εικόνα (16α):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU (επίπεδο 1).	27
Εικόνα (16β):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU (επίπεδο 2).	27
Εικόνα (17):	Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.	27

Εικόνα (18): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU (2 επιπέδων).....	28
Εικόνα (19α): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU (επίπεδο 1).....	28
Εικόνα (19β): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU (επίπεδο 2).....	28
Εικόνα (20α): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU (επίπεδο 1).....	28
Εικόνα (20β): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU (επίπεδο 2).....	29
Εικόνα (21α): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU (2 επιπέδων).	29
Εικόνα (21β): Σχέδιο στο πρόγραμμα Eagle αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU (2 επιπέδων).	29
Εικόνα (22): Πρόγραμμα Arduino IDE.	31
Εικόνα (23): Έτοιμα παραδείγματα κώδικα σε Wiring.....	31
Εικόνα (24): Εισαγωγική απεικόνιση του προγράμματος Arduino IDE.	33
Εικόνα (25): Σύνδεση του προγραμματιστή με τον «κενό» μικροελεγκτή ATmega328. .	38
Εικόνα (26): Σύνδεση του προγραμματιστή με τον «κενό» μικροελεγκτή ATtiny85-20PU.	41
Εικόνα (27α): Μενού επιλογών: Αρχείο/ Προτιμήσεις καρτέλα «Ρυθμίσεις».	42
Εικόνα (27β): Μενού επιλογών: Αρχείο/ Προτιμήσεις καρτέλα «Ρυθμίσεις».	43
Εικόνα (28): Επιπλέον URLs διαχειριστή πλακετών.....	43
Εικόνα (29): Διαχειριστής πλακετών προγράμματος Arduino IDE.	44
Εικόνα (30): Σύνδεση του μικροελεγκτή ATmega328 με FTDI module.	45
Εικόνα (31): Μετασχηματιστής τάσης στο breadboard.	49
Εικόνα (32): Μετασχηματιστής τάσης σε διάτρητη πλακέτα.	52
Εικόνα (33): Μετασχηματιστής τάσης σε χάλκινη πλακέτα.	54
Εικόνα (34): Breadboard για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.....	57
Εικόνα (35): Διάτρητη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.....	59
Εικόνα (36): Χάλκινη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.....	60
Εικόνα (37): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU για το breadboard.....	62

Εικόνα (38):	Χάλκινη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU.	64
Εικόνα (39):	Breadboard για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.	67
Εικόνα (40):	Διάτρητη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.	69
Εικόνα (41):	Χάλκινη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.	70
Εικόνα (42):	Breadboard για προγραμματισμό μέσω FTDI (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU.	73
Εικόνα (43):	Breadboard αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.	75
Εικόνα (44):	Διάτρητη πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.	77
Εικόνα (45):	Χάλκινη πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.	79
Εικόνα (46):	Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU για το breadboard.	81
Εικόνα (47):	Χάλκινη πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU.	82
Εικόνα (48):	Μετρήσεις ρεύματος [2].	89
Εικόνα (49):	Μετρήσεις ρεύματος με το πρώτο φίλτρο [2].	90
Εικόνα (50):	Μετρήσεις ρεύματος με το δεύτερο φίλτρο [2].	90

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

Πίνακας (1α):	Μοντέλα Arduino και διαθεσιμότητα [6].	4
Πίνακας (1β):	Μοντέλα Raspberry Pi και διαθεσιμότητα [8].	5
Πίνακας (1γ):	Μοντέλα SparkFun και διαθεσιμότητα [9].	5
Πίνακας (1δ):	Μοντέλα Orange Pi και διαθεσιμότητα [10].	5
Πίνακας (2α):	Υλικά που χρησιμοποιήθηκαν στο breadboard.	16
Πίνακας (2β):	Υλικά που χρησιμοποιήθηκαν στο breadboard.	17
Πίνακας (3α):	Συναρτήσεις και εντολές για το πρόγραμμα Arduino IDE.	33
Πίνακας (3β):	Συναρτήσεις και εντολές για το πρόγραμμα Arduino IDE.	34
Πίνακας (3γ):	Συναρτήσεις και εντολές για το πρόγραμμα Arduino IDE.	35
Πίνακας (4α):	Υλικά που χρησιμοποιήθηκαν στο breadboard.	48
Πίνακας (4β):	Υλικά που χρησιμοποιήθηκαν στο breadboard.	49
Πίνακας (5α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	50
Πίνακας (5β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	51
Πίνακας (6α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	52
Πίνακας (6β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	53
Πίνακας (7):	Υλικά που χρησιμοποιήθηκαν για breadboard.	56
Πίνακας (8α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	57
Πίνακας (8β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	58
Πίνακας (9α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	59
Πίνακας (9β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	60
Πίνακας (10α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	63
Πίνακας (10β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	64
Πίνακας (11):	Υλικά που χρησιμοποιήθηκαν για breadboard.	66
Πίνακας (12α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	67
Πίνακας (12β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	68
Πίνακας (13α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	69
Πίνακας (13β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	70
Πίνακας (14):	Υλικά που χρησιμοποιήθηκαν για breadboard.	72
Πίνακας (15α):	Υλικά που χρησιμοποιήθηκαν για breadboard.	74
Πίνακας (15β):	Υλικά που χρησιμοποιήθηκαν για breadboard.	75

Πίνακας (16α):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	76
Πίνακας (16β):	Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.	77
Πίνακας (17α):	Υλικά που χρησιμοποιήθηκαν για τη χάλκινη πλακέτα.	78
Πίνακας (17β):	Υλικά που χρησιμοποιήθηκαν για τη χάλκινη πλακέτα.	79
Πίνακας (18):	Υλικά που χρησιμοποιήθηκαν για τη χάλκινη πλακέτα.	82
Πίνακας (19):	Μετρήσεις στη χάλκινη πλακέτα του μετασχηματιστή τάσης.	88

ΚΕΦΑΛΑΙΟ 1^ο

ΕΙΣΑΓΩΓΗ ΣΤΑ ΟΛΟΚΛΗΡΩΜΕΝΑ ΚΥΚΛΩΜΑΤΑ

1.1 Εισαγωγή

Τα ολοκληρωμένα κυκλώματα είναι προηγμένα ηλεκτρικά κυκλώματα. Έχουν εισαχθεί σε πλήθος συσκευών που χρησιμοποιούμε στην καθημερινότητά μας και στη βιομηχανία, παρέχοντας διάφορες ευκολίες στην επικοινωνία, τη μετακίνηση, την ψυχαγωγία, την ιατρική και σε αρκετούς άλλους τομείς καθώς βρίσκουν πλήθος εφαρμογών [3-5].

Στην παρούσα ερευνητική εργασία πραγματοποιήθηκε ένα αυτόνομο κύκλωμα αισθητήρων, μία υπολογιστική πλατφόρμα με τη χρήση μικροελεγκτή, μία απλή πλακέτα ανοικτού κώδικα με τις ελάχιστες θύρες εισόδου / εξόδου για επικοινωνία με τις εξωτερικές συσκευές. Με αυτόν τον τρόπο ελαχιστοποιήθηκαν οι πόροι, ο όγκος που καταλαμβάνει καθώς και το γενικό κόστος. Προγραμματίστηκε και μπορεί να επαναπρογραμματιστεί σε γλώσσα Wiring, όπου πρόκειται για μία γλώσσα προγραμματισμού παρόμοια με τη C / C++ με απλοποιήσεις και αλλαγές, καθώς απαρτίζεται και από ένα σύνολο βιβλιοθηκών [6].

Μπορεί να συνδεθεί με υπολογιστή μέσω FTDI για την παρακολούθηση της σειριακής θύρας, τον επαναπρογραμματισμό του και μέσω προγραμμάτων σε Processing να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων [6].

1.2 Ιστορική αναδρομή

Από τα αρχαία χρόνια ο άνθρωπος προσπαθούσε να υλοποιεί υπολογιστικές μηχανές, προκειμένου να πραγματοποιεί πολύπλοκους ή ογκώδεις υπολογισμούς. Αυτές τις κατασκευές φέρνουν στο φως οι αρχαιολόγοι και οι αρχαιολογικές υπηρεσίες [4].

Οι Σουμέριοι στη Μεσοποταμία, είχαν επινοήσει τον άβακα (Abacus) περί το 2700-2300 π.Χ. [4]. Από τις παλαιότερες είναι και ο μηχανισμός των Αντικυθήρων (Antikythera Mechanism 150 π.Χ.), ένας μηχανισμός μηχανικής ακρίβειας μέτρησης της θέσης των αστεριών [4]. Ο λογαριθμικός κανόνας (Slide Rule), ο οποίος αποτελούσε ένα είδος αναλογικού υπολογιστή. Επινοήθηκε από τον άγγλο μαθηματικό William Oughtred (1622) και ήταν σε χρήση μέχρι τη δεκαετία του 1970, οπότε και αντικαταστάθηκε από τα κομπιουτεράκια χειρός [4]. Από τις πρώτες υπολογιστικές μηχανές, την οποία σχεδίασε ο γάλλος μαθηματικός Blaise Pascal (1642), θεωρείται και η Πασκαλίνα (Pascaline), μία μηχανή όπου μπορούσε να εκτελεί μόνο προσθέσεις και αφαιρέσεις [4]. Η συμβολή του άγγλου μαθηματικού, φιλοσόφου και μηχανικού Charles Babbage (1791-1871), ο οποίος σχεδίασε τη διαφορική μηχανή (Difference Engine), η οποία υπολόγιζε λογαριθμικές και τριγωνομετρικές συναρτήσεις μέσω πολυωνύμων, εκτελούσε πρόσθεση, αφαίρεση, πολλαπλασιασμό και διαίρεση με την ακρίβεια 6 δεκαδικών ψηφίων και την αναλυτική μηχανή (Analytical Engine), σχέδιο ενός μηχανικού υπολογιστή γενικής χρήσης, που θα μπορούσε να προγραμματιστεί με τη χρήση διάτρητων καρτών (δεν κατασκευάστηκε ποτέ) [4]. Ο αμερικανός στατιστικολόγος και εφευρέτης Herman Hollerith (1860-1929) με τη χρήση διάτρητων καρτών, είχε αναπτύξει ένα μηχανικό τρόπο κατανομής μεγάλου όγκου δεδομένων σε πίνακα (Tabulating Machine), έτσι ώστε να μπορέσει να τα επεξεργαστεί [4]. Το 1936 προτάθηκε από τον Alan Turing βρετανό μαθηματικό, κρυπτογράφο και καθηγητή Λογικής, ως μια «λογική υπολογιστική καθολική μηχανή» μια υποθετική συσκευή όπου η ιδέα υπήρξε καθοριστική για την εξέλιξη της Πληροφορικής, αποδεικνύοντας ότι οποιοσδήποτε υπολογισμός μπορεί να εκτελεστεί από μια μηχανή εάν έχουν εκφραστεί ξεκάθαρα τα αναγκαία βήματα [4].

Στο Β' Παγκόσμιο Πόλεμο οι Βρετανοί κατασκεύασαν μια ηλεκτρομηχανική συσκευή, τον Bombe (Bombe), για την αποκρυπτογράφηση των μηνυμάτων της μηχανής κρυπτογράφησης Enigma (Enigma Machine), που χρησιμοποιούσε ο γερμανικός στρατός [4]. Με τη λήξη του Β' Παγκόσμιου Πολέμου, οι εξελίξεις στο χώρο των ηλεκτρονικών υπολογιστών ήταν ραγδαίες, διότι έως εκείνη τη χρονική περίοδο το κύριο τεχνικό χαρακτηριστικό τους ήταν η χρήση των ηλεκτρονικών λυχνιών, ως βασικό εξάρτημα για τη λειτουργία τους [4]. Οι Presper Eckert και John Mauchly καθηγητές στο Moore School του University of Pennsylvania (ΗΠΑ) κατασκεύασαν μία από τις πρώτες υπολογιστικές μηχανές (αποκαλύφθηκε δημόσια το 1946), τον ηλεκτρονικό αριθμητικό ολοκληρωτή και υπολογιστή (Electronic Numerical Integrator and Computer / ENIAC) ο πρώτος

επαναπρογραμματιζόμενος ψηφιακός υπολογιστής, ο οποίος χρησιμοποιήθηκε για στρατιωτικούς σκοπούς. Ήταν αρκετά μεγάλος, ζύγιζε περίπου 30 τόνους και κατανάλωνε 200 KW ηλεκτρικής ενέργειας. Διέθετε περίπου 18000 ηλεκτρονικές λυχνίες που καθιστούσαν το σύστημα αναξιόπιστο, εφόσον χρειάζονταν συχνό έλεγχο και αντικατάσταση [3, 4].

Κατά τη δεκαετία του 1950, με την έλευση του τρανζίστορ (Transistor), το οποίο εφευρέθηκε στα εργαστήρια Bell Labs της αμερικανικής εταιρείας AT&T, έδωσε τεράστια ώθηση στην εξέλιξη των ηλεκτρονικών υπολογιστών και στην αξιοποίησή τους [4]. Πριν από το τρανζίστορ, οι μηχανικοί αναγκάζονταν να χρησιμοποιούν ηλεκτρονικές λυχνίες οι οποίες ήταν αργές, ογκώδης και συμπεριφέρονταν παρόμοια με μία ηλεκτρική λάμπα, παράγοντας αρκετή θερμότητα, είχαν την τάση να καίγονται και κατανάλωναν περισσότερη ενέργεια σε σχέση με το τρανζίστορ [3]. Πριν την ανακάλυψη του ολοκληρωμένου, η συναρμολόγηση των κυκλωμάτων γινόταν με το χέρι [3]. Ο Jack Kilby το 1958 δουλεύοντας στην εταιρεία Texas Instruments αντιλήφθηκε το πρόβλημα σμίκρυνσης των κυκλωμάτων και εφάρμοσε τη βασική ιδέα του που ήταν η κατασκευή όλων των στοιχείων πάνω στο ίδιο κομμάτι (μονόλιθος) ημιαγωγού (semiconductor). Το πρώτο λειτουργικό ολοκληρωμένο κύκλωμα ήταν έτοιμο [3]. Οι ενώσεις δεν γίνονταν πλέον με το χέρι. Η διαδικασία κατασκευής μπορούσε να αυτοματοποιηθεί και τα κυκλώματα μπορούσαν να γίνουν μικρότερα [3]. Ο Robert Noyce, μισό χρόνο μετά από τον Kilby επιλύει πρακτικά το πρόβλημα της διασύνδεσης των διάφορων στοιχείων πάνω στο ολοκληρωμένο κύκλωμα, που είχε το κύκλωμα του Kilby. Αυτή η αλλαγή, κατέστησε δυνατή την μαζική παραγωγή των ολοκληρωμένων κυκλωμάτων [3]. Στη δεκαετία του 1960 και του 1970 εμφανίστηκαν υπολογιστικές μηχανές σχετικά ευέλικτες όπου εγκαταστάθηκαν σε επιχειρήσεις, οργανισμούς, πανεπιστήμια, ινστιτούτα και στη συνέχεια σε μεμονωμένους χρήστες [4].

Την εξέλιξη αυτή μας έδωσε η εξέλιξη στα ηλεκτρικά κυκλώματα, τα οποία ενώνονται μεταξύ τους με διάφορους τρόπους και αποτελούνται από διάφορα ηλεκτρικά στοιχεία, που έχουν διαφορετική συμπεριφορά, όπως:

- Τρανζίστορ: δουλεύει σαν ένας διακόπτης
- Αντίσταση: περιορίζει την ροή του ηλεκτρισμού
- Πυκνωτής: συγκεντρώνει ρεύμα και το απελευθερώνει απότομα

Δίοδος: σταματά τη διέλευση ρεύματος κάτω από κάποιες συνθήκες και επιτρέπει τη διέλευση ρεύματος όταν οι συνθήκες αυτές αλλάζουν.

Το ολοκληρωμένο κύκλωμα είναι ένα προηγμένο ηλεκτρικό κύκλωμα [3, 7]. Η παραγωγή των ολοκληρωμένων κυκλωμάτων σήμερα βασίζεται στην φωτολιθογραφία, όπου υπεριώδη ακτινοβολία εκπέμπεται περνώντας από μια μάσκα σε ένα κομμάτι πυρίτιο που είναι καλυμμένο με ένα φωτοευαίσθητο φιλμ [3]. Τα τρανζίστορ φθάνουν πλέον μέγεθος περίπου 90nm, δηλαδή 0.00009 χιλιοστάμετρα [3].

1.3 Μοντέλα από υπολογιστικές πλατφόρμες

Τα μοντέλα από τις απλές υπολογιστικές πλατφόρμες διαφοροποιούνται ανάλογα με τη χρήση, τον μικροελεγκτή, την τάση λειτουργίας, τη μνήμη, την ταχύτητα του ρολογιού καθώς και τις εισόδους και εξόδους που παρέχουν.

Ορισμένα από αυτά παρατίθενται ακολούθως καθώς και η διαθεσιμότητά τους:

- Η πλακέτα arduino αρχικά ξεκίνησε ως ερευνητικό έργο από τους Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino και David Mellis στο Interaction Design Institute of Ivrea (Ιταλία) στις αρχές της δεκαετίας του 2000 [6].
- Το raspberry pi αναπτύχθηκε στο Ηνωμένο Βασίλειο [8].
- Η πλακέτα sparkfun κατασκευάζεται κυρίως από την SparkFun Electronics με έδρα το Boulder στο Κολοράντο των ΗΠΑ [9].
- Το orange pi κατασκευάζεται από την Shenzhen Xunlong Software CO [10].

A/A	Μοντέλο	Διαθέσιμο στην ιστοσελίδα
1	Arduino Uno Rev3	https://store.arduino.cc/arduino-uno-rev3
2	Arduino Nano	https://store.arduino.cc/arduino-nano
3	Arduino Mega 2560 Rev3	https://store.arduino.cc/arduino-mega-2560-rev3
4	Arduino Uno Rev3 SMD	https://store.arduino.cc/arduino-uno-rev3-smd

Πίνακας (1α): Μοντέλα Arduino και διαθεσιμότητα [6].

A/A	Μοντέλο	Διαθέσιμο στην ιστοσελίδα
1	Raspberry Pi 2 Model B	https://www.raspberrypi.org/products/raspberry-pi-2-model-b/
2	Raspberry Pi 3 Model A+	https://www.raspberrypi.org/products/raspberry-pi-3-model-a-plus/
3	Raspberry Pi 4	https://www.raspberrypi.org/products/raspberry-pi-4-model-b/

Πίνακας (1β): Μοντέλα Raspberry Pi και διαθεσιμότητα [8].

A/A	Μοντέλο	Διαθέσιμο στην ιστοσελίδα
1	SparkFun Qwiic Pro Micro – USB-C (ATmega32U4)	https://www.sparkfun.com/products/15795
2	SparkFun RedBoard – Programmed with Arduino	https://www.sparkfun.com/products/13975
3	SparkFun AST-CAN485 Dev Board	https://www.sparkfun.com/products/14483

Πίνακας (1γ): Μοντέλα SparkFun και διαθεσιμότητα [9].

A/A	Μοντέλο	Διαθέσιμο στην ιστοσελίδα
1	Orange Pi Zero	http://www.orangepi.org/orangepizero/
2	Orange Pi One	http://www.orangepi.org/orangepione/
3	Orange Pi 4B	http://www.orangepi.org/Orange 20Pi 204B/

Πίνακας (1δ): Μοντέλα Orange Pi και διαθεσιμότητα [10].

1.4 Μικροελεγκτές

Ο μικροελεγκτής είναι ένα ολοκληρωμένο κύκλωμα όπου διαθέτει επεξεργαστή, μνήμη, θύρες εισόδου / εξόδου και μπορεί να προγραμματιστεί. Οι εταιρείες που απασχολούνται σε αυτό τον τομέα παγκοσμίως, παράγουν μία μεγάλη ποικιλία, από τους πολύ μικρούς και

οικονομικούς, για τις απλές εφαρμογές, μέχρι τους προηγμένους, για πολύπλοκες και απαιτητικές χρήσεις.

Ορισμένες και από τις πιο διαδεδομένες είναι οι ακόλουθες:

Microchip, www.microchip.com

NXP Semiconductor, www.nxp.com

Texas Instruments, www.ti.com

Από την εταιρεία Atmel όπου εξαγοράστηκε από την Microchip το 2016, επιλέχθηκαν οι μικροελεγκτές ATmega328P-PU, ATmega328P-AU και ATtiny85-20PU, για τις ανάγκες της εργασίας και τα χαρακτηριστικά που μας παρέχουν, οι οποίοι παραθέτονται παρακάτω [1].

1.5 Μικροελεγκτής Atmel ATmega328P-PU & AU

Ο μικροελεγκτής ATmega328P-PU έχει 28 ακροδέκτες και ο μικροελεγκτής ATmega328P-AU έχει 32 ακροδέκτες για τροφοδοσία, γείωση, λειτουργία και επικοινωνία. Είναι μικροελεγκτές CMOS 8-bit χαμηλής ισχύος, που βασίζονται στην αρχιτεκτονική RISC ενισχυμένη με AVR, επιτρέποντας στον σχεδιαστή του συστήματος να βελτιστοποιήσει την κατανάλωση ισχύος έναντι της ταχύτητας επεξεργασίας [5, 11].

1.5.1 Γενικά Χαρακτηριστικά

Ταχύτητα εσωτερικού ρολογιού φθάνει έως και τα 8 MHz και με εξωτερικό κρυσταλλικό ταλαντωτή φθάνει έως και τα 20 MHz [11].

Θερμοκρασία λειτουργίας από -40 έως 85 °C [11].

1.5.1.1 Μικροελεγκτής ATmega328P-PU

Πλάτος: 7,620 με 8,255 χιλ [11].

Μήκος: 34,544 με 34,798 χιλ [11].

1.5.1.2 Μικροελεγκτής ATmega328P-AU

Πλάτος: 8,75 με 9,25 χιλ [11].

Μήκος: 8,75 με 9,25 χιλ [11].

1.5.2 Μνήμη και τροφοδοσία

Οι μικροελεγκτές ATmega328P-PU & AU χειρίζονται μνήμη flash 32KB για αποθήκευση κώδικα, από τα οποία αφαιρούνται τα 0,5KB διότι χρησιμοποιούνται για την φόρτωση εκκίνησης (bootloader) από εξωτερικό προγραμματισμό, SRAM 2 KB και EEPROM 1 KB [5, 11].

Η παροχή ρεύματος είναι από πηγή 3,3 έως 5 volt και το ρεύμα λειτουργίας είναι από 1,8 έως 5,5 volt [11].

1.5.3 Ακροδέκτες

1.5.3.1 Μικροελεγκτής ATmega328P-PU

Οι 28 ακροδέκτες του μικροελεγκτή ATmega328P-PU αναλύονται ως ακολούθως, με τη σειρά που είναι επάνω στον μικροελεγκτή [5, 11]:

Pin 1: Ακροδέκτης επανεκκίνησης.

Pin 2: RX (receive) Ψηφιακός ακροδέκτης λήψης για τη σειριακή λειτουργία.

Pin 3: TX (transmit) Ψηφιακός ακροδέκτης αποστολής για τη σειριακή λειτουργία.

Pin 4: Ψηφιακός ακροδέκτης εισόδου / εξόδου.

Pin 5: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 6: Ψηφιακός ακροδέκτης εισόδου / εξόδου.

Pin 7: Ακροδέκτης τροφοδοσίας.

Pin 8: Ακροδέκτης γείωσης.

- Pin 9: XTAL1 για τη σύνδεση με τον εξωτερικό κρυσταλλικό ταλαντωτή.
- Pin 10: XTAL2 για τη σύνδεση με τον εξωτερικό κρυσταλλικό ταλαντωτή.
- Pin 11: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).
- Pin 12: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).
- Pin 13: Ψηφιακός ακροδέκτης εισόδου / εξόδου.
- Pin 14: Ψηφιακός ακροδέκτης εισόδου / εξόδου.
- Pin 15: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).
- Pin 16: SS Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).
- Pin 17: MOSI Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).
- Pin 18: MISO Ψηφιακός ακροδέκτης εισόδου / εξόδου.
- Pin 19: SCK Ψηφιακός ακροδέκτης εισόδου / εξόδου.
- Pin 20: Ακροδέκτης τροφοδοσίας.
- Pin 21: AREF (ο αναλογικός ακροδέκτης αναφοράς για τον μετατροπέα A/D).
- Pin 22: Ακροδέκτης γείωσης.
- Pin 23: Αναλογικός ακροδέκτης εισόδου.
- Pin 24: Αναλογικός ακροδέκτης εισόδου.
- Pin 25: Αναλογικός ακροδέκτης εισόδου.
- Pin 26: Αναλογικός ακροδέκτης εισόδου.

Pin 27: Αναλογικός ακροδέκτης εισόδου.

Pin 28: Αναλογικός ακροδέκτης εισόδου.

1.5.3.2 Μικροελεγκτής ATmega328P-AU

Οι 32 ακροδέκτες του μικροελεγκτή ATmega328P-AU αναλύονται ως ακολούθως, με τη σειρά που είναι επάνω στον μικροελεγκτή [11]:

Pin 1: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 2: Ψηφιακός ακροδέκτης εισόδου / εξόδου.

Pin 3: Ακροδέκτης γείωσης.

Pin 4: Ακροδέκτης τροφοδοσίας.

Pin 5: Ακροδέκτης γείωσης.

Pin 6: Ακροδέκτης τροφοδοσίας.

Pin 7: XTAL1 για τη σύνδεση με τον εξωτερικό κρυσταλλικό ταλαντωτή.

Pin 8: XTAL2 για τη σύνδεση με τον εξωτερικό κρυσταλλικό ταλαντωτή.

Pin 9: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 10: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 11: Ψηφιακός ακροδέκτης εισόδου / εξόδου.

Pin 12: Ψηφιακός ακροδέκτης εισόδου / εξόδου.

Pin 13: Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 14: SS Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 15: MOSI Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 16: MISO Ψηφιακός ακροδέκτης εισόδου / εξόδου.

Pin 17: SCK Ψηφιακός ακροδέκτης εισόδου / εξόδου.

Pin 18: Ακροδέκτης τροφοδοσίας.

Pin 19: Αναλογικός ακροδέκτης εισόδου (στο μετατροπέα A/D ο οποίος τροφοδοτείται από την αναλογική τροφοδοσία και χρησιμεύει ως κανάλι ADC 10-bit).

Pin 20: AREF (ο αναλογικός ακροδέκτης αναφοράς για τον μετατροπέα A/D).

Pin 21: Ακροδέκτης γείωσης.

Pin 22: Αναλογικός ακροδέκτης εισόδου (στο μετατροπέα A/D ο οποίος τροφοδοτείται από την αναλογική τροφοδοσία και χρησιμεύει ως κανάλι ADC 10-bit).

Pin 23: Αναλογικός ακροδέκτης εισόδου.

Pin 24: Αναλογικός ακροδέκτης εισόδου.

Pin 25: Αναλογικός ακροδέκτης εισόδου.

Pin 26: Αναλογικός ακροδέκτης εισόδου.

Pin 27: Αναλογικός ακροδέκτης εισόδου.

Pin 28: Αναλογικός ακροδέκτης εισόδου.

Pin 29: Ακροδέκτης επανεκκίνησης.

Pin 30: RX (receive) Ψηφιακός ακροδέκτης λήψης για τη σειριακή λειτουργία.

Pin 31: TX (transmit) Ψηφιακός ακροδέκτης αποστολής για τη σειριακή λειτουργία.

Pin 32: Ψηφιακός ακροδέκτης εισόδου / εξόδου.

1.6 Μικροελεγκτής Atmel ATtiny85-20PU

Ο μικροελεγκτής ATtiny85-20PU έχει 8 ακροδέκτες για τροφοδοσία, γείωση, λειτουργία και επικοινωνία. Είναι ένας μικροελεγκτής CMOS 8-bit χαμηλής ισχύος που βασίζεται στην αρχιτεκτονική RISC ενισχυμένη με AVR [11].

1.6.1 Γενικά Χαρακτηριστικά

Πλάτος: 7,620 με 8,255 χιλ [11].

Μήκος: 9,017 με 10,16 χιλ [11].

Ταχύτητα εσωτερικού ρολογιού φθάνει έως και τα 16 MHz και με εξωτερικό κρυσταλλικό ταλαντωτή τα 20 MHz [11].

Θερμοκρασία λειτουργίας από -40 έως 85 °C [11].

1.6.2 Μνήμη και τροφοδοσία

Ο Μικροελεγκτής ATtiny85-20PU χειρίζεται μνήμη flash 8KB για αποθήκευση κώδικα, SRAM 256 bytes και EEPROM 512 bytes. Η παροχή ρεύματος είναι από πηγή 3,3 έως 5 volt και το ρεύμα λειτουργίας είναι από 2,7 έως 5,5 volt [11].

1.6.3 Ακροδέκτες

Οι 8 ακροδέκτες του μικροελεγκτή ATtiny85-20PU αναλύονται ως ακολούθως, με τη σειρά που είναι επάνω στον μικροελεγκτή [11]:

Pin 1: Ακροδέκτης επανεκκίνησης.

Pin 2: XTAL1 για τη σύνδεση με τον εξωτερικό κρυσταλλικό ταλαντωτή, αναλογικός ακροδέκτης εισόδου.

Pin 3: XTAL2 για τη σύνδεση με τον εξωτερικό κρυσταλλικό ταλαντωτή, αναλογικός ακροδέκτης εισόδου.

Pin 4: Ακροδέκτης γείωσης.

Pin 5: MOSI/AREF, Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 6: MISO, Ψηφιακός ακροδέκτης εισόδου / εξόδου όπου μπορεί να χρησιμοποιηθεί και ως PWM (Pulse Width Modulation-Διαμόρφωση Πλάτους Παλμού).

Pin 7: SCK, αναλογικός ακροδέκτης εισόδου.

Pin 8: Ακροδέκτης τροφοδοσίας.

ΚΕΦΑΛΑΙΟ 2^ο

ΛΟΓΙΣΜΙΚΟ ΣΧΕΔΙΑΣΗΣ

2.1 Εισαγωγή

Τα προγράμματα σχεδίασης ηλεκτρονικών, πλακετών και πειραματισμού σε breadboard είναι αρκετά, ορισμένα από αυτά αναφέρονται παρακάτω:

Τα επικρατέστερα είναι ανοικτού λογισμικού (Open Source), όπως το **Fritzing**, το **KiCad** και το **gEDA**. Ορισμένα από τα πλεονεκτήματα του ανοικτού λογισμικού που αξίζει να αναφερθούν είναι η δωρεάν διάθεση και χρήση, διορθώσεις και επεκτάσεις, οι οποίες δημιουργούνται από τους συμμετέχοντες στην ανάπτυξη των προγραμμάτων, καθώς και από τους ίδιους τους χρήστες [12-14].

Το PCB Artist (**Free PCB Layout Software**) εκτελείται σε λειτουργικό σύστημα Windows. Το Linux και το MAC OS δεν υποστηρίζονται προς το παρόν, αν και οι χρήστες είχαν επιτυχία χρησιμοποιώντας εξομοιωτές. Συνιστάται επεξεργαστής Pentium ταχύτερος από 1,8Ghz και με τουλάχιστον 512mb RAM. Το PCB Artist δεν απαιτεί ειδικό υλικό υψηλής ισχύος για να επιτύχει καλή απόδοση. Η πλήρης εγκατάσταση του απαιτεί περίπου 1,3 GB χώρου στο σκληρό δίσκο. Συνιστάται η πρόσβαση στο Διαδίκτυο [15].

Το **TinyCAD** είναι ένα δωρεάν πρόγραμμα σχεδίασης διαγραμμάτων ηλεκτρονικών κυκλωμάτων (σχηματικά σχέδια). Υποστηρίζει βιβλιοθήκες πρότυπων και προσαρμοσμένων συμβόλων. Υποστηρίζει προγράμματα διάταξης PCB με διάφορες μορφές. Συχνά χρησιμοποιείται για την κατάρτιση διαγραμμάτων μιας γραμμής, διαγραμμάτων μπλοκ και σχεδίων παρουσίασης [16].

Το **PCB Creator** βασίζεται στο λογισμικό επαγγελματικών σχεδιασμών PCB και είναι πολύ εύκολο στη χρήση. Πρόκειται για ένα πλήρες πρόγραμμα σχεδίασης ηλεκτρονικών κυκλωμάτων και τυπωμένων κυκλωμάτων με πολύ μεγάλες δυνατότητες. Η χρήση του είναι πολύ απλή [17].

Το **Multisim Live** είναι ένας δωρεάν, online προσομοιωτής κυκλώματος που περιλαμβάνει λογισμικό SPICE, το οποίο μας επιτρέπει να δημιουργήσουμε και να μοιραζόμαστε κυκλώματα ηλεκτρονικά [18].

Το **Multisim** είναι τυποποιημένο λογισμικό προσομοίωσης και σχεδιασμού κυκλωμάτων SPICE για αναλογικά, ψηφιακά και ηλεκτρονικά στην εκπαίδευση και την έρευνα. Το Multisim ενσωματώνει την τυπική βιομηχανική προσομοίωση SPICE με ένα διαδραστικό σχηματικό περιβάλλον για την άμεση οπτικοποίηση και ανάλυση της συμπεριφοράς των ηλεκτρονικών κυκλωμάτων. Το Multisim έχει μια διαισθητική διεπαφή που βοηθά τους εκπαιδευτικούς να ενισχύσουν τη θεωρία του κυκλώματος και να βελτιώσουν τη διατήρηση της θεωρίας σε όλο το πρόγραμμα σπουδών μηχανικής. Οι ερευνητές και οι σχεδιαστές χρησιμοποιούν το Multisim για να μειώσουν τις επαναλήψεις πρωτοτύπων PCB και να εξοικονομήσουν κόστος ανάπτυξης προσθέτοντας ισχυρή προσομοίωση κυκλώματος και αναλύσεις στη ροή του σχεδιασμού [19].

Το πρόγραμμα Multisim της εταιρίας National Instruments είναι η εξέλιξη του προγράμματος Electronics Workbench [19].

Το **Eagle** είναι ένα λογισμικό για το σχεδιασμό κυκλωμάτων με αρκετές δυνατότητες. Αποθηκεύει αρχεία διάταξης Gerber και PostScript, καθώς και αρχεία τρυπανιών Excellon και Sieb & Meyer, τυπικές μορφές αρχείων, αποδεκτές από εταιρείες κατασκευής PCB [20].

Από τα παραπάνω επιλέχθηκαν:

- το Fritzing για τη σχεδίαση στο breadboard, του ηλεκτρονικού σχεδίου και του PCB
- το Multisim για την υλοποίηση του ηλεκτρονικού σχεδίου
- το Eagle για την υλοποίηση του ηλεκτρονικού σχεδίου και σχεδίαση PCB

2.2 Σχεδίαση για breadboard

Για τις ανάγκες της εργασίας σχεδιάστηκαν:

- ένας ψηφιακός μετασχηματιστής τάσης
- μία πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-PU
- μία πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-AU
- μία πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATtiny85-20PU
- μία πλακέτα για προγραμματισμό μέσω FTDI (σειριακής θύρας) για μικροελεγκτή ATmega328P-PU

όπου χρειάστηκαν για την επίτευξη και ολοκλήρωση της εργασίας και τέλος:












- η πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU και
- η πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU.

Για τη σχεδίαση και πειραματισμό, πριν την υλοποίηση με τα υλικά, χρησιμοποιήθηκαν τα προγράμματα Fritzing, Multisim καθώς και το πρόγραμμα Eagle, για τα οποία θα αναφερθούμε παρακάτω.



2.2.1 Σχεδίαση ψηφιακού μετασχηματιστή τάσης

Για τον ψηφιακό μετασχηματιστή τάσης, οι οδηγίες και περιορισμοί της κατασκευής ήταν να λαμβάνει ως τάση εισόδου 12 volt και να μας παρέχει τη δυνατότητα επιλογής εξόδου σε 3,3 - 5 - 9 και 12 volt.

Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά με βάση τις ανάγκες της εργασίας, όπου και παραθέτονται ακολούθως [11, 21, 22]:

Dc Power Jack	
Electrolytic Capacitor 25V 47uF	
Voltage Regulator LD1117 - 3.3V 800mA	
Voltage Regulator L78S05CV - 5V 2A	
Voltage Regulator L7809CV - 9V 1.5A	
LED Green 5mm THT	
LED Yellow 5mm THT	
Resistor THT 1kΩ και 220Ω	
Diodes 1N4007	
Tact Switch	
Cap for Tact Button	
DIP Socket 8 Pin	
Microcontroller ATtiny85-20PU	

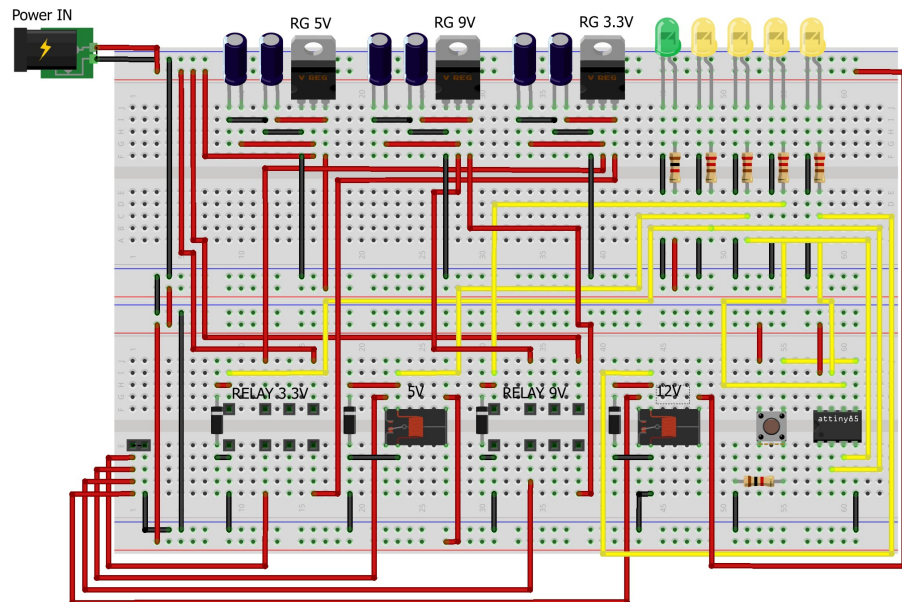
Πίνακας (2α): Υλικά που χρησιμοποιήθηκαν στο breadboard.

Relay JRC-19F & MT2 C93428 12V DPDT (1A/125V)	
Relay HUI KE HK4100F-DC12V 3A	

Πίνακας (2β): Υλικά που χρησιμοποιήθηκαν στο breadboard.

Από τα βασικά υλικά που χρησιμοποιήθηκαν ήταν το Voltage Regulator, για την καλύτερη σταθεροποίηση της αποδιδόμενης τάσης, ο μικροελεγκτής ATtiny85-20PU, προγραμματισμένος κατάλληλα καθώς και το Relay για την επίτευξη του αποτελέσματος.

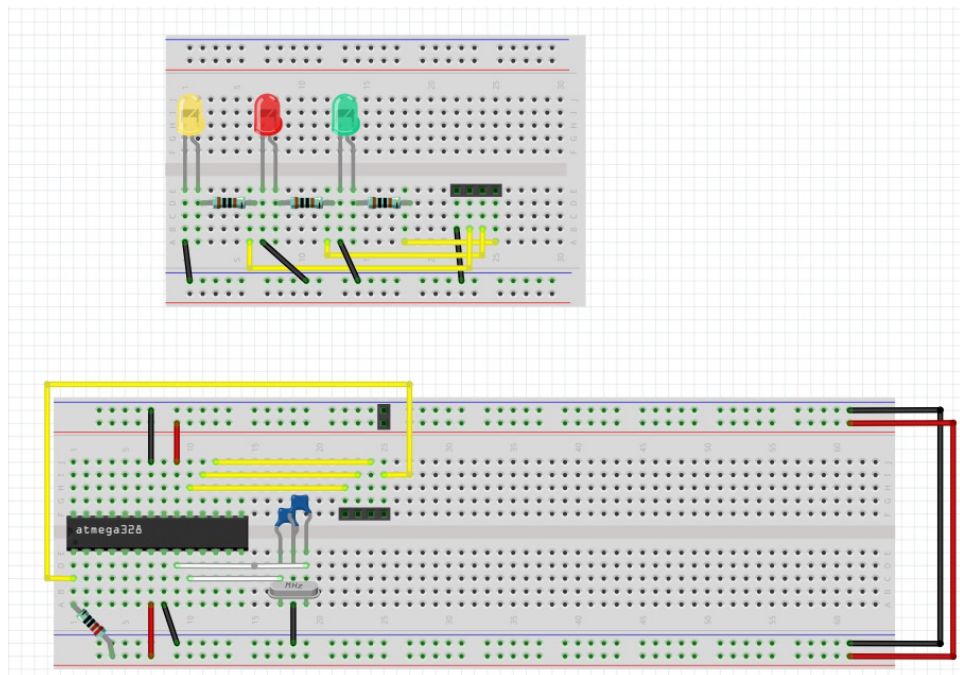
Από τη σχεδίαση στο πρόγραμμα Fritzing καταλήξαμε στο ακόλουθο σχέδιο για το breadboard.



Εικόνα (1): Σχέδιο στο πρόγραμμα Fritzing του μετασχηματιστή τάσης για το breadboard.

2.2.2 Σχεδίαση για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU

Για τις ανάγκες της εργασίας ήταν απαραίτητο οι κενοί (άδειοι) μικροελεγκτές ATmega328P-PU, για να μπορέσουμε να τους προγραμματίσουμε, να κατασκευάσουμε μια μόνιμη πλακέτα που θα μας έδινε αυτή τη δυνατότητα. Η διαδικασία και περισσότερες λεπτομέρειες θα παρατεθούν σε ακόλουθη παράγραφο.

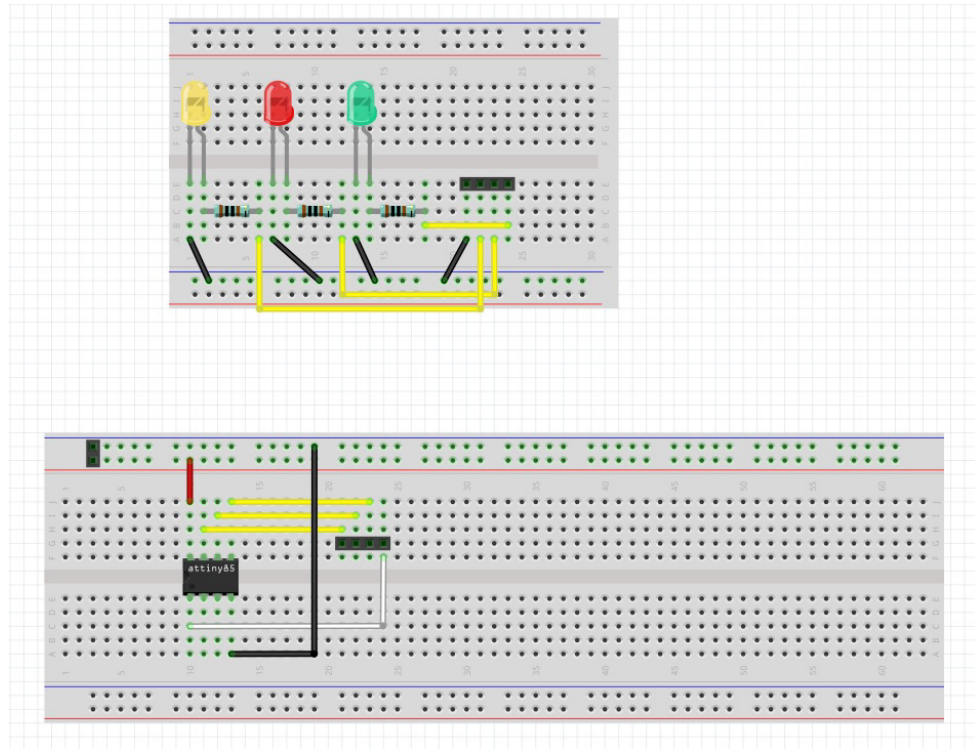


Εικόνα (2): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU για το breadboard.

Το σχέδιο αποτελείται από δύο μέρη, όπως απεικονίζεται στην παραπάνω εικόνα. Το κάτω breadboard απεικονίζει τη συνδεσμολογία για τον προγραμματισμό του μικροελεγκτή και το επάνω breadboard απεικονίζει τη συνδεσμολογία για τον οπτικό έλεγχο της ανατροφοδότησης σχετικά με τη διαδικασία του προγραμματισμού [6, 7].

2.2.3 Σχεδίαση για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU

Απαραίτητο ήταν οι κενοί (άδειοι) μικροελεγκτές ATtiny85-20PU να προγραμματιστούν, κατασκευάζοντας μία μόνιμη πλακέτα. Η διαδικασία και περισσότερες λεπτομέρειες θα παρατεθούν σε ακόλουθη παράγραφο.



Εικόνα (3): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU για το breadboard.

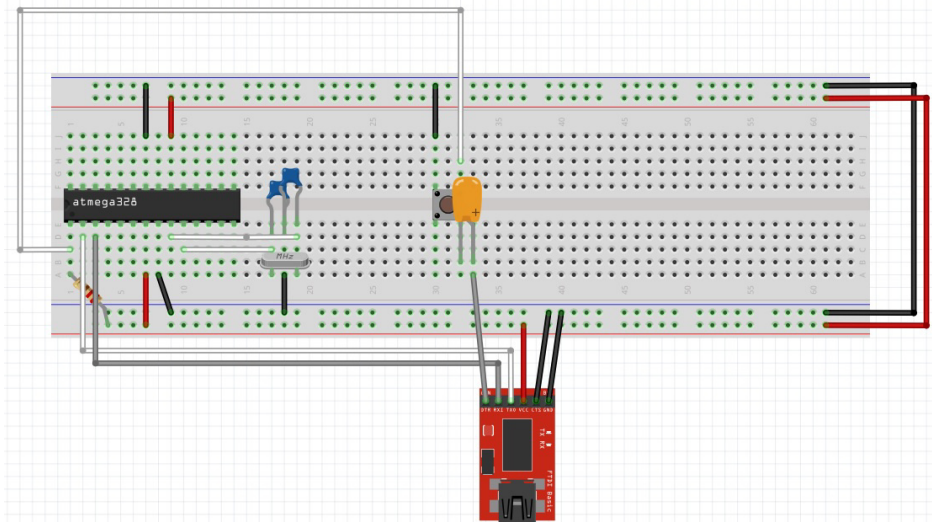
Το σχέδιο αποτελείται από δύο μέρη, όπως απεικονίζεται στην παραπάνω εικόνα. Το κάτω breadboard απεικονίζει τη συνδεσμολογία για τον προγραμματισμό του μικροελεγκτή και το επάνω breadboard απεικονίζει τη συνδεσμολογία για τον οπτικό έλεγχο της ανατροφοδότησης σχετικά με τη διαδικασία του προγραμματισμού [6, 7].

2.2.4 Σχεδίαση για προγραμματισμό μέσω FTDI FT232R Module (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU

Για τον ευκολότερο προγραμματισμό και επαναπρογραμματισμό του μικροελεγκτή ATmega328P-PU σχεδιάστηκε ο προγραμματισμός μέσω FTDI FT232R Module.

Υποσημείωση: Θα πρέπει να έχουμε φορτώσει επιτυχώς το bootloader στον μικροελεγκτή μέσω της διαδικασίας του ISP και δεν θα έχουμε φορτώσει ακολούθως κάποιο άλλο πρόγραμμα με την ίδια διαδικασία στον ίδιο μικροελεγκτή, διότι είναι πιθανό να μας σβήσει το bootloader [6, 7].

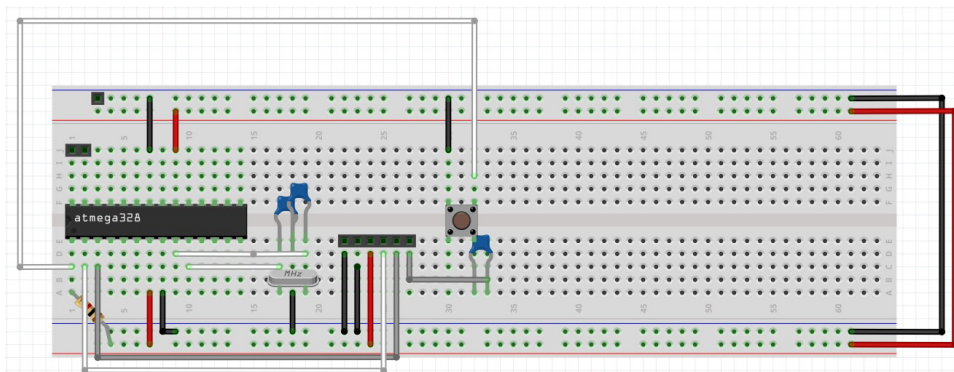
Η διαδικασία ISP θα αναλυθεί σε ακόλουθη παράγραφο.



Εικόνα (4): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω FTDI Module (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU για το breadboard.

2.2.5 Σχεδίαση αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU

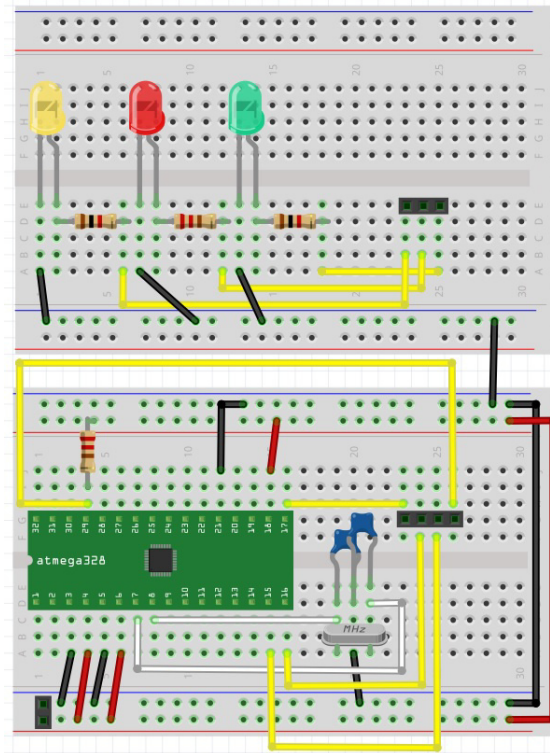
Παίρνοντας τα δεδομένα και τις ανάγκες της εργασίας για την ελαχιστοποίηση των εξόδων, την παρακολούθηση της σειριακής θύρας καθώς και τη χρησιμοποίηση αναλογικών θυρών του μικροελεγκτή, προχωρήσαμε στον ακόλουθο σχεδιασμό του αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.



Εικόνα (5): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU για το breadboard.

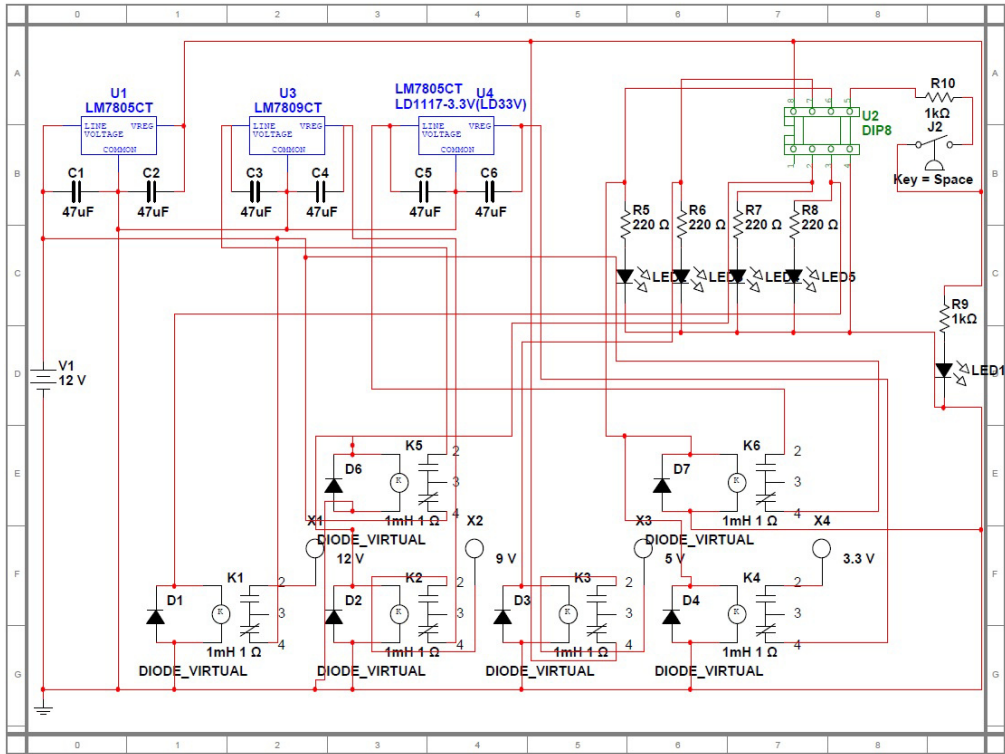
2.2.6 Σχεδίαση για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU

Για τις ανάγκες της εργασίας ήταν απαραίτητο οι κενοί (άδειοι) μικροελεγκτές ATmega328P-AU, για να μπορέσουμε να τους προγραμματίσουμε, να κατασκευάσουμε μια μόνιμη πλακέτα που θα μας έδινε αυτή τη δυνατότητα. Η διαδικασία και περισσότερες λεπτομέρειες θα παρατεθούν σε ακόλουθη παράγραφο.

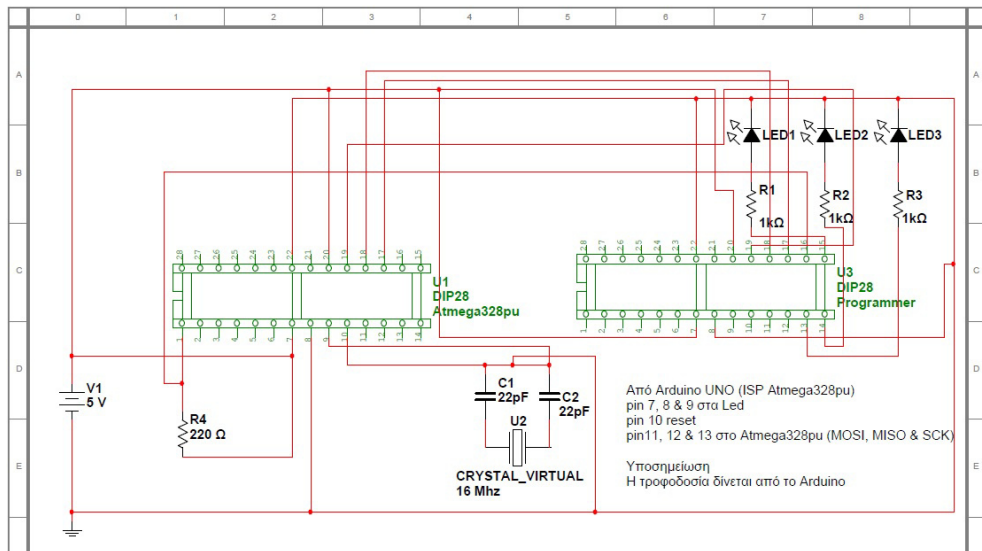


Εικόνα (6): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU για το breadboard.

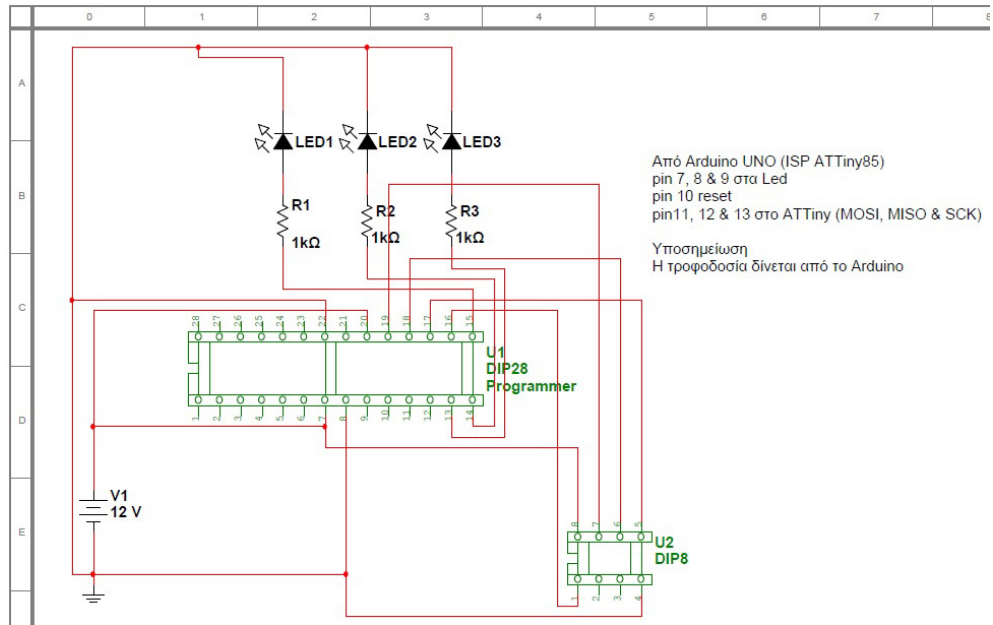
Το σχέδιο αποτελείται από δύο μέρη, όπως απεικονίζεται στην παραπάνω εικόνα. Το κάτω breadboard απεικονίζει τη συνδεσμολογία για τον προγραμματισμό του μικροελεγκτή και το επάνω breadboard απεικονίζει τη συνδεσμολογία για τον οπτικό έλεγχο της ανατροφοδότησης σχετικά με τη διαδικασία του προγραμματισμού [6].



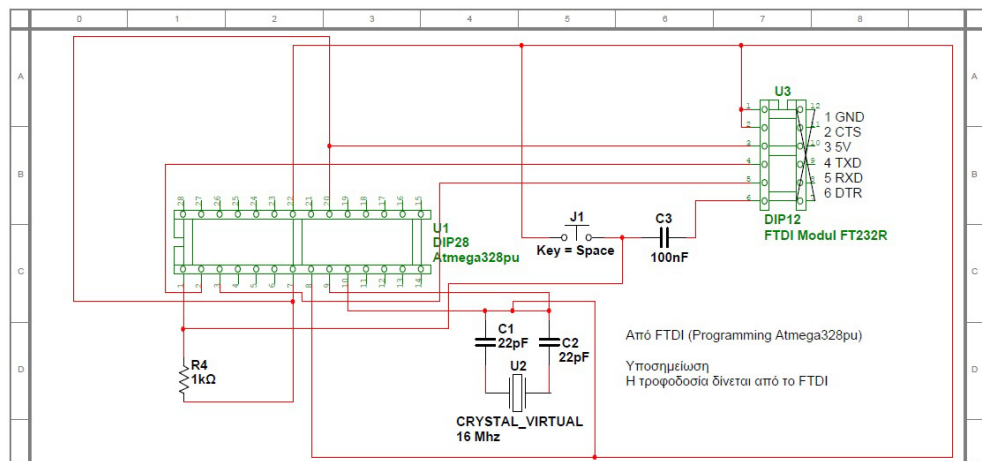
Εικόνα (8): Σχέδιο στο πρόγραμμα Multisim του μετασχηματιστή τάσης.



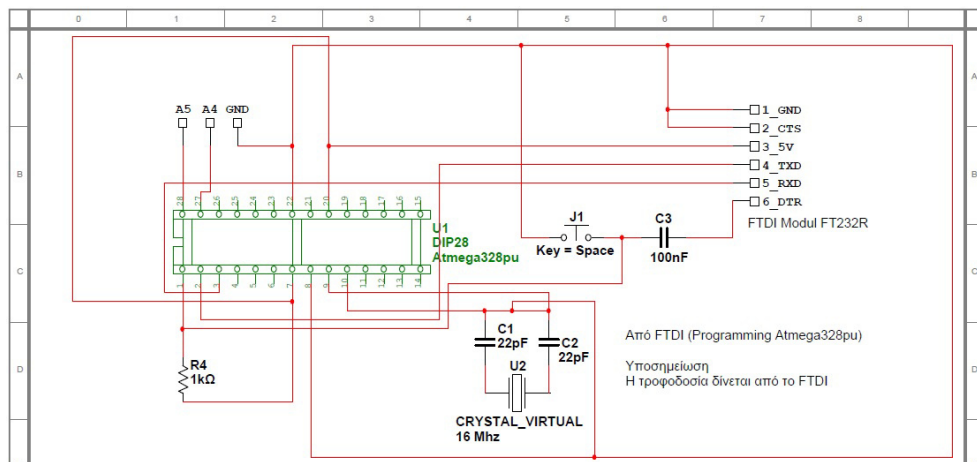
Εικόνα (9): Σχέδιο στο πρόγραμμα Multisim για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.



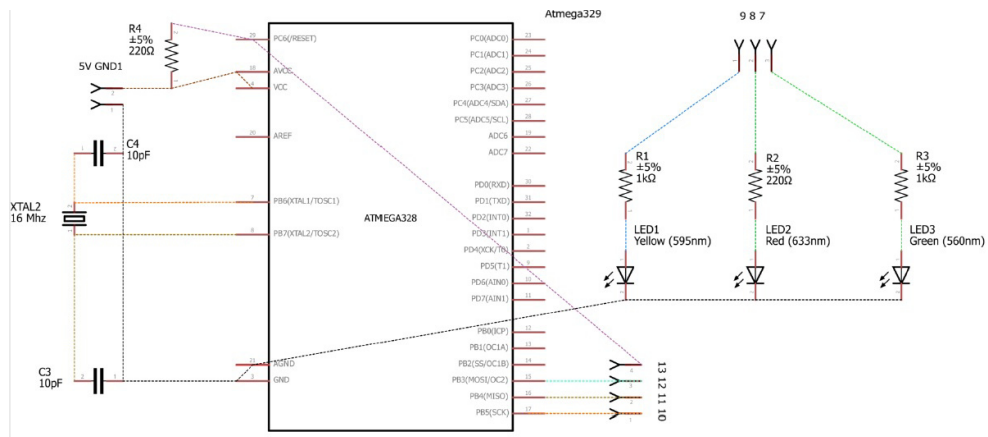
Εικόνα (10): Σχέδιο στο πρόγραμμα Multisim για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.



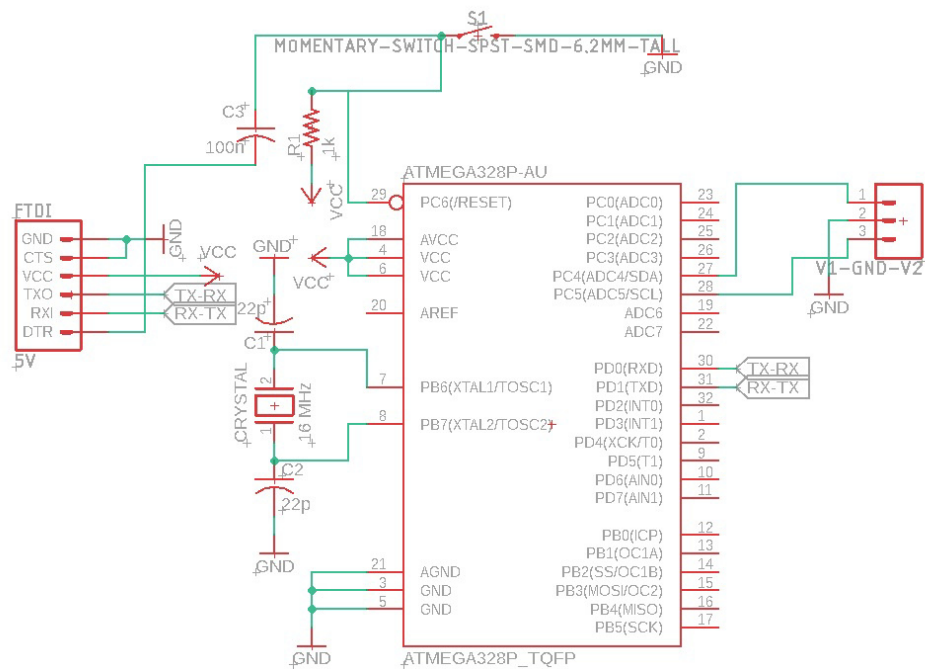
Εικόνα (11): Σχέδιο στο πρόγραμμα Multisim για προγραμματισμό μέσω FTDI Module (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU.



Εικόνα (12): Σχέδιο στο πρόγραμμα Multisim αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.



Εικόνα (13): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU.

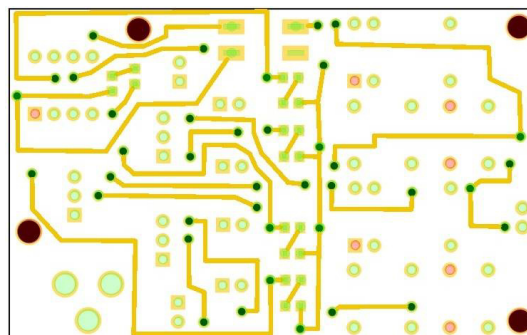


Εικόνα (14): Σχέδιο στο πρόγραμμα Eagle αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU.

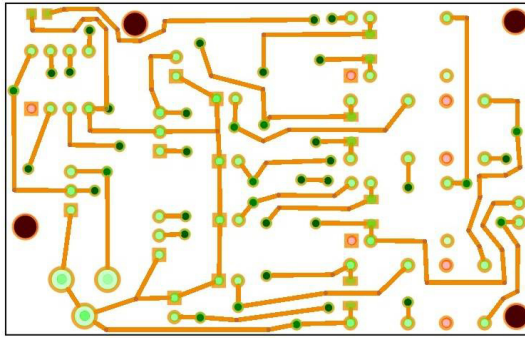
2.4 Σχεδίαση PCB

Επόμενο επίπεδο της ηλεκτρονικής σχεδίασης και έπειτα την υλοποίηση των σχεδίων σε breadboard και σε διάτρητη πλακέτα έχει η σχεδίαση των PCB πλακετών για την μόνιμη και σταθερή διαδικασία των παραπάνω σχεδίων.

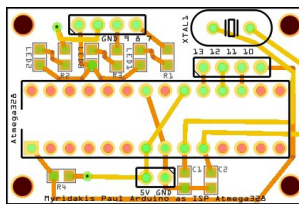
Πραγματοποιήθηκε η σχεδιάσή τους χρησιμοποιώντας τα προγράμματα Fritzing και Eagle, τα οποία και παραθέτονται ακολούθως:



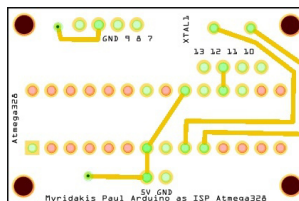
Εικόνα (15α): Σχέδιο στο πρόγραμμα Fritzing του μετασχηματιστή τάσης (επίπεδο 1).



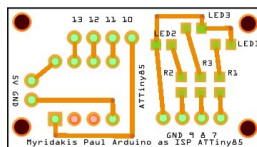
Εικόνα (15β): Σχέδιο στο πρόγραμμα Fritzing του μετασχηματιστή τάσης (επίπεδο 2).



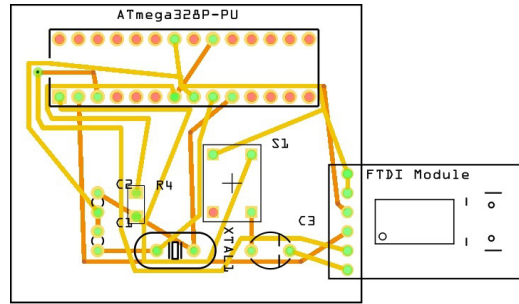
Εικόνα (16α): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU (επίπεδο 1).



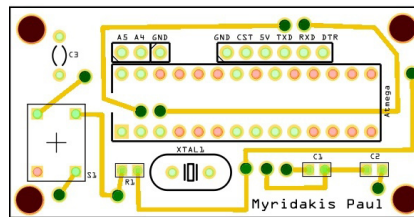
Εικόνα (16β): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU (επίπεδο 2).



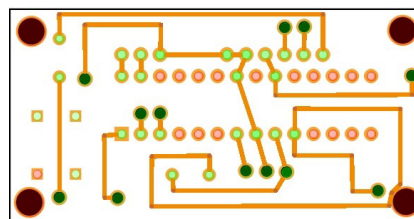
Εικόνα (17): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATTiny85-20PU.



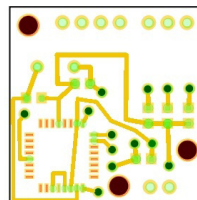
Εικόνα (18): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU (2 επιπέδων).



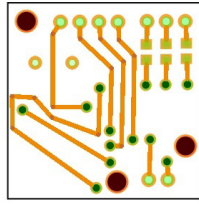
Εικόνα (19α): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU (επίπεδο 1).



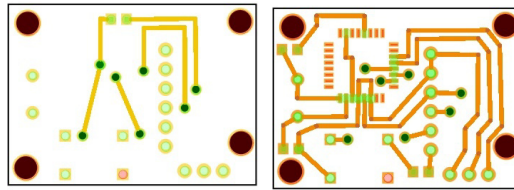
Εικόνα (19β): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU (επίπεδο 2).



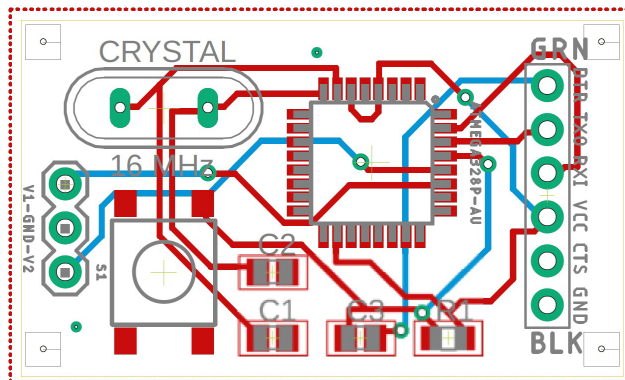
Εικόνα (20α): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU (επίπεδο 1).



Εικόνα (20β): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU (επίπεδο 2).



Εικόνα (21α): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU (2 επιπέδων).



Εικόνα (21β): Σχέδιο στο πρόγραμμα Eagle αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU (2 επιπέδων).

ΚΕΦΑΛΑΙΟ 3^ο

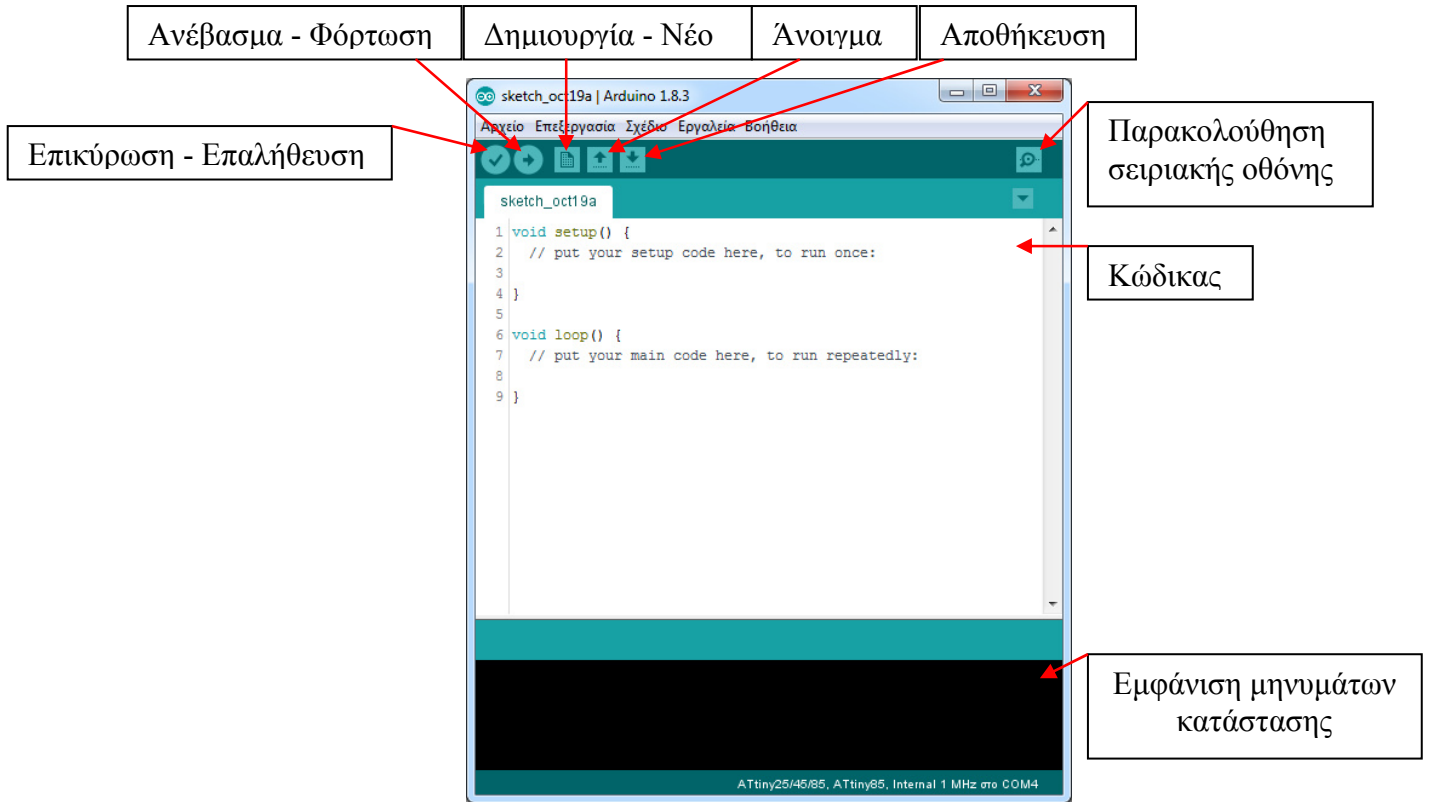
ΛΟΓΙΣΜΙΚΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

3.1 Εισαγωγή

Ως λογισμικό προγραμματισμού επιλέχθηκε το λογισμικό προγραμματισμού του Arduino. Το Arduino IDE, είναι γραμμένο σε γλώσσα Java, παρέχοντας δυνατότητα συμβατότητας σε πολλά λειτουργικά συστήματα. Διατίθεται δωρεάν από την επίσημη ιστοσελίδα του Arduino [6]. Η γλώσσα προγραμματισμού που μεταγλωττίζει ονομάζεται Wiring η οποία είναι παρόμοια της C / C++ με απλοποιήσεις και αλλαγές. Αναγνωρίζει την θύρα που συνδέεται η πλακέτα και μεταφέρει το πρόγραμμα μέσω σειριακής θύρας από τον υπολογιστή στην πλακέτα [6].

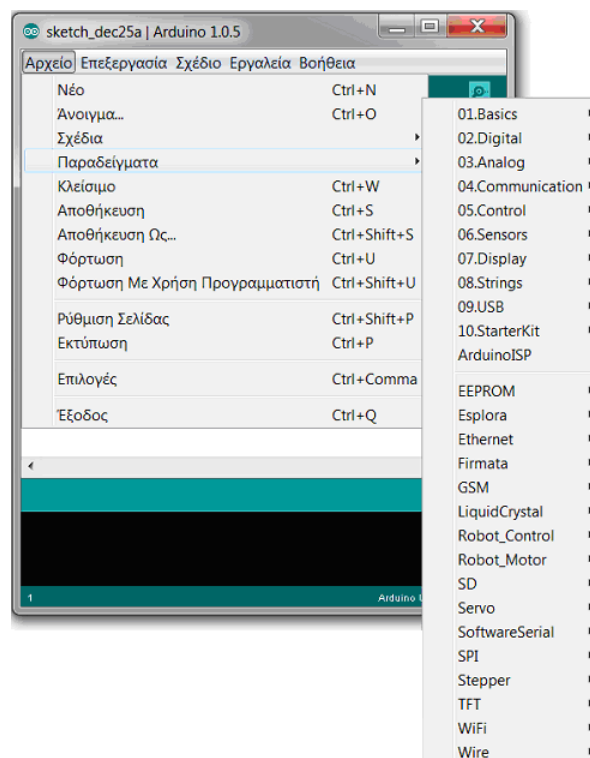
3.2 Βασικές λειτουργίες λογισμικού

Ένα περιβάλλον ανάπτυξης σχεδιασμένο για αρχάριους χρήστες, το Arduino IDE συντάσσεται με χρωματική σήμανση [6]. Στην εικόνα που ακολουθεί φαίνονται μερικές από τις άμεσες επιλογές του καθώς και τα κύρια μέρη του.



Εικόνα (22): Πρόγραμμα Arduino IDE.

Στην επόμενη εικόνα φαίνεται η επιλογή με τα έτοιμα παραδείγματα κώδικα σε Wiring που περιέχονται στην έκδοση του Arduino IDE.



Εικόνα (23): Έτοιμα παραδείγματα κώδικα σε Wiring.

3.3 Δομή προγράμματος

Κάθε σχέδιο (sketch) του arduino έχει την εξής δομή:

```
// δηλώσεις μεταβλητών και βιβλιοθηκών

void setup()

{

    // αρχικοποιήσεις

    // οι εντολές εδώ θα τρέξουν μόνο μία φορά στην ενεργοποίηση ή μετά από

    // κάποια επανεκκίνηση

}

void loop()

{

    // οι εντολές εδώ θα τρέχουν ξανά και ξανά, μέχρι την απενεργοποίηση ή

    // κάποια επανεκκίνηση

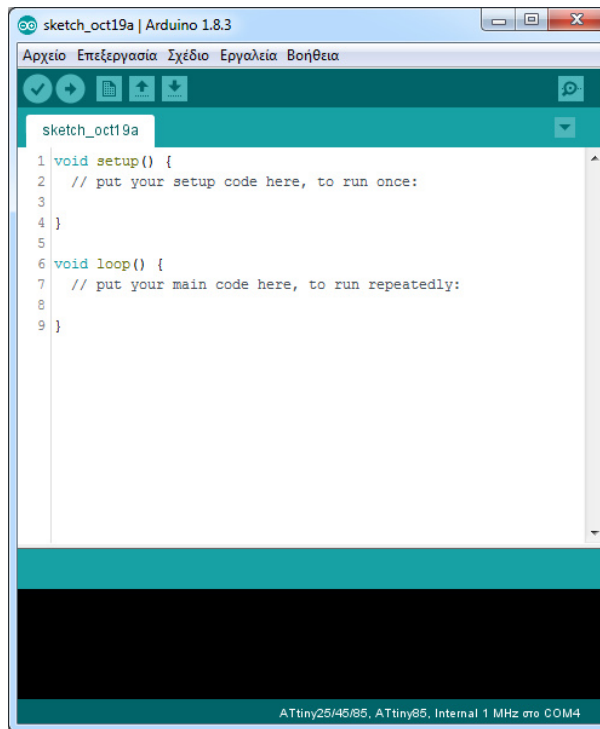
}

// Υπόλοιπες συναρτήσεις
```

Περιέχει τις δύο κύριες συναρτήσεις, την setup και την loop, όπως φαίνεται και στην παρακάτω εικόνα.

Η συνάρτηση setup() εκτελείται στην αρχή του προγράμματος για μία φορά όταν ενεργοποιείται ο μικροελεγκτής ή έπειτα από κάποια επανεκκίνησή του, περιλαμβάνει τις αρχικοποιήσεις των μεταβλητών, τις δηλώσεις των ακροδεκτών (αν ένας ακροδέκτης είναι εισόδου ή εξόδου) και τις αρχικοποιήσεις των βιβλιοθηκών.

Η συνάρτηση loop(), είναι ένας βρόγχος επανάληψης και ο κώδικας που γράφεται μέσα σε αυτή δίνει την δυνατότητα στο πρόγραμμα να αλλάζει τιμές και ο μικροελεγκτής να ανταποκρίνεται ανάλογα.



Εικόνα (24): Εισαγωγική απεικόνιση του προγράμματος Arduino IDE.

3.4 Συναρτήσεις και εντολές

Οι συναρτήσεις και εντολές, όπου μπορούν να χρησιμοποιηθούν στο πρόγραμμα σύνταξης Arduino IDE, είναι παρόμοιες με της γλώσσας C/ C++, με απλοποιήσεις, αλλαγές και εμπλουτισμένες για την διαχείριση του μικροελεγκτή.

Ορισμένες από αυτές τις συναρτήσεις παρατίθενται ακολούθως [6].

Συναρτήσεις & Εντολές	Είδος	Τύπος	Παράμετροι	Περιγραφή
LOW	Σταθερά	Int	-	Έχει την τιμή 0 και είναι αντίστοιχη του λογικού false.
HIGH	Σταθερά	Int	-	Έχει την τιμή 1 και είναι αντίστοιχη του λογικού true.
INPUT	Σταθερά	Int	-	Έχει την τιμή 0 και είναι αντίστοιχη του λογικού false.
OUTPUT	Σταθερά	Int	-	Έχει την τιμή 1 και είναι αντίστοιχη του λογικού true.
pinMode	Εντολή	-	(pin, mode)	Καθορίζει αν το συγκεκριμένο ψηφιακό pin θα είναι εισόδου ή εξόδου ανάλογα με την τιμή που δίνεται στην παράμετρο mode (INPUT ή OUTPUT).

Πίνακας (3α): Συναρτήσεις και εντολές για το πρόγραμμα Arduino IDE.

Συναρτήσεις & Εντολές	Είδος	Τύπος	Παράμετροι	Περιγραφή
digitalWrite	Εντολή	-	(pin, pinstatus)	Θέτει την κατάσταση pinstatus (LOW ή HIGH) στο συγκεκριμένο ψηφιακό pin.
digitalRead	Συνάρτηση	Int	(pin)	Επιστρέφει την κατάσταση του συγκεκριμένου ψηφιακού pin (0 για LOW και 1 για HIGH) με προϋπόθεση αυτό είναι εισόδου.
analogReference	Εντολή	-	(type)	Δέχεται τις τιμές DEFAULT, INTERNAL ή EXTERNAL στην παράμετρο type για να καθορίσει την τάση αναφοράς (V_{ref}) των αναλογικών εισόδων (5V, 1.1V ή η εξωτερική τάση με την οποία τροφοδοτείται το pin AREF αντίστοιχα)
analogRead	Συνάρτηση	int	(pin)	Επιστρέφει έναν ακέραιο από 0 έως 1023, ανάλογα με την τάση που τροφοδοτείται το συγκεκριμένο pin αναλογικής εισόδου στην κλίμακα 0 ως V_{ref} .
analogWrite	Εντολή	-	(pin, value)	Θέτει το συγκεκριμένο ψηφιακό pin σε κατάσταση ψευδο αναλογικής εξόδου (PWM). Η παράμετρος value καθορίζει το πλάτος του παλμού σε σχέση με την περίοδο του παραγόμενου σήματος στην κλίμακα από 0 ως 255 (π.χ. με value 127, το πλάτος του παλμού είναι ίσο με μισή περίοδο).
millis	Συνάρτηση	unsigned long	()	Μετρητής που επιστρέφει το χρονικό διάστημα σε ms από την στιγμή που άρχισε η εκτέλεση του προγράμματος. Λάβετε υπόψη ότι λόγω του τύπου μεταβλητής (unsigned long δηλ. 32bit) θα γίνει overflow σε 2^{32} ms δηλαδή περίπου σε 50 μέρες, οπότε ο μετρητής θα ξεκινήσει πάλι από το μηδέν.
delay	Εντολή	-	(time)	Σταματά προσωρινά την ροή του προγράμματος για time ms. Η παράμετρος time είναι unsigned long (από 0 ως 2^{32}). Σημειώστε ότι παρά την προσωρινή παύση, συναρτήσεις των οποίων η εκτέλεση ενεργοποιείται από interrupt θα εκτελεστούν κανονικά κατά την διάρκεια μιας καθυστέρησης (delay).

Πίνακας (3β): Συναρτήσεις και εντολές για το πρόγραμμα Arduino IDE.

Συναρτήσεις & Εντολές	Είδος	Τύπος	Παράμετροι	Περιγραφή
attachInterrupt	Εντολή	-	(interrupt, function, triggermode)	Θέτει σε λειτουργία το συγκεκριμένο interrupt, ώστε να ενεργοποιεί την συνάρτηση (function), κάθε φορά που ικανοποιείται η συνθήκη που ορίζεται από την παράμετρο triggermode: 1) LOW (ενεργοποίηση όταν η κατάσταση του pin που αντιστοιχεί στο συγκεκριμένο interrupt γίνει LOW) 2) RISING (όταν από LOW γίνει HIGH) 3) FALLING (όταν από HIGH γίνει LOW) 4) CHANGE (όταν αλλάξει κατάσταση γενικά)
detachInterrupt	Εντολή	-	(interrupt)	Απενεργοποιεί το συγκεκριμένο interrupt.
noInterrupts	Εντολή	-	()	Σταματά προσωρινά την λειτουργία όλων των interrupt
interrupts	Εντολή	-	()	Επαναφέρει την λειτουργία των interrupt που διακόπηκε προσωρινά από μια εντολή noInterrupts.
Serial.begin	Μέθοδος κλάσης	-	(datarate)	Θέτει τον ρυθμό μεταφοράς δεδομένων του σειριακού interface (σε baud)
Serial.println	Μέθοδος κλάσης	-	(data)	Διοχετεύει τα δεδομένα data για αποστολή μέσω του σειριακού interface. Η παράμετρος data μπορεί να είναι είτε αριθμός είτε αλφαριθμητικό.

Πίνακας (3γ): Συναρτήσεις και εντολές για το πρόγραμμα Arduino IDE.

Παρακάτω παρατίθεται ένα μικρό παράδειγμα για τη χρήση των εντολών.

```
// Δηλώνεται η μεταβλητή led ως integer (Pin 13)

int led = 13;

void setup()

{

// Αρχικοποιήσεις

// Δηλώνεται το LED ως έξοδος

pinMode(led, OUTPUT);
```

```

    }

void loop()

{

// Βρόγχος επανάληψης

// Δηλώνεται το LED ως HIGH

digitalWrite(led, HIGH);

// Αναμονή 1 δευτερολέπτου

delay(1000);

// Δηλώνεται το LED ως LOW

digitalWrite(led, LOW);

// Αναμονή 1 δευτερολέπτου

delay(1000);

}

```

Στο παραπάνω παράδειγμα το LED του μικροελεγκτή θα αναβοσβήνει κάθε ένα δευτερόλεπτο μέχρι να το αποσυνδέσουμε.

3.5 In System Programming (ISP)

Με τη διαδικασία In System Programming μπορούμε να φορτώσουμε προγράμματα σε έναν «κενό» μικροελεγκτή μέσω ενός άλλου μικροελεγκτή ATmega328, να τον προγραμματίσουμε και επαναπρογραμματίσουμε. Με τη φόρτωση του κώδικα bootloader, ένα μικρό κώδικα ο οποίος αποθηκεύεται στη μνήμη του μικροελεγκτή, επιτρέπουμε να δουλεύει ένας μικροελεγκτής ATmega ως «Arduino», διαχειριζόμενος από το λογισμικό Arduino (IDE). Το «Arduino as ISP» είναι η φθηνότερη και πιο πρακτικότερη λύση για να προγραμματίσουμε έναν άλλο μικροελεγκτή ATmega ή ATtiny και να φορτώσουμε το bootloader σε μικροελεγκτή ATmega [6, 7].

Αυτή τη διαδικασία την ακολουθούμε και την εκτελούμε μία φορά για κάθε έναν καινούριο μικροελεγκτή ATmega, διότι πιθανόν να είναι τελείως κενός, χωρίς να είναι φορτωμένο το bootloader. Η διαδικασία προγραμματισμού χρησιμοποιεί στο σύνολο έξι (6) ακροδέκτες τροφοδοσίας, δεδομένων και λειτουργικότητας (VCC, GND, MOSI, MISO, SCK και RESET). Οι ακροδέκτες MOSI, MISO και SCK είναι οι ίδιοι με τους ψηφιακούς ακροδέκτες 11, 12 και 13, αντίστοιχα [6, 7]. Στην οικογένεια MKR, δεν μπορούμε να χρησιμοποιήσετε το D10 για επανεκκίνηση, προτείνεται το D6 και πρέπει να αλλάξουμε τη γραμμή 73 του σκίτσου ArduinoISP - #define RESET 10 - βάζοντας το «6» στη θέση του «10» [6].

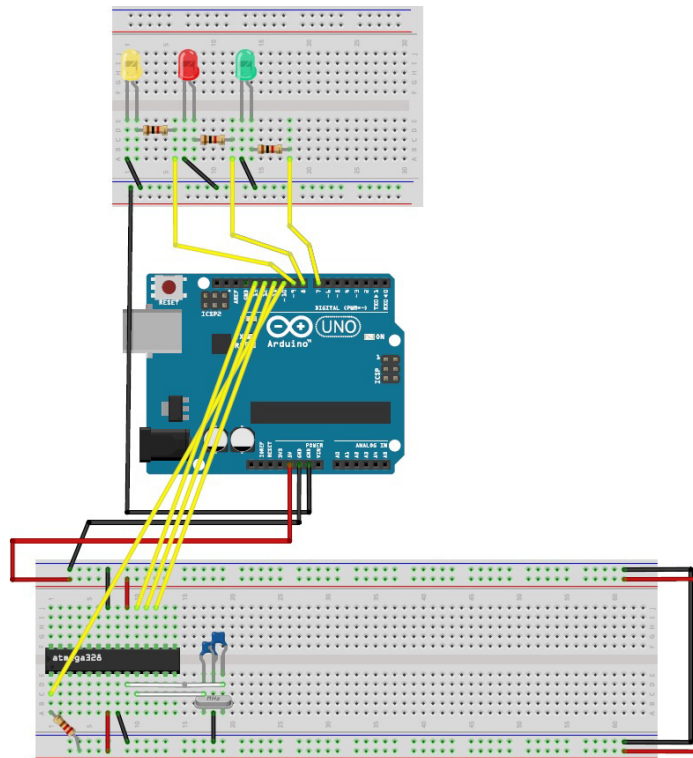
3.5.1 Φόρτωση μέσω ISP σε μικροελεγκτή ATmega328

3.5.1.1 Φόρτωση bootloader μέσω ISP σε μικροελεγκτή ATmega328

Για να πραγματοποιήσουμε τη φόρτωση του bootloader μέσω ISP σε μικροελεγκτή ATmega328, ακολουθούμε τα παρακάτω βήματα [6, 7]:

- Συνδέουμε την πλακέτα του Arduino Uno (προγραμματιστής) στον υπολογιστή.
- Εκτελούμε το πρόγραμμα Arduino IDE.
- Επιλέγουμε από το Μενού επιλογών, με τα έτοιμα παραδείγματα κώδικα που περιέχονται στην έκδοση του Arduino IDE, όπως η εικόνα (23): Αρχείο/Παραδείγματα/ArduinoISP/ArduinoISP.
- Επεξεργαζόμαστε το αρχείο στα σημεία όπου χρειάζεται:
 - Ρύθμιση ακροδέκτη επανεκκίνησης.
 - Ρύθμιση ρολογιού SPI (σε Hz).
 - Ρύθμιση ρυθμού μετάδοσης.
 - Ενεργοποίηση παλαιού στυλ.
- Από το Μενού επιλογών, επιλέγουμε την πλακέτα: Εργαλεία/Πλακέτα/Arduino Uno
- Από το Μενού επιλογών, επιλέγουμε τη θύρα επικοινωνίας: Εργαλεία/Θύρα/COM ...

- Πραγματοποιούμε ανέβασμα (φόρτωση) του αρχείου από τις άμεσες επιλογές του προγράμματος (εικόνα 22).
- Αποσυνδέουμε τον προγραμματιστή.
- Συνδέουμε τον προγραμματιστή με τον «κενό» μικροελεγκτή ATmega328 (προγραμματιζόμενο), όπως η ακόλουθη εικόνα.



Εικόνα (25): Σύνδεση του προγραμματιστή με τον «κενό» μικροελεγκτή ATmega328.

Στον προγραμματιστή χρησιμοποιούμε τις ακίδες 7, 8, 9 και μία γείωση, για τη σύνδεση με το επάνω breadboard, για τον οπτικό έλεγχο της ανατροφοδότησης σχετικά με τη διαδικασία του προγραμματισμού και τις ακίδες 10, 11, 12, 13, γείωση και μία τάση (τροφοδοσία 5V) για τη σύνδεση με το κάτω breadboard, για τον προγραμματισμό του «κενού» μικροελεγκτή [6, 7].

- Συνδέουμε τον προγραμματιστή στον υπολογιστή.
- Από το Μενού επιλογών, επιλέγουμε τη θύρα επικοινωνίας: Εργαλεία/Θύρα/COM ...
- Από το Μενού επιλογών, επιλέγουμε τον προγραμματιστή: Εργαλεία/Προγραμματιστής/Arduino as ISP

- Από το Μενού επιλογών, επιλέγουμε εγγραφή bootloader: Εργαλεία/Γράψιμο Bootloader.

Σημείωση: Η διαδικασία διαρκεί λίγα δευτερόλεπτα και στο τέλος θα μας εμφανίσει ένα μήνυμα για την επιτυχή ολοκλήρωση της διαδικασίας.

3.5.1.2 Φόρτωση προγράμματος μέσω ISP σε μικροελεγκτή ATmega328

Η διαδικασία είναι παρόμοια με την προηγούμενη παράγραφο και η συνδεσμολογία ίδια, οπότε σε αυτή την περίπτωση ακολουθούμε τα βήματα:

- Συνδέουμε την πλακέτα του Arduino Uno (προγραμματιστής) με τον «κενό» μικροελεγκτή ATmega328 (προγραμματιζόμενο), όπως η προηγούμενη εικόνα.

Στον προγραμματιστή χρησιμοποιούμε τις ακίδες 7, 8, 9 και μία γείωση, για τη σύνδεση με το επάνω breadboard, για τον οπτικό έλεγχο της ανατροφοδότησης σχετικά με τη διαδικασία του προγραμματισμού και τις ακίδες 10, 11, 12, 13, γείωση και μία τάση (τροφοδοσία 5V) για τη σύνδεση με το κάτω breadboard, για τον προγραμματισμό του «κενού» μικροελεγκτή [6, 7].

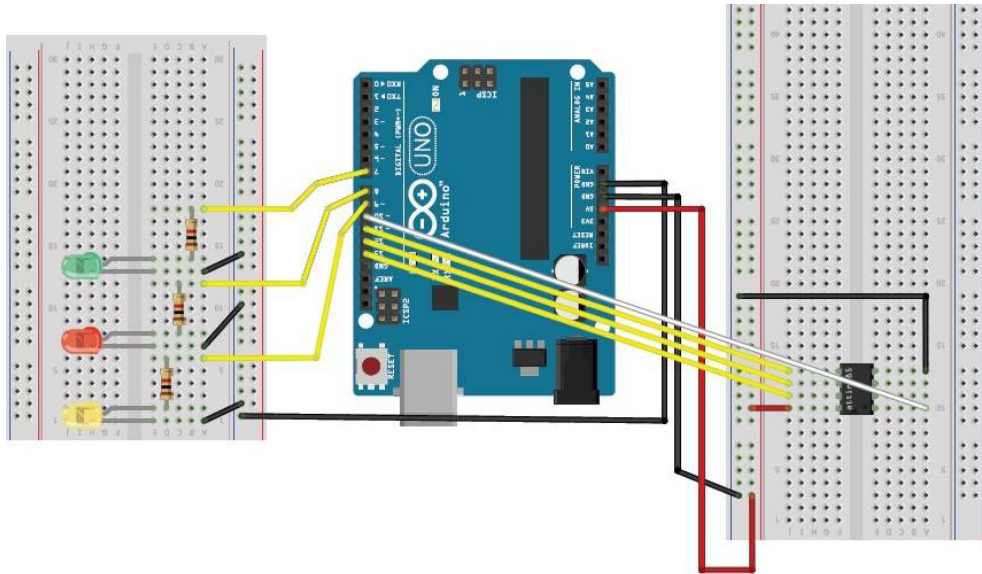
- Συνδέουμε τον προγραμματιστή στον υπολογιστή.
- Εκτελούμε το πρόγραμμα Arduino IDE.
- Επιλέγουμε ή δημιουργούμε το πρόγραμμα το οποίο επιθυμούμε να φορτώσουμε.
- Επεξεργαζόμαστε το αρχείο εάν απαιτείτε.
- Από το Μενού επιλογών, επιλέγουμε την πλακέτα: Εργαλεία/Πλακέτα/Arduino Uno
- Από το Μενού επιλογών, επιλέγουμε τη θύρα επικοινωνίας: Εργαλεία/Θύρα/COM ...
- Από το Μενού επιλογών, επιλέγουμε τον προγραμματιστή: Εργαλεία/Προγραμματιστής/Arduino as ISP
- Πραγματοποιούμε ανέβασμα (φόρτωση) του αρχείου μέσω προγραμματιστή, επιλέγουμε από το Μενού επιλογών: Σχέδιο/Ανέβασμα μέσω προγραμματιστή.

- Η διαδικασία διαρκεί λίγα δευτερόλεπτα και στο τέλος θα μας εμφανίσει ένα μήνυμα για την επιτυχή ολοκλήρωση της διαδικασίας.
- Αποσυνδέουμε τον προγραμματιστή.

3.5.2 Φόρτωση προγράμματος μέσω ISP σε μικροελεγκτή ATtiny85-20PU

Για να πραγματοποιήσουμε τη φόρτωση προγράμματος μέσω ISP σε μικροελεγκτή ATtiny85-20PU, ακολουθούμε τα παρακάτω βήματα:

- Συνδέουμε την πλακέτα του Arduino Uno (προγραμματιστής) στον υπολογιστή.
- Εκτελούμε το πρόγραμμα Arduino IDE.
- Επιλέγουμε από το Μενού επιλογών, με τα έτοιμα παραδείγματα κώδικα που περιέχονται στην έκδοση του Arduino IDE, όπως η εικόνα (23): Αρχείο/Παραδείγματα/ArduinoISP/ ArduinoISP.
- Επεξεργαζόμαστε το αρχείο στα σημεία όπου απαιτείται:
 - Ρύθμιση ακροδέκτη επανεκκίνησης.
 - Ρύθμιση ρολογιού SPI (σε Hz).
 - Ρύθμιση ρυθμού μετάδοσης.
 - Ενεργοποίηση παλαιού στυλ.
- Από το Μενού επιλογών, επιλέγουμε την πλακέτα: Εργαλεία/Πλακέτα/Arduino Uno
- Από το Μενού επιλογών, επιλέγουμε τη θύρα επικοινωνίας: Εργαλεία/Θύρα/COM ...
- Πραγματοποιούμε ανέβασμα (φόρτωση) του αρχείου από τις άμεσες επιλογές του προγράμματος (εικόνα 22).
- Αποσυνδέουμε τον προγραμματιστή.
- Συνδέουμε τον προγραμματιστή με τον «κενό» μικροελεγκτή ATtiny85-20PU (προγραμματιζόμενο), όπως η ακόλουθη εικόνα.



Εικόνα (26): Σύνδεση του προγραμματιστή με τον «κενό» μικροελεγκτή ATtiny85-20PU.

Στον προγραμματιστή χρησιμοποιούμε τις ακίδες 7, 8, 9 και μία γείωση, για τη σύνδεση με το αριστερό breadboard, για τον οπτικό έλεγχο της ανατροφοδότησης σχετικά με τη διαδικασία του προγραμματισμού και τις ακίδες 10, 11, 12, 13, γείωση και μία τάση (τροφοδοσία 5V) για τη σύνδεση με το δεξί breadboard, για τον προγραμματισμό του «κενού» μικροελεγκτή [6, 7].

- Συνδέουμε τον προγραμματιστή στον υπολογιστή.
- Επιλέγουμε ή δημιουργούμε το πρόγραμμα το οποίο επιθυμούμε να φορτώσουμε.
- Επεξεργαζόμαστε το αρχείο εάν απαιτείτε.
- Από το Μενού επιλογών, επιλέγουμε την πλακέτα: Εργαλεία/Πλακέτα/ATtiny25/45/85.
- Διαπιστώνουμε ότι μας έδωσε και άλλες επιλογές. Από το Μενού επιλογών, επιλέγουμε τον επεξεργαστή: Εργαλεία/Επεξεργαστής/ATtiny85.
- Από το Μενού επιλογών, επιλέγουμε την συχνότητα του ρολογιού: Εργαλεία/Clock/Internal 1MHz.
- Από το Μενού επιλογών, επιλέγουμε τη θύρα επικοινωνίας: Εργαλεία/Θύρα/COM ...
- Από το Μενού επιλογών, επιλέγουμε τον προγραμματιστή: Εργαλεία/Προγραμματιστής/Arduino as ISP.

- Πραγματοποιούμε ανέβασμα (φόρτωση) του αρχείου μέσω προγραμματιστή, επιλέγουμε από το Μενού επιλογών: Σχέδιο/Ανέβασμα μέσω προγραμματιστή.

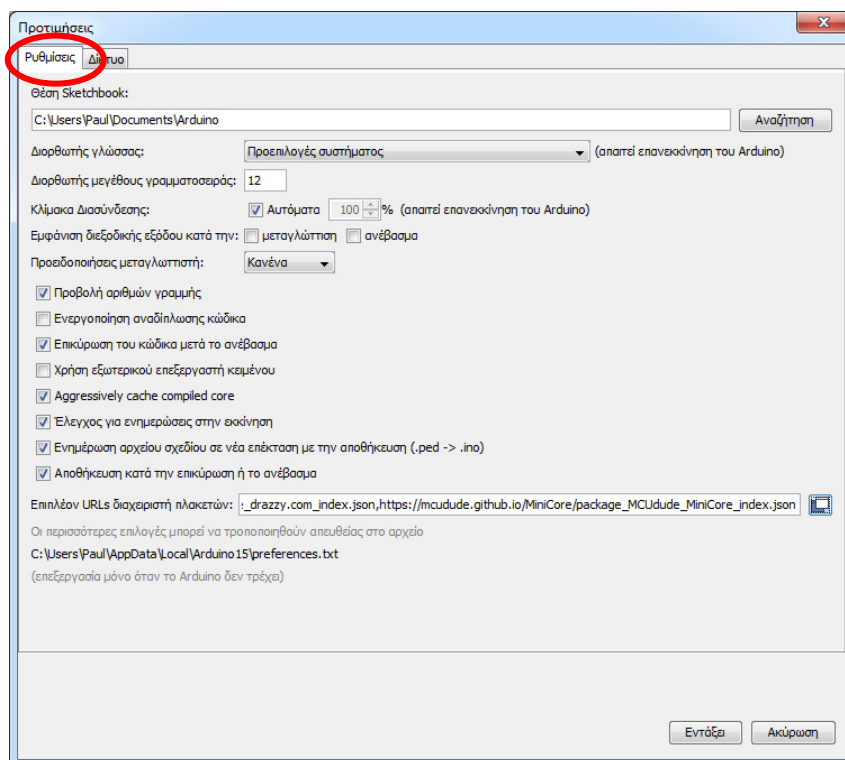
Η διαδικασία διαρκεί λίγα δευτερόλεπτα και στο τέλος θα μας εμφανίσει ένα μήνυμα για την επιτυχή ολοκλήρωση της διαδικασίας.

- Αποσυνδέουμε τον προγραμματιστή.

Σημείωση:

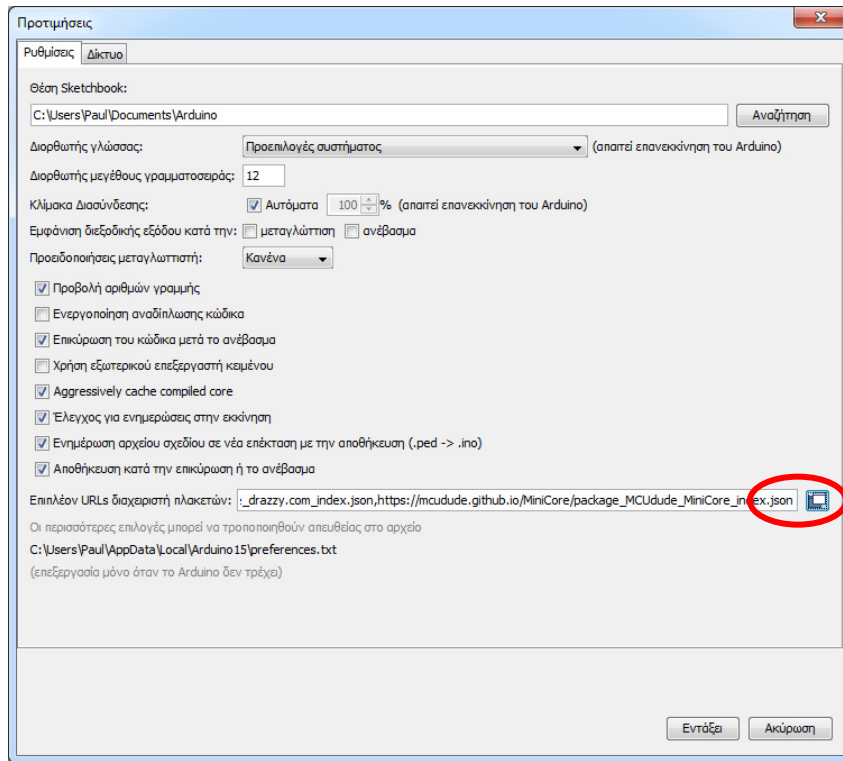
Σε περίπτωση που δεν έχουμε την επιλογή από το Μενού επιλογών, του μικροελεγκτή «ATtiny25/45/85» (Εργαλεία/Πλακέτα/ATtiny25/45/85), τότε πραγματοποιούμε την ακόλουθη διαδικασία:

- Από το Μενού επιλογών, επιλέγουμε τις Προτιμήσεις: Αρχείο/ Προτιμήσεις.
- Στο παράθυρο που ανοίγει, ακόλουθη εικόνα, επιλέγουμε την καρτέλα «Ρυθμίσεις».



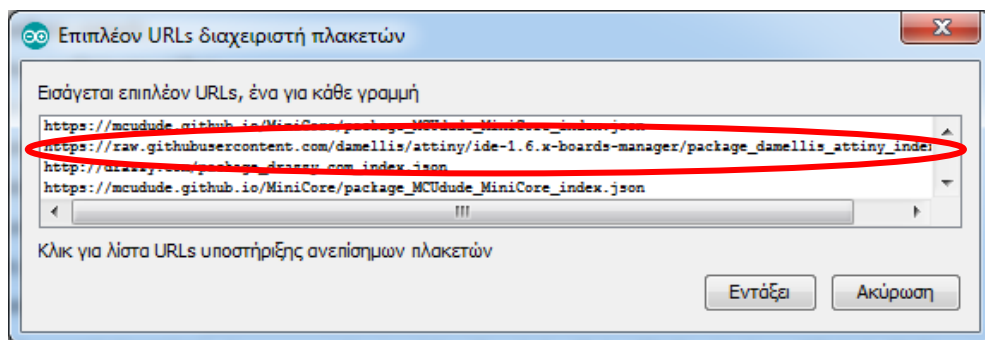
Εικόνα (27α): Μενού επιλογών: Αρχείο/ Προτιμήσεις καρτέλα «Ρυθμίσεις».

- Ακολούθως την επιλογή «Επιπλέον URLs διαχειριστή πλακετών», παρακάτω εικόνα.



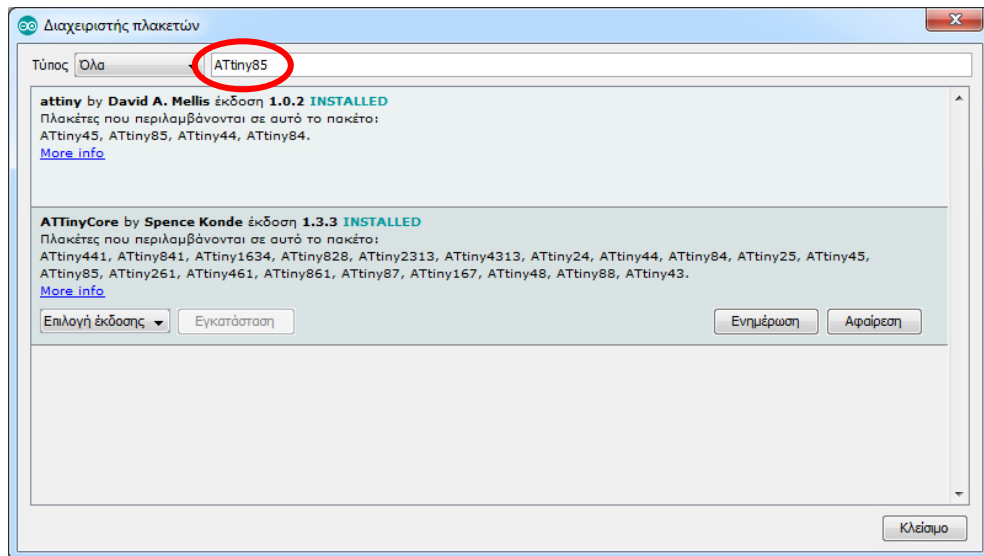
Εικόνα (27β): Μενού επιλογών: Αρχείο/ Προτιμήσεις καρτέλα «Ρυθμίσεις».

- Στο ανοιγόμενο παράθυρο, όπως ακόλουθη εικόνα, προσθέτουμε την ακόλουθη διεύθυνση στο πλαίσιο κειμένου: «https://raw.githubusercontent.com/damellis/attiny/ide-1.6.x-boards-manager/package_damellis_attiny_index.json» [7].



Εικόνα (28): Επιπλέον URLs διαχειριστή πλακετών.

- Από το Μενού επιλογών, επιλέγουμε την πλακέτα: Εργαλεία/Πλακέτα/Διαχειριστής πλακετών...
- Στο παράθυρο που ανοίγει, όπως στην ακόλουθη εικόνα, πληκτρολογούμε στην αναζήτησή του «ATtiny85».



Εικόνα (29): Διαχειριστής πλακετών προγράμματος Arduino IDE.

- Ακολουθώς προχωράμε στην εγκατάσταση των επιθυμητών.

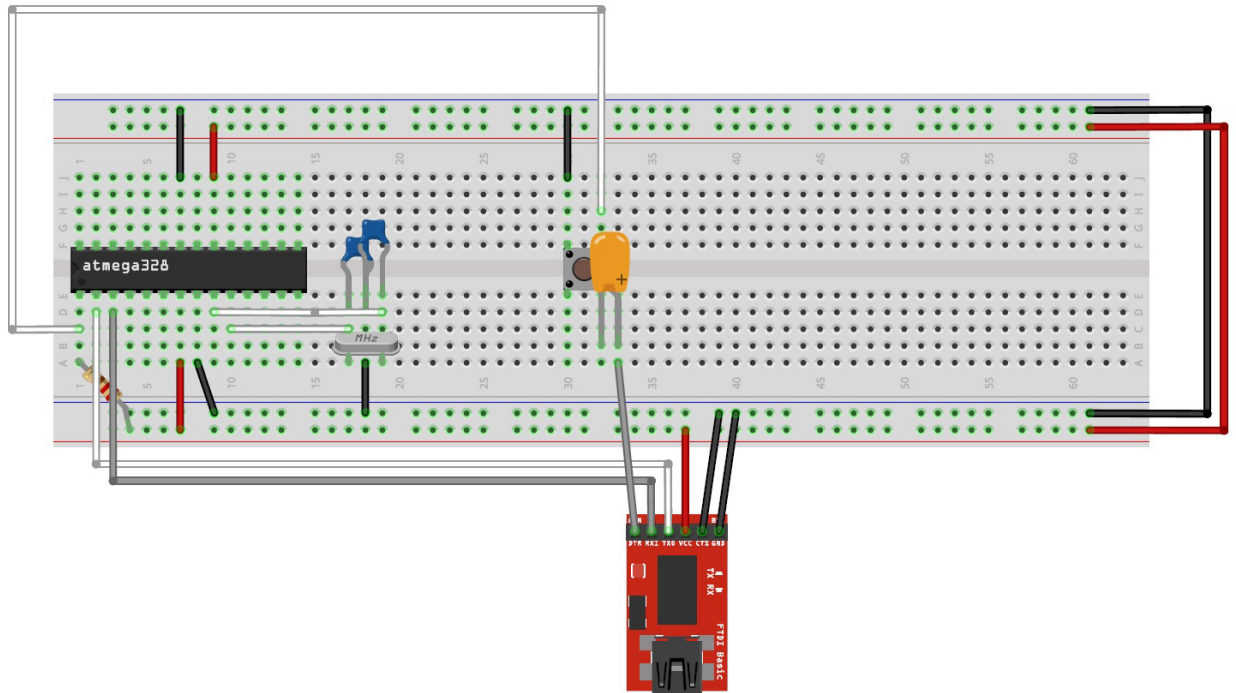
3.6 Φόρτωση προγράμματος μέσω FTDI (σειριακής θύρας) σε μικροελεγκτή ATmega328

Το FTDI module είναι ένα ολοκληρωμένο κύκλωμα για επικοινωνία, το οποίο επιτυγχάνει τη μετατροπή από το USB σε σειριακό. Από τη μία πλευρά συνδέεται με USB καλώδιο και από την άλλη διαθέτει έξι ακροδέκτες λειτουργίας και επικοινωνίας [7, 23].

Για να πραγματοποιήσουμε τη φόρτωση προγράμματος μέσω FTDI σε μικροελεγκτή ATmega328, ακολουθούμε τα παρακάτω βήματα:

Θα πρέπει να έχουμε φορτώσει επιτυχώς το bootloader στον μικροελεγκτή μέσω της διαδικασίας του ISP και δεν θα έχουμε φορτώσει ακολούθως κάποιο άλλο πρόγραμμα με την ίδια διαδικασία στον ίδιο μικροελεγκτή, διότι είναι πιθανό να μας σβήσει το bootloader [7].

- Συνδέουμε το κύκλωμα του μικροελεγκτή ATmega328 με το FTDI, όπως την ακόλουθη εικόνα.



Εικόνα (30): Σύνδεση του μικροελεγκτή ATmega328 με FTDI module.

Οι ακροδέκτες του FTDI module συνδέονται ως εξής:

- DTR με τον πυκνωτή για το κουμπί της επανεκκίνησης.
- RXD με τον ακροδέκτη του μικροελεγκτή ATmega328 TX.
- TXD με τον ακροδέκτη του μικροελεγκτή ATmega328 RX.
- 5V με την τροφοδοσία του κυκλώματος μικροελεγκτή ATmega328.
- CTS και GND με τη γείωση του κυκλώματος μικροελεγκτή ATmega328.

Να σημειωθεί ότι σε ορισμένα FTDI module υπάρχει ένας διακόπτης επιλογής των 3,3V και 5V, όπως και στη δικιά μας περίπτωση, τον οποίο και θέτουμε στη θέση 5V.

- Ακολούθως, συνδέουμε το FTDI module, μέσω ενός USB καλωδίου, με τον υπολογιστή.
- Εκτελούμε το πρόγραμμα Arduino IDE.

- Επιλέγουμε από το Μενού επιλογών, είτε καινούριο, είτε κάποιο από τα έτοιμα παραδείγματα κώδικα που περιέχονται στην έκδοση του Arduino IDE, είτε να ανοίξουμε κάποιο το οποίο είχαμε δημιουργήσει παλαιότερα.
- Επεξεργαζόμαστε το αρχείο στα σημεία όπου χρειάζεται.
- Από το Μενού επιλογών, επιλέγουμε την πλακέτα: Εργαλεία/Πλακέτα/Arduino Uno
- Από το Μενού επιλογών, επιλέγουμε τη θύρα επικοινωνίας: Εργαλεία/Θύρα/COM ...
- Πραγματοποιούμε ανέβασμα (φόρτωση) του αρχείου από τις άμεσες επιλογές του προγράμματος (εικόνα 22).
- Αποσυνδέουμε το καλώδιο από το FTDI module.

Ο μικροελεγκτής ATmega328 έχει πλέον φορτωμένο το πρόγραμμα και συνδέοντάς τον στο αντίστοιχο κύκλωμα με τροφοδοσία και γείωση θα εκτελέσει ότι του φορτώσαμε στην τελευταία διαδικασία.

Εάν θέλουμε να επαναπρογραμματίσουμε τον ίδιο μικροελεγκτή ακολουθούμε την ίδια διαδικασία της παραγράφου.

ΚΕΦΑΛΑΙΟ 4^ο

ΥΛΟΠΟΙΗΣΗ ΚΑΤΑΣΚΕΥΗΣ

4.1 Εισαγωγή

Η παραγωγή των τσιπ σήμερα βασίζεται στην φωτολιθογραφία, όπου υπεριώδης ακτινοβολία (ακτίνες UV) εκπέμπεται περνώντας μέσα από μια αποτυπωμένη διαφάνεια (μάσκα) σε ένα κομμάτι πυρίτιο που είναι καλυμμένο με ένα φωτοευαίσθητο film [3].

Η αποτυπωμένη διαφάνεια περιγράφει τα στοιχεία του τσιπ και έτσι οι ακτίνες UV κτυπούν περιοχές που δεν καλύπτονται από αυτή. Όταν εμφανιστεί το film οι περιοχές που δεν κτυπήθηκαν από το φως αφαιρούνται, με αποτέλεσμα που μορφοποιεί το σχέδιο. Ακολούθως οι μη προστατευόμενες περιοχές επεξεργάζονται κατάλληλα, έτσι ώστε να αλλάξουν οι ηλεκτρικές τους ιδιότητες. Για τα πολυεπίπεδα (πολυστρωματικά) προστίθεται ένα νέο στρώμα υλικού και η διαδικασία επαναλαμβάνεται για να κατασκευαστεί το κύκλωμα. Όταν το κύκλωμα ολοκληρωθεί, προστίθεται το τελευταίο στρώμα από μέταλλο και επαναλαμβάνεται η ίδια διαδικασία όπου η αποτυπωμένη διαφάνεια έχει ως σχέδιο τις ενώσεις από τα διάφορα στοιχεία του τσιπ. Στη συνέχεια εμφανίζεται το φιλμ και αφαιρούνται τα κομμάτια που δεν είχαν εκτεθεί στην ακτινοβολία. Ακολούθως, αφαιρούνται τα κομμάτια μετάλλου που δεν προστατεύτηκαν από το film και έτσι τα σύρματα παίρνουν μορφή και τέλος το τσιπ ελέγχεται [3].

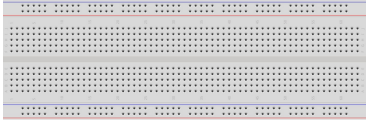








4.2 Ψηφιακός μετασχηματιστής τάσης

Για τον ψηφιακό μετασχηματιστή τάσης, οι οδηγίες και περιορισμοί της κατασκευής ήταν να λαμβάνει ως τάση εισόδου 12 volt και να μας παρέχει τη δυνατότητα επιλογής εξόδου σε 3,3 - 5 - 9 και 12 volt.

4.2.1 Σε breadboard

4.2.1.1 Υλικά κατασκευής

Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά με βάση τις ανάγκες της εργασίας, όπου και παραθέτονται ακολούθως [11, 21, 22]:

Breadboard	
Dc Power Jack	
Electrolytic Capacitor 25V 47uF	
Voltage Regulator LD1117 - 3.3V 800mA	
Voltage Regulator L78S05CV - 5V 2A	
Voltage Regulator L7809CV - 9V 1.5A	
LED Green 5mm THT	
LED Yellow 5mm THT	
Resistor THT 1kΩ και 220Ω	
Diodes 1N4007	
Tact Switch	

Πίνακας (4α): Υλικά που χρησιμοποιήθηκαν στο breadboard.

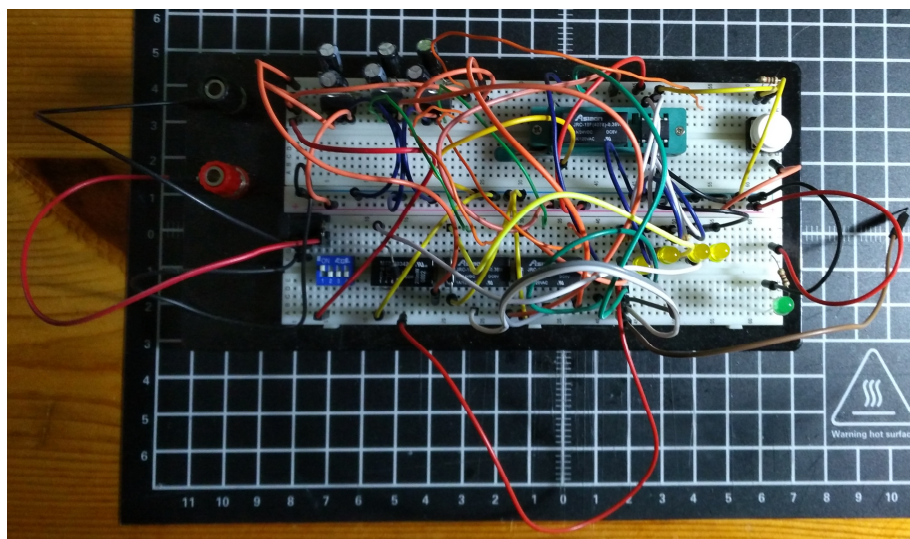
Cap for Tact Button	
ZIF Socket 28 Pin	
DIP Socket 8 Pin	
Microcontroller ATtiny85-20PU	
Relay JRC-19F & MT2 C93428 12V DPDT (1A/125V)	

Πίνακας (4β): Υλικά που χρησιμοποιήθηκαν στο breadboard.

4.2.1.2 Υλοποίηση κατασκευής

Από την τελική σχεδίαση στο πρόγραμμα Fritzing περάσαμε στην υλοποίηση σε breadboard.

Βασικά υλικά που χρησιμοποιήθηκαν ήταν το Voltage Regulator για την καλύτερη σταθεροποίηση της αποδιδόμενης τάσης, ο μικροελεγκτής ATtiny85-20PU προγραμματισμένος κατάλληλα καθώς και το Relay για την επίτευξη του αποτελέσματος.

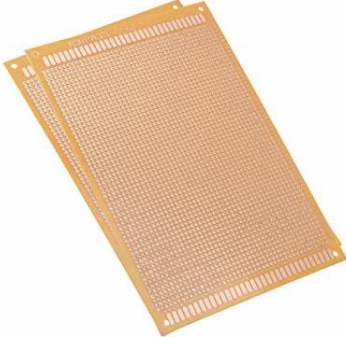







Εικόνα (31): Μετασχηματιστής τάσης στο breadboard.

4.2.2 Σε διάτρητη πλακέτα

4.2.2.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά υλοποίησης για το breadboard, όπου και τα παραθέτουμε ακολούθως [11, 21, 22]:

Διάτρητη πλακέτα	
Dc Power Jack	
Electrolytic Capacitor 25V 47uF	
Voltage Regulator LD1117 - 3.3V 800mA	
Voltage Regulator L78S05CV - 5V 2A	
Voltage Regulator L7809CV - 9V 1.5A	
LED Green 5mm THT	
LED Yellow 5mm THT	

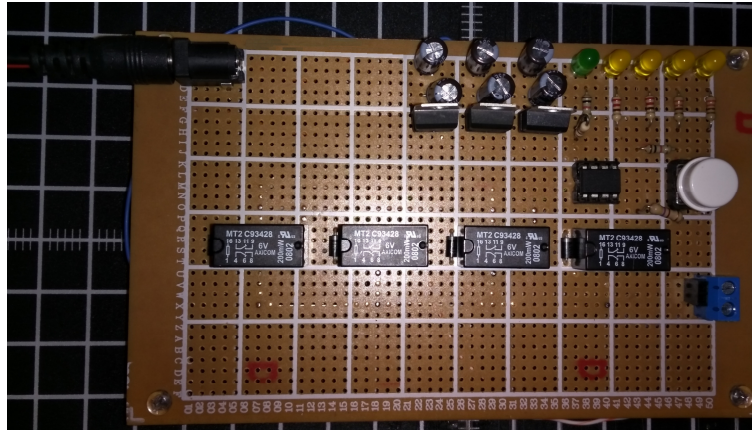
Πίνακας (5α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

Resistor THT 1kΩ και 220Ω	
Diodes 1N4007	
Tact Switch	
Cap for Tact Button	
DIP Socket 8 Pin	
Microcontroller ATtiny85-20PU	
Relay JRC-19F & MT2 C93428 12V DPDT (1A/125V)	
Pin Header 1x2 Female	

Πίνακας (5β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.2.2.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε breadboard περάσαμε στην υλοποίηση σε διάτρητη πλακέτα, με τις ανάλογες συνδέσεις και κολλήσεις.







Εικόνα (32): Μετασχηματιστής τάσης σε διάτρητη πλακέτα.

4.2.3 Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)

4.2.3.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά της υλοποίησης για τη διάτρητη πλακέτα και το σχεδιασμό, όπου και τα παραθέτουμε ακολούθως [11, 21, 22]:

Χάλκινη πλακέτα δύο επιπέδων	
Dc Power Jack	
Electrolytic Capacitor 25V 47uF	
Voltage Regulator LM1085 IT - 3.3V 3A	
Voltage Regulator LM1085 IT - 5V 3A	
Voltage Regulator L7809CV - 9V 1.5A	

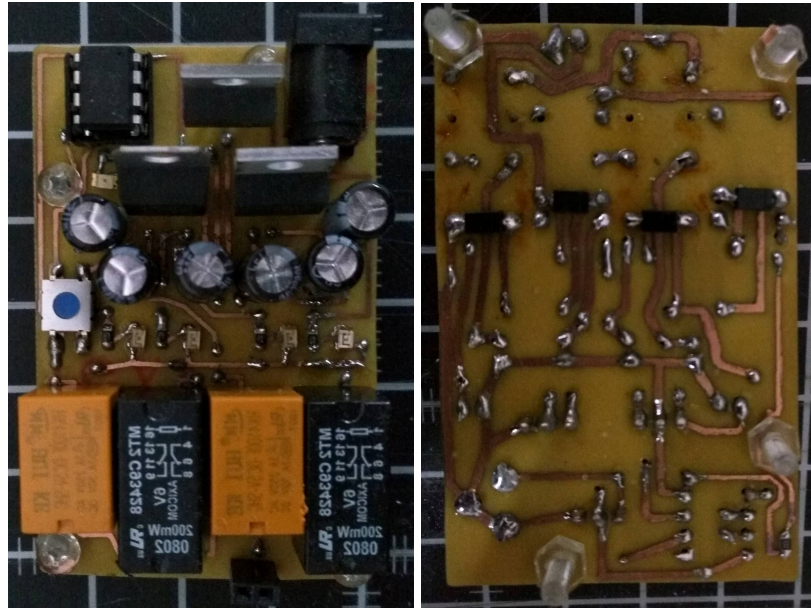
Πίνακας (6α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

LED Green 0805 SMD	
LED Yellow 0805 SMD	
Resistor 0805 SMD 1kΩ και 220Ω	
Diode 1A SMD	
Tact Switch SMD 4.5x4.5mm 3.8mm 4pin	
DIP Socket 8 Pin	
Microcontroller ATtiny85-20PU	
Relay JRC-19F & MT2 C93428 12V DPDT (1A/125V)	
Relay HUI KE HK4100F-DC12V 3A	
Pin Header 1x2 Female	

Πίνακας (6β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.2.3.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε διάτρητη πλακέτα περάσαμε στην υλοποίηση σε χάλκινη πλακέτα δύο επιπέδων, με τις ανάλογες κολλήσεις.



Εικόνα (33): Μετασχηματιστής τάσης σε χάλκινη πλακέτα.

Για την υλοποίηση της κατασκευής της παραπάνω πλακέτας, παίρνοντας όπου ήταν απαραίτητο τα κατάλληλα μέτρα προστασίας, ακολουθήθηκε μια διαδικασία παρόμοια της εισαγωγής της υλοποίησης κατασκευής.

- Αρχικά κατασκευάστηκε μία συσκευή εκπομπής ακτίνων UV.

Προσοχή την ώρα της λειτουργίας (εκπομπής ακτίνων UV) δεν πρέπει να έρθει η πηγή σε άμεση οπτική επαφή με τα μάτια.

- Στη συνέχεια αποτυπώθηκε το σχέδιο της πλακέτας από το αντίστοιχο πρόγραμμα σε διαφανές φύλλο.
- Ακολούθως, κόπηκε η χάλκινη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών.
- Καθαρίστηκε ο χαλκός.
- Κόπηκε το φωτοευαίσθητο film στο επιθυμητό μέγεθος και αφαιρέθηκε η προστατευτική μεμβράνη.
- Τοποθετήθηκε η πλακέτα σε χαρτί A4.
- Πάνω από την πλακέτα τοποθετήθηκε το φωτοευαίσθητο film.

- Διπλώθηκε το χαρτί και θερμάνθηκε η πλακέτα υπό πίεση.
- Ξεδιπλώθηκε το χαρτί και κολλήθηκε το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Εκτέθηκε σε ακτίνες UV (στη δικιά μας περίπτωση επί μιάμιση ώρα, η χρονική διάρκεια έκθεσης σε ακτίνες UV μεταβάλλεται ανάλογα την πηγή εκπομπής).
- Αφαιρέθηκε το χαρτί A4 και το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Ίδια διαδικασία και για το δεύτερο επίπεδο.
- Στη συνέχεια, αφαιρέθηκε η προστατευτική μεμβράνη από το φωτοευαίσθητο film.
- Τοποθετήθηκε η πλακέτα σε διάλυμα, για να αφαιρεθεί το μαλακό film (διάλυμα θερμοκρασίας ~ 85 οC).
- Ακολούθως, τέθηκε η πλακέτα σε διάλυμα, για να επιτευχθεί η αποχάλκωσή της (πλην της επιφανείας η οποία καλυπτόταν από το φωτοευαίσθητο film).
- Βυθίστηκε η πλακέτα σε διάλυμα, για την απομάκρυνση του φωτοευαίσθητου film.
- Διάτρηση της πλακέτας σε σημεία όπου ήταν απαραίτητο.
- Τοποθέτηση και κόλληση των υλικών.

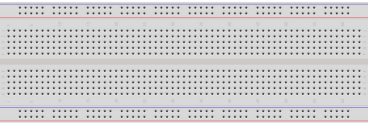




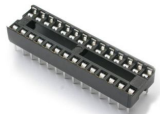



4.3 Πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-PU

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, ήταν απαραίτητο οι κενοί (άδειοι) μικροελεγκτές ATmega328P-PU, για να μπορέσουμε να τους προγραμματίσουμε, να κατασκευάσουμε μια μόνιμη πλακέτα που θα μας έδινε αυτή τη δυνατότητα. Μία απαραίτητη διαδικασία διότι κάθε ένας καινούριος μικροελεγκτής ATmega πιθανών να είναι τελείως κενός, χωρίς να είναι φορτωμένο το bootloader. Τη διαδικασία φόρτωσης του bootloader την εκτελούμε μία φορά για κάθε έναν καινούριο μικροελεγκτή ATmega.

4.3.1 Σε breadboard

4.3.1.1 Υλικά κατασκευής

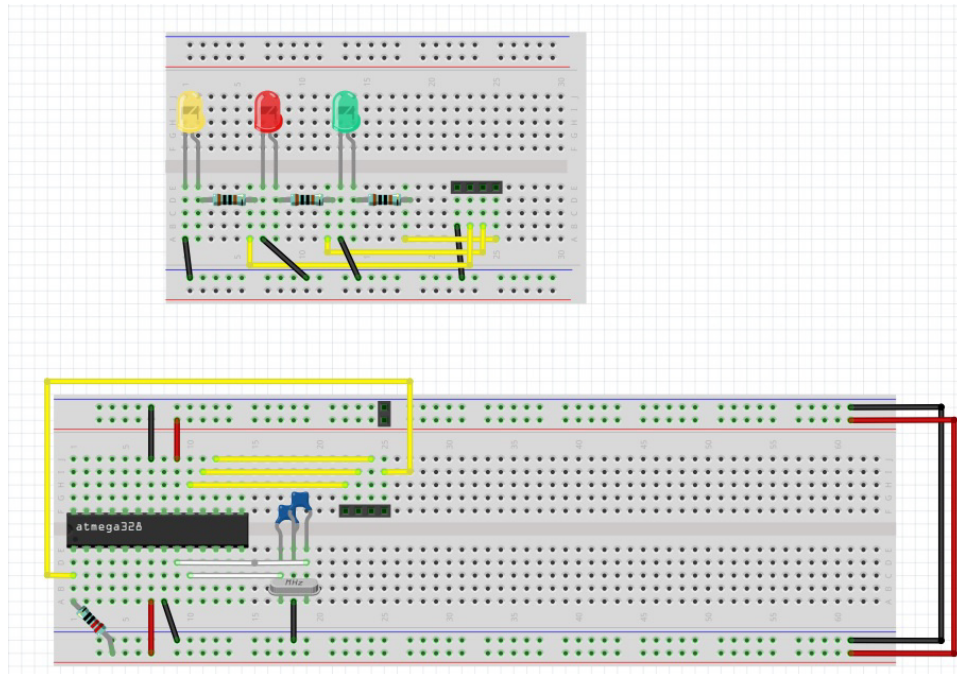
Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά με βάση τις ανάγκες της εργασίας, όπου και τα παραθέτουμε ακολούθως [11, 22]:

Breadboard	
LED Green 5mm THT	
LED Yellow 5mm THT	
LED Red 5mm THT	
Resistor THT 1kΩ και 220Ω	
DIP Socket 28 Pin	
Microcontroller ATmega328P-PU	
Crystal 16MHz (Slime profile)	
Πυκνωτής Κεραμικός Μονολιθικός 22pF	

Πίνακας (7): Υλικά που χρησιμοποιήθηκαν για breadboard.

4.3.1.2 Υλοποίηση κατασκευής

Από την τελική σχεδίαση στο πρόγραμμα Fritzing περάσαμε στην υλοποίηση σε breadboard.

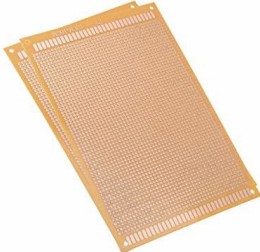


Εικόνα (34): Breadboard για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.

4.3.2 Σε διάτρητη πλακέτα

4.3.2.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά υλοποίησης για το breadboard, όπου και τα παραθέτουμε ακολούθως [11, 22]:

Διάτρητη πλακέτα	
------------------	---

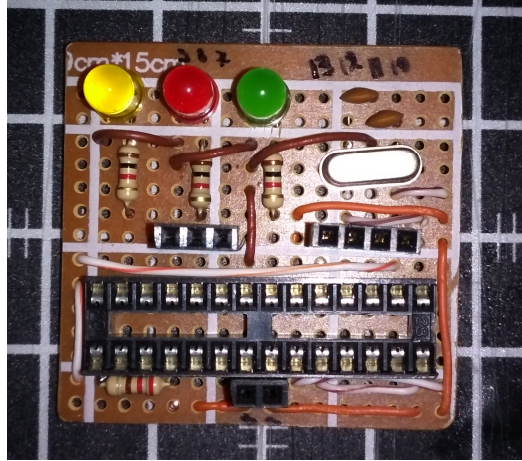
Πίνακας (8α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

LED Green 5mm THT	
LED Yellow 5mm THT	
LED Red 5mm THT	
Resistor THT 1kΩ και 220Ω	
DIP Socket 28 Pin	
Microcontroller ATmega328P-PU	
Crystal 16MHz (Slime profile)	
Πυκνωτής Κεραμικός Μονολιθικός 22pF	
Pin Header 1x2 Female 1x3 Female 1x4 Female	

Πίνακας (8β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.3.2.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε breadboard περάσαμε στην υλοποίηση σε διάτρητη πλακέτα. Αρχικά, κόψαμε τη διάτρητη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών. Ακολούθως, τοποθετήθηκαν τα υλικά στην πλακέτα με τις ανάλογες συνδέσεις και κολλήσεις.



Εικόνα (35): Διάτρητη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.







4.3.3 Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)

4.3.3.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά της υλοποίησης για τη διάτρητη πλακέτα και το σχεδιασμό, με βάση τη διαθεσιμότητα, όπου και τα παραθέτουμε ακολούθως [11, 22]:

Χάλκινη πλακέτα δύο επιπέδων	
LED Green 0805 SMD	
LED Yellow 0805 SMD	
LED Red 0805 SMD	

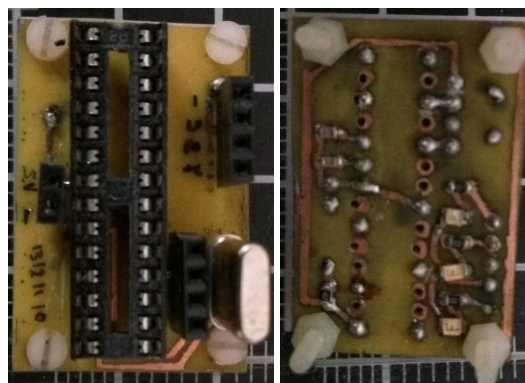
Πίνακας (9α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

Resistor 0805 SMD 1kΩ και 220Ω	
DIP Socket 28 Pin	
Microcontroller ATmega328P-PU	
Crystal 16MHz ((High profile))	
Ποικνωτής Κεραμικός 0805 SMD 22pF	
Pin Header 1x2 Female 1x3 Female 1x4 Female	

Πίνακας (9β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.3.3.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε διάτρητη πλακέτα περάσαμε στην υλοποίηση σε χάλκινη πλακέτα δύο επιπέδων, με τις ανάλογες κολλήσεις.



Εικόνα (36): Χάλκινη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-PU.

Για την υλοποίηση της κατασκευής της παραπάνω πλακέτας, παίρνοντας όπου ήταν απαραίτητο τα κατάλληλα μέτρα προστασίας, ακολουθήθηκε μια διαδικασία παρόμοια της εισαγωγής της υλοποίησης κατασκευής.

- Αρχικά αποτυπώθηκε το σχέδιο της πλακέτας από το αντίστοιχο πρόγραμμα σε διαφανές φύλλο.
- Ακολούθως, κόπηκε τη χάλκινη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών.
- Καθαρίστηκε ο χαλκός.
- Κόπηκε το φωτοευαίσθητο film στο επιθυμητό μέγεθος και αφαιρέθηκε η προστατευτική μεμβράνη.
- Τοποθετήθηκε η πλακέτα σε χαρτί A4.
- Πάνω από την πλακέτα τοποθετήθηκε το φωτοευαίσθητο film.
- Διπλώθηκε το χαρτί και θερμάνθηκε η πλακέτα υπό πίεση.
- Ξεδιπλώθηκε το χαρτί και κολλήθηκε το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Εκτέθηκε σε ακτίνες UV (στη δικιά μας περίπτωση επί μιάμιση ώρα, η χρονική διάρκεια έκθεσης σε ακτίνες UV μεταβάλλεται ανάλογα την πηγή εκπομπής).
- Αφαιρέθηκε το χαρτί A4 και το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Ίδια διαδικασία και για το δεύτερο επίπεδο.
- Στη συνέχεια, αφαιρέθηκε η προστατευτική μεμβράνη από το φωτοευαίσθητο film.
- Τοποθετήθηκε η πλακέτα σε διάλυμα, για να αφαιρέσω το μαλακό film (διάλυμα θερμοκρασίας ~ 85 οC).
- Ακολούθως, τέθηκε η πλακέτα σε διάλυμα, για να επιτευχθεί η αποχάλκωσή της (πλην της επιφανείας η οποία καλύπτεται από το φωτοευαίσθητο film).

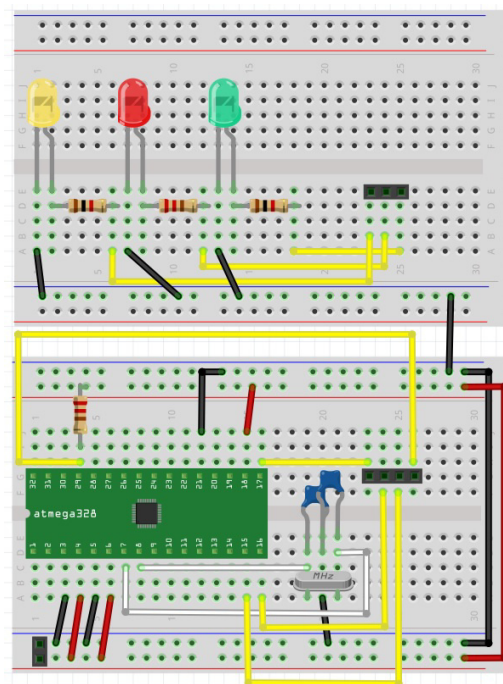
- Βυθίστηκε η πλακέτα σε διάλυμα, για την απομάκρυνση του φωτοευαίσθητου film.
- Διάτρηση της πλακέτας σε σημεία όπου ήταν απαραίτητο.
- Τοποθέτηση και κόλληση των υλικών.

4.4 Πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATmega328P-AU

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, ήταν απαραίτητο οι κενοί (άδειοι) μικροελεγκτές ATmega328P-AU, για να μπορέσουμε να τους προγραμματίσουμε, να κατασκευάσουμε μια μόνιμη πλακέτα που θα μας έδινε αυτή τη δυνατότητα. Μία απαραίτητη διαδικασία διότι κάθε ένας καινούριος μικροελεγκτής ATmega πιθανών να είναι τελείως κενός, χωρίς να είναι φορτωμένο το bootloader. Τη διαδικασία φόρτωσης του bootloader την εκτελούμε μία φορά για κάθε έναν καινούριο μικροελεγκτή ATmega328P-AU.

4.4.1 Σε breadboard

Η διαδικασία αυτή πραγματοποιήθηκε μέσω του προγράμματος Fritzing και μόνο, λόγω της διαθεσιμότητας των υλικών.



Εικόνα (37): Σχέδιο στο πρόγραμμα Fritzing για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU για το breadboard.







4.4.2 Σε διάτρητη πλακέτα

Η διαδικασία αυτή δεν παρακάμφθηκε λόγω της διαθεσιμότητας των υλικών.



4.4.3 Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)

4.4.3.1 Υλικά κατασκευής

Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά κατασκευής με βάση το σχεδιασμό και τη διαθεσιμότητά τους, όπου και τα παραθέτουμε ακολούθως [11, 22]:

Χάλκινη πλακέτα δύο επιπέδων	
LED Green 0805 SMD	
LED Yellow 0805 SMD	
LED Red 0805 SMD	
Resistor 0805 SMD 1kΩ και 220Ω	
Microcontroller ATmega328P-AU	
Crystal 16MHz ((Slime profile))	

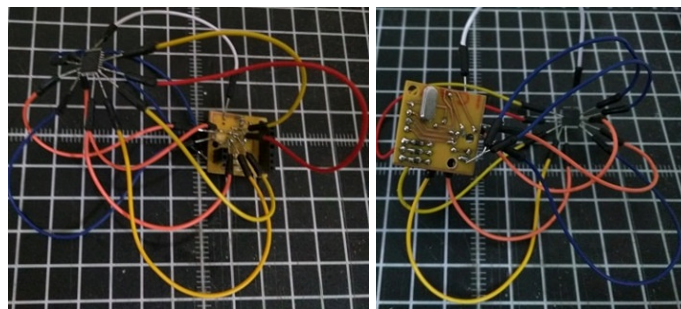
Πίνακας (10α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

Ποικνωτής Κεραμικός 0805 SMD 22pF	
Pin Header 1x2 Female 1x3 Female 1x4 Female	

Πίνακας (10β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.4.3.2 Υλοποίηση κατασκευής

Την υλοποίηση σε χάλκινη πλακέτα δύο επιπέδων, με τις ανάλογες κολλήσεις, παραθέτουμε ακολούθως.



Εικόνα (38): Χάλκινη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATmega328P-AU.

Για την υλοποίηση της κατασκευής της παραπάνω πλακέτας, παίρνοντας όπου ήταν απαραίτητο τα κατάλληλα μέτρα προστασίας, ακολουθήθηκε μια διαδικασία παρόμοια της εισαγωγής της υλοποίησης κατασκευής.

- Αρχικά αποτυπώθηκε το σχέδιο της πλακέτας από το αντίστοιχο πρόγραμμα σε διαφανές φύλλο.
- Ακολούθως, κόπηκε η χάλκινη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών.
- Καθαρίστηκε ο χαλκός.

- Κόπηκε το φωτοευαίσθητο film στο επιθυμητό μέγεθος και αφαιρέθηκε η προστατευτική μεμβράνη.
- Τοποθετήθηκε η πλακέτα σε χαρτί A4.
- Πάνω από την πλακέτα τέθηκε το φωτοευαίσθητο film.
- Διπλώθηκε το χαρτί και θερμάνθηκε η πλακέτα υπό πίεση.
- Ξεδιπλώθηκε το χαρτί και κολλήθηκε το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Εκτέθηκε σε ακτίνες UV (στη δικιά μας περίπτωση επί μιάμιση ώρα, η χρονική διάρκεια έκθεσης σε ακτίνες UV μεταβάλλεται ανάλογα την πηγή εκπομπής).
- Αφαιρέθηκε το χαρτί A4 και το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Ίδια διαδικασία και για το δεύτερο επίπεδο.
- Στη συνέχεια, αφαιρέθηκε η προστατευτική μεμβράνη από το φωτοευαίσθητο film.
- Τοποθετήθηκε η πλακέτα σε διάλυμα, για να αφαιρεθεί το μαλακό film (διάλυμα θερμοκρασίας ~ 85 oC).
- Ακολούθως, τέθηκε η πλακέτα σε διάλυμα, για να επιτευχθεί η αποχάλκωσή της (πλην της επιφανείας η οποία καλυπτόταν από το φωτοευαίσθητο film).
- Βυθίστηκε η πλακέτα σε διάλυμα, για την απομάκρυνση του φωτοευαίσθητου film.
- Διάτρηση της πλακέτας σε σημεία όπου ήταν απαραίτητο.
- Τοποθέτηση και κόλληση των υλικών.

Σημείωση: δεν κολλήθηκε ο μικροελεγκτής ATmega328P-AU απευθείας επάνω στην πλακέτα αλλά με πρόσθετους διαύλους επικοινωνίας, διότι δεν ήταν η μόνιμη θέση του μικροελεγκτή και στην πλακέτα θα εφαρμόζαμε και άλλους ίδιους μικροελεγκτές.

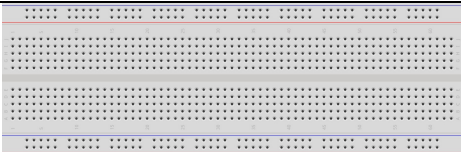





4.5 Πλακέτα για προγραμματισμό μέσω ISP για μικροελεγκτή ATtiny85-20PU

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, ήταν απαραίτητο οι κενοί (άδειοι) μικροελεγκτές ATtiny85-20PU, για να μπορέσουμε να τους προγραμματίσουμε, να κατασκευάσουμε μια μόνιμη πλακέτα που θα μας έδινε αυτή τη δυνατότητα.

4.5.1 Σε breadboard

4.5.1.1 Υλικά κατασκευής

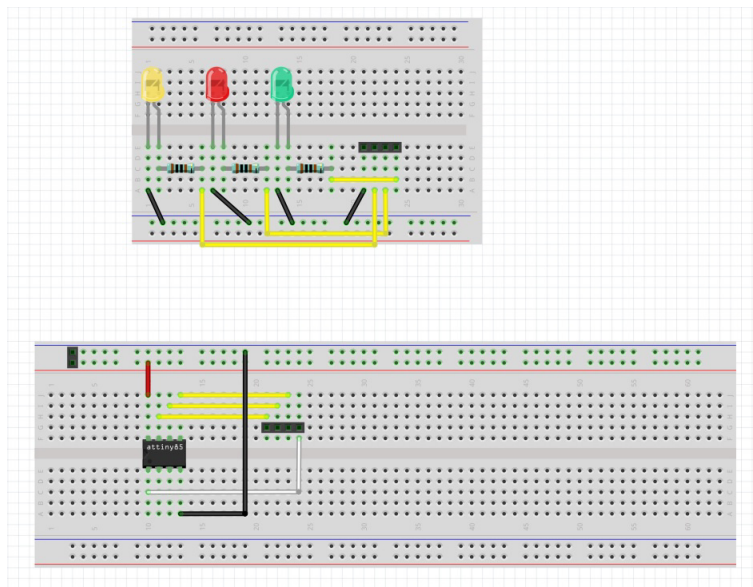
Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά με βάση τις ανάγκες της εργασίας, όπου και παραθέτονται ακολούθως [11, 22]:

Breadboard	
LED Green 5mm THT	
LED Yellow 5mm THT	
LED Red 5mm THT	
Resistor THT των 220Ω	
Microcontroller ATtiny85-20PU	

Πίνακας (11): Υλικά που χρησιμοποιήθηκαν για breadboard.

4.5.1.2 Υλοποίηση κατασκευής

Από την τελική σχεδίαση στο πρόγραμμα Fritzing περάσαμε στην υλοποίηση σε breadboard.



Εικόνα (39): Breadboard για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.




4.5.2 Σε διάτρητη πλακέτα

4.5.2.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά υλοποίησης για το breadboard, όπου και τα παραθέτουμε ακολούθως [11, 22]:



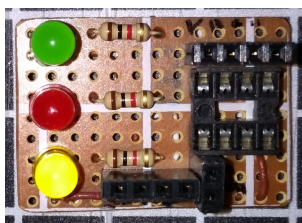
Πίνακας (12α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

LED Green 5mm THT	
LED Yellow 5mm THT	
LED Red 5mm THT	
Resistor THT των 220Ω	
DIP Socket 8 Pin	
Microcontroller ATtiny85-20PU	
Pin Header 1x2 Female 1x4 Female 1x4 Female	

Πίνακας (12β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.5.2.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε breadboard περάσαμε στην υλοποίηση σε διάτρητη πλακέτα. Αρχικά, κόπηκε τη διάτρητη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών. Ακολούθως, τοποθετήθηκαν τα υλικά στην πλακέτα με τις ανάλογες συνδέσεις και κολλήσεις.








Εικόνα (40): Διάτρητη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.




4.5.3 Σε χάλκινη πλακέτα ενός επιπέδου (δημιουργία PCB)

4.5.3.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά της υλοποίησης για τη διάτρητη πλακέτα και το σχεδιασμό, με βάση τη διαθεσιμότητα, όπου και τα παραθέτουμε ακολούθως [11, 22]:

Χάλκινη πλακέτα ενός επιπέδου	
LED Green 0805 SMD	
LED Yellow 0805 SMD	
LED Red 0805 SMD	
Resistor 0805 SMD των 220Ω	

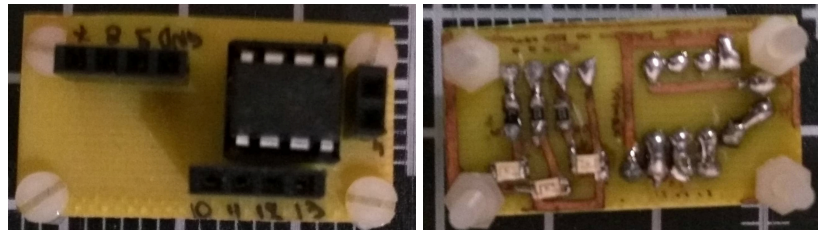
Πίνακας (13α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

DIP Socket 8 Pin	
Microcontroller ATtiny85-20PU	
Pin Header 1x2 Female 1x4 Female 1x4 Female	

Πίνακας (13β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.5.3.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε διάτρητη πλακέτα περάσαμε στην υλοποίηση σε χάλκινη πλακέτα ενός επιπέδου, με τις ανάλογες κολλήσεις.



Εικόνα (41): Χάλκινη πλακέτα για προγραμματισμό μέσω ISP του μικροελεγκτή ATtiny85-20PU.

Για την υλοποίηση της κατασκευής της παραπάνω πλακέτας, παίρνοντας όπου ήταν απαραίτητο τα κατάλληλα μέτρα προστασίας, ακολουθήθηκε μια διαδικασία παρόμοια της εισαγωγής της υλοποίησης κατασκευής.

- Αρχικά αποτυπώθηκε το σχέδιο της πλακέτας από το αντίστοιχο πρόγραμμα σε διαφανές φύλλο.
- Ακολούθως, κόπηκε η χάλκινη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών.
- Καθαρίστηκε ο χαλκός.

- Κόπηκε το φωτοευαίσθητο film στο επιθυμητό μέγεθος και αφαιρέθηκε η προστατευτική μεμβράνη.
- Τοποθετήθηκε η πλακέτα σε χαρτί A4.
- Πάνω από την πλακέτα τοποθετήθηκε το φωτοευαίσθητο film.
- Διπλώθηκε το χαρτί και θερμάνθηκε η πλακέτα υπό πίεση.
- Ξεδιπλώθηκε το χαρτί και κολλήθηκε το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Εκτέθηκε σε ακτίνες UV (στη δικιά μας περίπτωση επί μιάμιση ώρα, η χρονική διάρκεια έκθεσης σε ακτίνες UV μεταβάλλεται ανάλογα την πηγή εκπομπής).
- Αφαιρέθηκε το χαρτί A4 και το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Στη συνέχεια, αφαιρέθηκε η προστατευτική μεμβράνη από το φωτοευαίσθητο film.
- Τοποθετήθηκε η πλακέτα σε διάλυμα, για να αφαιρεθεί το μαλακό film (διάλυμα θερμοκρασίας ~ 85 οC).
- Ακολούθως, τέθηκε η πλακέτα σε διάλυμα, για να επιτευχθεί η αποχάλκωσή της (πλην της επιφανείας η οποία καλύπτεται από το φωτοευαίσθητο film).
- Βυθίστηκε η πλακέτα σε διάλυμα, για την απομάκρυνση του φωτοευαίσθητου film.
- Διάτρηση της πλακέτας σε σημεία όπου ήταν απαραίτητο.
- Τοποθέτηση και κόλληση των υλικών.

4.6 Πλακέτα για προγραμματισμό μέσω FTDI (σειριακής θύρας) σε μικροελεγκτή ATmega328P-PU

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, ακολούθως της απαραίτητης διαδικασίας ISP έτσι ώστε να φορτώσουμε το bootloader σε κάθε έναν καινούριο μικροελεγκτή ATmega, περάσαμε στην υλοποίηση της πλακέτας του προγραμματισμού και επαναπρογραμματισμού μέσω FTDI (σειριακής θύρας) του μικροελεγκτή.

4.6.1 Σε breadboard

4.6.1.1 Υλικά κατασκευής

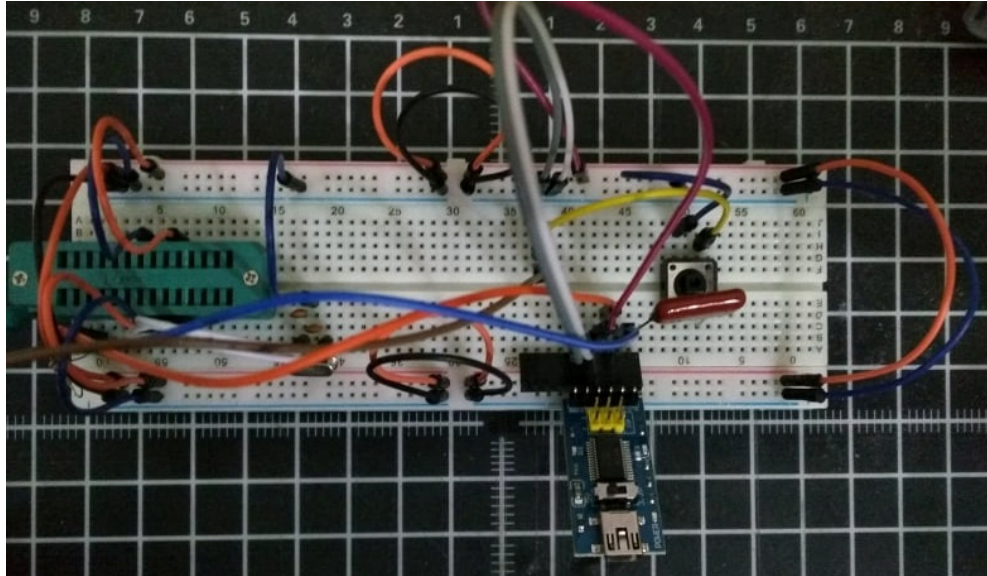
Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά με βάση τις ανάγκες της εργασίας, όπου και παραθέτονται ακολούθως [11, 22, 23]:

Breadboard	
Resistor THT 1kΩ	
ZIF Socket 28 Pin	
Microcontroller ATmega328P-PU	
Crystal 16MHz THT (Slime profile)	
Ποικνωτής Κεραμικός Μονολιθικός 22pF THT	
Ποικνωτής Κεραμικός 100nF THT	
Tact Switch	
FTDI FT232RL USB to TTL Serial Converter Adapter Module Switchable 3.3 and 5V (Blue)	

Πίνακας (14): Υλικά που χρησιμοποιήθηκαν για breadboard.

4.6.1.2 Υλοποίηση κατασκευής

Από την τελική σχεδίαση στο πρόγραμμα Fritzing περάσαμε στην υλοποίηση σε breadboard.



Εικόνα (42): Breadboard για προγραμματισμό μέσω FTDI (σειριακής θύρας) του μικροελεγκτή ATmega328P-PU.

4.6.2 Σε διάτρητη και χάλκινη πλακέτα

Η διαδικασία αυτή παρακάμφθηκε και εφόσον ελέγξαμε τη λειτουργικότητα του κυκλώματος σε breadboard, προχωρήσαμε στη μελέτη και δημιουργία της πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.

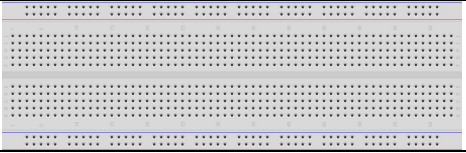








4.7 Πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU

Προχωρώντας την εργασία, έπειτα τον έλεγχο της ορθής λειτουργικότητας του μικροελεγκτή ATmega328P-PU σε breadboard για τον προγραμματισμό και επαναπρογραμματισμό μέσω FTDI (σειριακής θύρας) του μικροελεγκτή και παίρνοντας τα ζητούμενα για τη δημιουργία της πλακέτας αυτόνομου κυκλώματος αισθητήρων, ακολουθήθηκαν οι οδηγίες για την δημιουργία της πλακέτας.

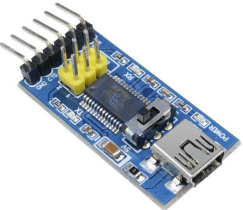

4.7.1 Σε breadboard

4.7.1.1 Υλικά κατασκευής

Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά με βάση τις ανάγκες της εργασίας, όπου και παραθέτονται ακολούθως [11, 22, 23]:

Breadboard	
Resistor THT 1kΩ	
ZIF Socket 28 Pin	
Microcontroller ATmega328P-PU	
Crystal 16MHz THT (Slime profile)	
Ποικνωτής Κεραμικός Μονολιθικός 22pF THT	
Ποικνωτής Κεραμικός 100nF THT	
Tact Switch	
Cap for Tact Button	

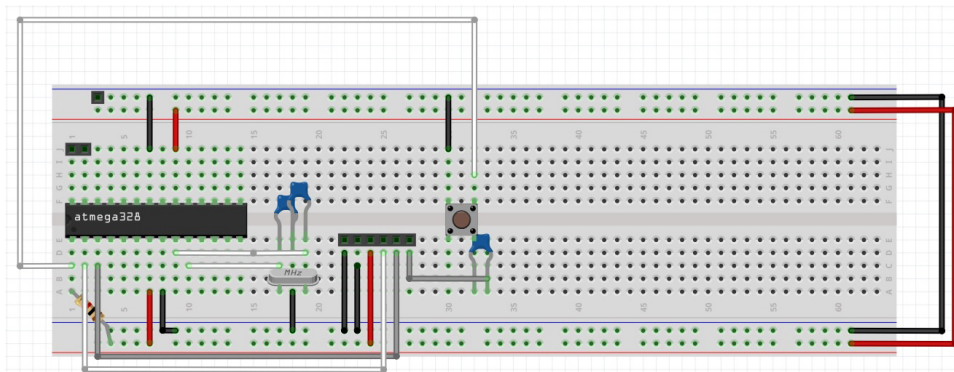
Πίνακας (15α): Υλικά που χρησιμοποιήθηκαν για breadboard.

<p>FTDI FT232RL USB to TTL Serial Converter Adapter Module Switchable 3.3 and 5V (Blue)</p>	
<p>Pin Header 1x2 Female 1x2 Female 1x2 Female 1x6 Female</p>	

Πίνακας (15β): Υλικά που χρησιμοποιήθηκαν για breadboard.

4.7.1.2 Υλοποίηση κατασκευής

Από την τελική σχεδίαση στο πρόγραμμα Fritzing περάσαμε στην υλοποίηση σε breadboard. Παίρνοντας τα δεδομένα και τις ανάγκες της εργασίας για την ελαχιστοποίηση των εξόδων, την παρακολούθηση της σειριακής θύρας καθώς και τη χρησιμοποίηση αναλογικών θυρών του μικροελεγκτή.



Εικόνα (43): Breadboard αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.

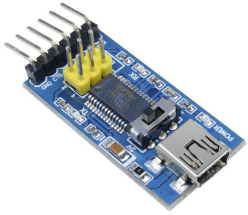


4.7.2 Σε διάτρητη πλακέτα

4.7.2.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά υλοποίησης για το breadboard, όπου και τα παραθέτουμε ακολούθως [11, 22, 23]:

Διάτρητη πλακέτα	
Resistor THT των 220Ω	
DIP Socket 28 Pin	
Microcontroller ATmega328P-PU	
Crystal 16MHz THT (Slime profile)	
Πυκνωτής Κεραμικός Μονολιθικός 22pF THT	
Πυκνωτής Κεραμικός 100nF THT	
Tact Switch	
Cap for Tact Button	

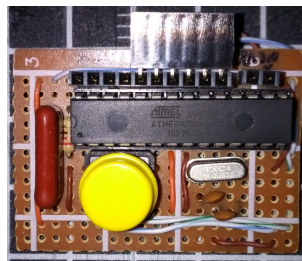
Πίνακας (16α): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

<p>FTDI FT232RL USB to TTL Serial Converter Adapter Module Switchable 3.3 and 5V (Blue)</p>	
<p>Pin Header 1x2 Female 1x10 Female</p>	
<p>Pin Header 1x6 Female Right Angle</p>	

Πίνακας (16β): Υλικά που χρησιμοποιήθηκαν για τη διάτρητη πλακέτα.

4.7.2.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε breadboard περάσαμε στην υλοποίηση σε διάτρητη πλακέτα. Αρχικά, κόπηκε η διάτρητη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών. Ακολούθως, τοποθετήθηκαν τα υλικά στην πλακέτα με τις ανάλογες συνδέσεις και κολλήσεις.



Εικόνα (44): Διάτρητη πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.

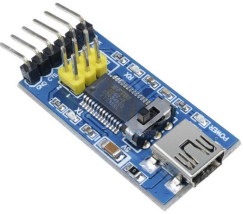
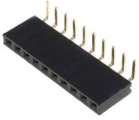
4.7.3 Σε γάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)

4.7.3.1 Υλικά κατασκευής

Τα υλικά κατασκευής ήταν τα ανάλογα με τα υλικά της υλοποίησης για τη διάτρητη πλακέτα, το σχεδιασμό και με βάση τη διαθεσιμότητα, όπου και τα παραθέτουμε ακολούθως [11, 22, 23]:

Χάλκινη πλακέτα δύο επιπέδων	
Resistor 0805 SMD των 220Ω	
DIP Socket 28 Pin	
Microcontroller ATmega328P-PU	
Crystal 16MHz THT (Slime profile)	
Πυκνωτής Κεραμικός 0805 SMD 22pF	
Πυκνωτής Κεραμικός 100nF THT	
Tact Switch SMD 4.5x4.5mm 3.8mm 4pin	

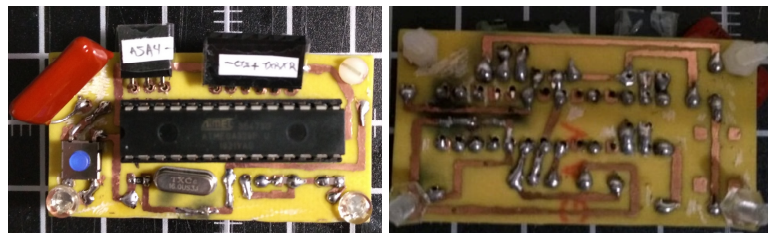
Πίνακας (17α): Υλικά που χρησιμοποιήθηκαν για τη γάλκινη πλακέτα.

<p>FTDI FT232RL USB to TTL Serial Converter Adapter Module Switchable 3.3 and 5V (Blue)</p>	
<p>Pin Header 1x6 Female Right Angle 1x3 Female Right Angle</p>	

Πίνακας (17β): Υλικά που χρησιμοποιήθηκαν για τη χάλκινη πλακέτα.

4.7.3.2 Υλοποίηση κατασκευής

Από την υλοποίηση σε διάτρητη πλακέτα περάσαμε στην υλοποίηση σε χάλκινη πλακέτα δύο επιπέδων, με τις ανάλογες κολλήσεις.



Εικόνα (45): Χάλκινη πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-PU.

Για την υλοποίηση της κατασκευής της παραπάνω πλακέτας, παίρνοντας όπου ήταν απαραίτητο τα κατάλληλα μέτρα προστασίας, ακολουθήθηκε μια διαδικασία παρόμοια της εισαγωγής της υλοποίησης κατασκευής.

- Αρχικά αποτυπώθηκε το σχέδιο της πλακέτας από το αντίστοιχο πρόγραμμα σε διαφανές φύλλο.
- Ακολούθως, κόπηκε η χάλκινη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών.
- Καθαρίστηκε ο χαλκός.

- Κόπηκε το φωτοευαίσθητο film στο επιθυμητό μέγεθος και αφαιρέθηκε η προστατευτική μεμβράνη.
- Τοποθετήθηκε η πλακέτα σε χαρτί A4.
- Πάνω από την πλακέτα τέθηκε το φωτοευαίσθητο film.
- Διπλώθηκε το χαρτί και θερμάνθηκε η πλακέτα υπό πίεση.
- Ξεδιπλώθηκε το χαρτί και κολλήθηκε το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Εκτέθηκε σε ακτίνες UV (στη δικιά μας περίπτωση επί μιάμιση ώρα, η χρονική διάρκεια έκθεσης σε ακτίνες UV μεταβάλλεται ανάλογα την πηγή εκπομπής).
- Αφαιρέθηκε το χαρτί A4 και το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Ίδια διαδικασία και για το δεύτερο επίπεδο.
- Στη συνέχεια, αφαιρέθηκε η προστατευτική μεμβράνη από το φωτοευαίσθητο film.
- Τοποθετήθηκε η πλακέτα σε διάλυμα, για να αφαιρεθεί το μαλακό film (διάλυμα θερμοκρασίας ~ 85 οC).
- Ακολούθως, τέθηκε η πλακέτα σε διάλυμα, για να επιτευχθεί η αποχάλκωσή της (πλην της επιφανείας η οποία καλύπτεται από το φωτοευαίσθητο film).
- Βυθίστηκε η πλακέτα σε διάλυμα, για την απομάκρυνση του φωτοευαίσθητου film.
- Διάτρηση της πλακέτας σε σημεία όπου ήταν απαραίτητο.
- Τοποθέτηση και κόλληση των υλικών.

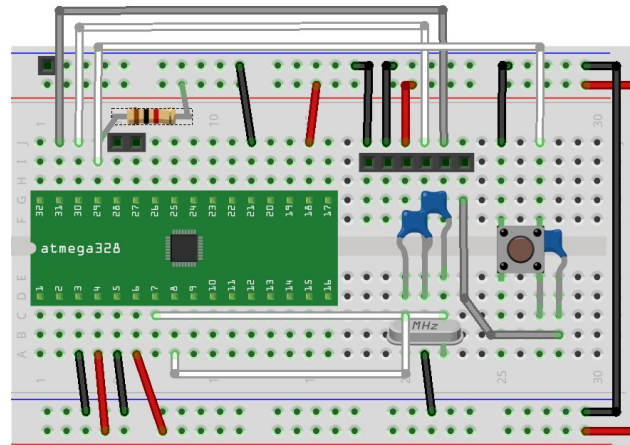
4.8 Πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU

Φτάνοντας την εργασία στο τελευταίο στάδιο, παίρνοντας τα ζητούμενα και τις ανάγκες για την ελαχιστοποίηση των εξόδων, την παρακολούθηση της σειριακής θύρας καθώς και τη χρησιμοποίηση αναλογικών θυρών του μικροελεγκτή για τη δημιουργία της πλακέτας αυτόνομου

κυκλώματος αισθητήρων, ακολουθήθηκαν οι οδηγίες για την δημιουργία της τελικής πλακέτας για την επίτευξη σμίκρυνσης του μεγέθους της.

4.8.1 Σε breadboard

Η διαδικασία αυτή πραγματοποιήθηκε μέσω του προγράμματος Fritzing και μόνο, λόγω της διαθεσιμότητας των υλικών.



Εικόνα (46): Σχέδιο στο πρόγραμμα Fritzing αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU για το breadboard.

4.8.2 Σε διάτρητη πλακέτα

Η διαδικασία αυτή δεν παρακάμφθηκε λόγω της διαθεσιμότητας των υλικών.

4.8.3 Σε χάλκινη πλακέτα δύο επιπέδων (δημιουργία PCB)

4.8.3.1 Υλικά κατασκευής

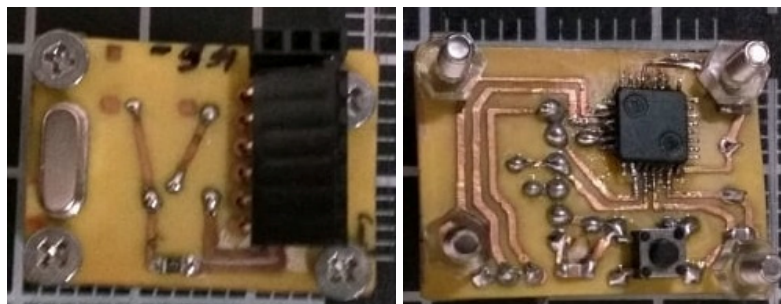
Επιλέχθηκαν τα κατάλληλα και προσβάσιμα υλικά κατασκευής με βάση το σχεδιασμό και τη διαθεσιμότητά τους, όπου και παραθέτονται ακολούθως [11, 22]:

Χάλκινη πλακέτα δύο επιπέδων	
Microcontroller ATmega328P-AU	
Resistor 0805 SMD 1kΩ	
Πυκνωτής Κεραμικός 0805 SMD 22pF 0805 SMD 100nF	
Crystal 16MHz ((Slime profile))	
Tact Switch SMD 4.5x4.5mm 3.8mm 4pin	
Pin Header 1x6 Female Right Angle 1x3 Female	

Πίνακας (18): Υλικά που χρησιμοποιήθηκαν για τη χάλκινη πλακέτα.

4.8.3.2 Υλοποίηση κατασκευής

Την υλοποίηση σε χάλκινη πλακέτα δύο επιπέδων, με τις ανάλογες κολλήσεις την παραθέτουμε παρακάτω.



Εικόνα (47): Χάλκινη πλακέτα αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328P-AU.

Για την υλοποίηση της κατασκευής της παραπάνω πλακέτας, παίρνοντας όπου ήταν απαραίτητο τα κατάλληλα μέτρα προστασίας, ακολουθήθηκε μια διαδικασία παρόμοια της εισαγωγής της υλοποίησης κατασκευής.

- Αρχικά αποτυπώθηκε το σχέδιο της πλακέτας από το αντίστοιχο πρόγραμμα σε διαφανές φύλλο.
- Ακολούθως, κόπηκε η χάλκινη πλακέτα στο επιθυμητό μέγεθος και έγινε απαλοιφή των γωνιών.
- Καθαρίστηκε ο χαλκός.
- Κόπηκε το φωτοευαίσθητο film στο επιθυμητό μέγεθος και αφαιρέθηκε η προστατευτική μεμβράνη.
- Τοποθετήθηκε η πλακέτα σε χαρτί A4.
- Πάνω από την πλακέτα τέθηκε το φωτοευαίσθητο film.
- Διπλώθηκε το χαρτί και θερμάνθηκε η πλακέτα υπό πίεση.
- Ξεδιπλώθηκε το χαρτί και κολλήθηκε το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Εκτέθηκε σε ακτίνες UV (στη δικιά μας περίπτωση επί μιάμιση ώρα, η χρονική διάρκεια έκθεσης σε ακτίνες UV μεταβάλλεται ανάλογα την πηγή εκπομπής).
- Αφαιρέθηκε το χαρτί A4 και το διαφανές φύλλο με το αποτυπωμένο σχέδιο της πλακέτας.
- Ίδια διαδικασία και για το δεύτερο επίπεδο.
- Στη συνέχεια, αφαιρέθηκε η προστατευτική μεμβράνη από το φωτοευαίσθητο film.
- Τοποθετήθηκε η πλακέτα σε διάλυμα, για να αφαιρεθεί το μαλακό film (διάλυμα θερμοκρασίας ~ 85 οC).
- Ακολούθως, τέθηκε η πλακέτα σε διάλυμα, για να επιτευχθεί η αποχάλκωσή της (πλην της επιφανείας η οποία καλύπτεται από το φωτοευαίσθητο film).

- Βυθίστηκε η πλακέτα σε διάλυμα, για την απομάκρυνση του φωτοευαίσθητου film.
- Διάτρηση της πλακέτας σε σημεία όπου ήταν απαραίτητο.
- Τοποθέτηση και κόλληση των υλικών.

ΚΕΦΑΛΑΙΟ 5^ο

ΥΛΟΠΟΙΗΣΗ ΛΟΓΙΣΜΙΚΟΥ

5.1 Εισαγωγή

Ως λογισμικό προγραμματισμού επιλέχθηκε το λογισμικό προγραμματισμού του Arduino. Το Arduino IDE, είναι γραμμένο σε γλώσσα Java, παρέχοντας δυνατότητα συμβατότητας σε πολλά λειτουργικά συστήματα. Διατίθεται δωρεάν από την επίσημη ιστοσελίδα του Arduino [6]. Η γλώσσα προγραμματισμού που μεταγλωττίζει ονομάζεται Wiring η οποία είναι παρόμοια της C / C++ με απλοποιήσεις και αλλαγές. Αναγνωρίζει την θύρα που συνδέεται η πλακέτα και μεταφέρει το πρόγραμμα μέσω σειριακής θύρας από τον υπολογιστή στην πλακέτα [6].

5.2 Υλοποίηση προγραμματισμού

Για να προγραμματιστούν οι μικροελεγκτές ATtiny85-20PU, ATmega328P-PU & AU ακολουθήθηκαν οι ανάλογες διαδικασίες που αναφέρθηκαν σε προηγούμενο κεφάλαιο, είτε μέσω της διαδικασίας In System Programming (ISP), είτε μέσω FTDI module (σειριακής θύρας) σε μικροελεγκτή ATmega328.

5.2.1 Προγραμματισμός μικροελεγκτή ATtiny85-20PU

Ακολούθως παραθέτουμε μέρος του κώδικα προγραμματισμού του μικροελεγκτή ATtiny85-20PU, ο οποίος χρησιμοποιήθηκε για την υλοποίηση της κατασκευής του ψηφιακού μετασχηματιστή τάσης.

```

void loop()

{

  button = digitalRead(switch_1);

  // Θέσεις διακόπτη επιλογής κατάστασης

  if (button == HIGH)

    {

      mode++;

      if (mode > 4) mode = 1;

    }

  // Κατάσταση 1 (3.3 Volt)

  if (mode == 1)

    {

      digitalWrite(volt_3, HIGH);

      digitalWrite(volt_5, LOW);

      digitalWrite(volt_9, LOW);

      digitalWrite(volt_12, LOW);

    }

```

5.2.2 Προγραμματισμός μικροελεγκτή ATmega328P-PU & AU

Ακολούθως παραθέτουμε μέρος του κώδικα προγραμματισμού των μικροελεγκτών ATmega328P-PU & AU, ο οποίος χρησιμοποιήθηκε για την υλοποίηση και έλεγχο της κατασκευής του αυτόνομου κυκλώματος αισθητήρων.

```
int mA = abs (ampere);

// απόλυτη τιμή

unsigned long cputime = millis();

/* Πρώτος αλγόριθμος κινούμενου παραθύρου για εξομάλυνση της
κυματομορφής,

* αυτό μας δίνει τη δυνατότητα να αυξήσουμε την
παρατηρησιμότητα σε σχέση με τις αλλαγές

* /

samples[200] = abs(ampere);

avg1 = 0.0;

for(int j=0; j<200; j++)

{

    samples[j] = samples[j+1];
```

ΚΕΦΑΛΑΙΟ 6^ο

ΠΕΙΡΑΜΑΤΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

6.1 Εισαγωγή

Ο έλεγχος και πειραματισμός τόσο του μετασχηματιστή τάσης όσο και του αυτόνομου κυκλώματος αισθητήρων με μικροελεγκτή ATmega328 ήταν πλέον και το τελικό στάδιο του όλου εγχειρήματος.

6.2 Μετασχηματιστής τάσης

Ακολούθως της υλοποίησης κατασκευής σε χάλκινη πλακέτα, εφαρμόστηκε ο έλεγχος της λειτουργικότητάς του, με βάση τις οδηγίες όπου είχαν δοθεί για τις προδιαγραφές του. Οι μετρήσεις όπου ελήφθησαν παραθέτονται στον ακόλουθο πίνακα:

	Πηγή (είσοδος)	12,13 Volt
Κοινή έξοδος	Απόδοση 1	3,31 Volt
	Απόδοση 2	5,00 Volt
	Απόδοση 3	9,00 Volt
	Απόδοση 4	12,12 Volt
Κατανάλωση ρεύματος του κυκλώματος σε όλες τις επιλογές (χωρίς πρόσθετο φορτίο)		0,04-0,05 Ampere

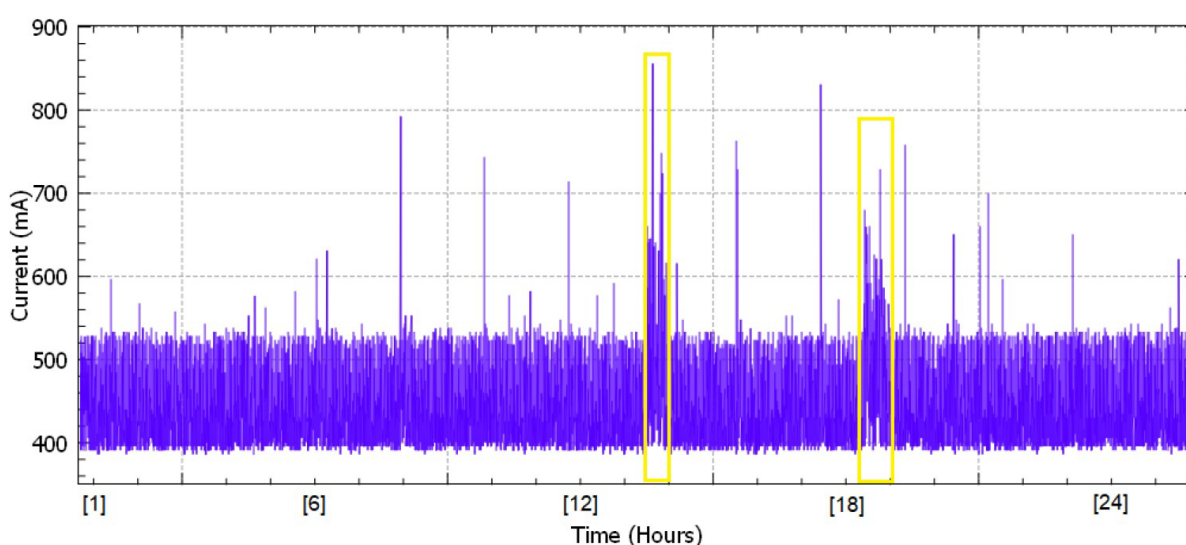
Πίνακας (19): Μετρήσεις στη χάλκινη πλακέτα του μετασχηματιστή τάσης.

6.3 Αυτόνομο κύκλωμα αισθητήρων με μικροελεγκτή ATmega328

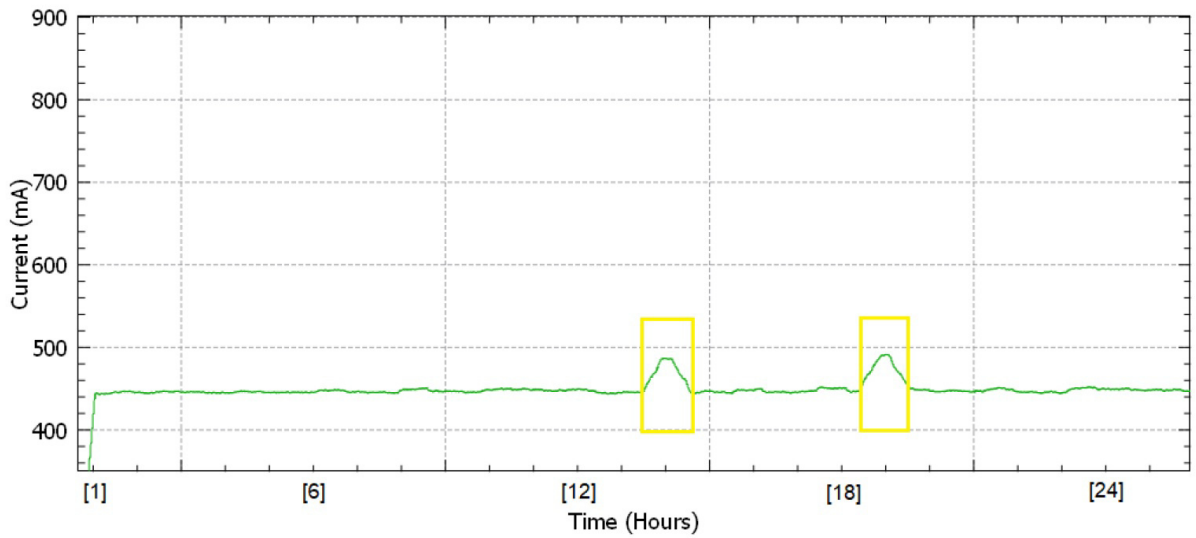
Μετά την υλοποίηση της κατασκευής σε χάλκινη πλακέτα, εφαρμόστηκε ο έλεγχος της λειτουργικότητάς του κυκλώματος μέσα από την εργασία «Intrusion Detection and Botnet Prevention Circuit for IoT Devices», όπου με τον κατάλληλο κώδικα, αντικατέστησε μία πλακέτα Arduino Uno και έδωσε αντίστοιχα και ορθά τα αποτελέσματα [2]. Η παραπάνω εργασία αποτελεί συνέχεια των [24-27], όπου μελετούν τη συσχέτιση του ρεύματος τροφοδοσίας μιας έξυπνης συσκευής, με τα λειτουργικά χαρακτηριστικά της, για τον εντοπισμό οποιασδήποτε λειτουργικής ανωμαλίας, αναπτύσσοντας μίας μεθοδολογία για τον εντοπισμό πιθανών επιθέσεων ή δυσλειτουργιών σε συσκευές IoT (Internet of Things), με σκοπό την ασφάλεια του υλικού.

Η εργασία είχε ως στόχο να εισαγάγει ένα κύκλωμα συνδεδεμένο με την τροφοδοσία συσκευής και να αναλύσει τη συμπεριφοράς της. Υπολογισμοί έγιναν με βάση την παροχή ρεύματος και τάσης, μετά το φιλτράρισμα των ηλεκτρικών σημάτων για να αφαιρεθεί ο θόρυβος. Το κύκλωμα ανίχνευσε οποιαδήποτε ανωμαλία της αναμενόμενης λειτουργίας με παρακολούθηση της υπερβολικής απορρόφησης ισχύος. Το κύκλωμα ανίχνευσης λειτουργεί σε πραγματικό χρόνο (σε πραγματικό χρόνο η ανίχνευση), δηλαδή συλλέγει τις πληροφορίες διαβάζοντας τις τιμές της τρέχουσας έντασης από τη συσκευή και την επεξεργάζεται για την ανίχνευση μη φυσιολογικών δραστηριοτήτων. Τα αποτελέσματα της δοκιμής του κυκλώματος έδειξαν εξαιρετική απόδοση στην ανίχνευση εισβολής, με 100% επιτυχία [2].

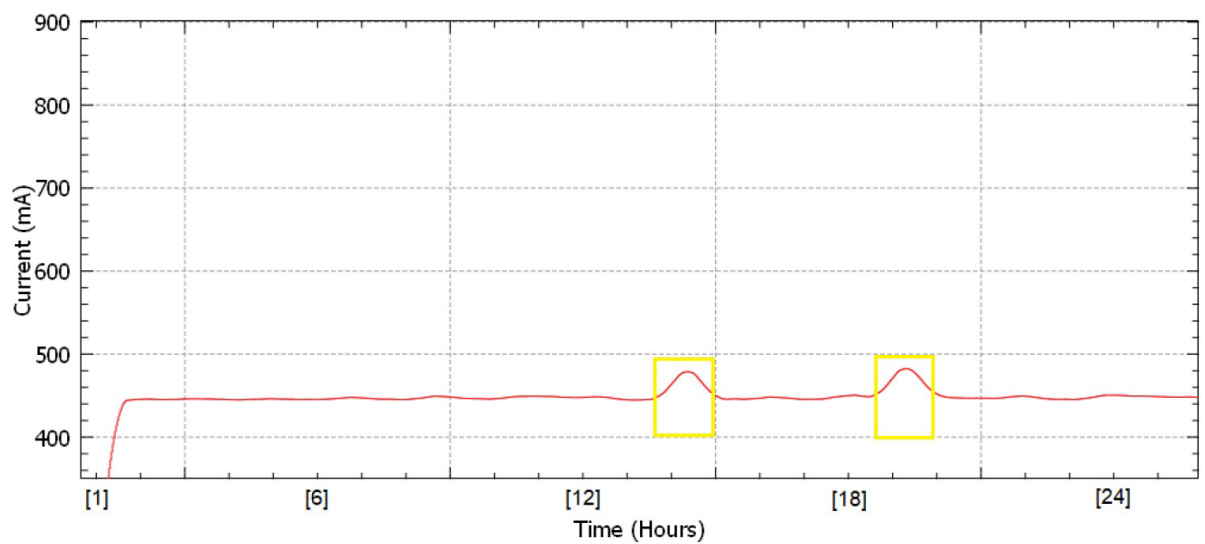
Οι μετρήσεις όπου ελήφθησαν παραθέτονται ακολούθως:



Εικόνα (48): Μετρήσεις ρεύματος [2].



Εικόνα (49): Μετρήσεις ρεύματος με το πρώτο φίλτρο [2].



Εικόνα (50): Μετρήσεις ρεύματος με το δεύτερο φίλτρο [2].

ΚΕΦΑΛΑΙΟ 7^ο

ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ

Σε αυτή την εργασία έγινε μία προσπάθεια για μία ολοκληρωμένη και τεκμηριωμένη προσέγγιση ενός ψηφιακού μετασχηματιστή τάσης και ενός αυτόνομου κυκλώματος αισθητήρων. Αρχικά πραγματοποιήθηκε μία εισαγωγή και μία ιστορική αναδρομή στα ολοκληρωμένα κυκλώματα. Αναφέρθηκαν ορισμένα μοντέλα από υπολογιστικές πλατφόρμες καθώς και ορισμένες εταιρίες όπου απασχολούνται με την παραγωγή μικροελεγκτών. Ιδιαίτερη έμφαση δόθηκε στους μικροελεγκτές όπου επιλέξαμε (ATmega328P-PU, ATmega328P-AU και ATtiny85-20PU). Εν συνεχεία αναφέρθηκαν ορισμένα προγράμματα σχεδίασης ηλεκτρονικών, πλακετών και πειραματισμού σε breadboard. Πραγματοποιήθηκε ο σχεδιασμός σε breadboard, αποδόθηκε το ηλεκτρονικό σχέδιο καθώς και η σχεδίαση PCB, στα επιλεγμένα προγράμματα (Fritzing, Multisim και Eagle). Αναλύθηκε εκτενώς το πρόγραμμα σύνταξης Arduino IDE, η διαδικασία ISP όπου εφαρμόστηκε καθώς και η φόρτωση προγράμματος μέσω FTDI. Στην υλοποίηση της κατασκευής, αναλύθηκε η διαδικασία για όλες τις πλακέτες, τα υλικά όπου χρησιμοποιήθηκαν σε breadboard, διάτρητη πλακέτα και τέλος σε χάλκινη πλακέτα. Δόθηκε μέρος του κώδικα τόσο για τον ψηφιακό μετασχηματιστή τάσης όσο και για το αυτόνομο κύκλωμα αισθητήρων. Αναφέρθηκαν τα πειράματα που εφαρμόστηκαν στο μετασχηματιστή τάσης και στο αυτόνομο κύκλωμα αισθητήρων καθώς και τα αποτελέσματα αυτών.

Το αυτόνομο κύκλωμα αισθητήρων είχε 100% επιτυχία για το σκοπό όπου σχεδιάστηκε. Μπορεί να μεταβληθεί η σχεδίασή του ως προς τις εξόδους του, με βάσει τις ανάγκες τόσο για το πλήθος όσο και για τον τύπο τους. Με αυτό τον τρόπο μπορούμε να έχουμε ένα μικρό και εξειδικευμένο επαναπρογραμματιζόμενο αυτόνομο κύκλωμα αισθητήρων με τους απαιτούμενους, για τις ανάγκες μας, ακροδέκτες.

Εκτός από τη δημιουργική απασχόληση σε ακαδημαϊκό επίπεδο, η ενασχόληση με τους μικροελεγκτές και τη δημιουργία PCB μπορεί να αναπτύξει και να εξελίξει τη συγκεκριμένη εργασία σε μία βάση για νέες και πολλαπλές εφαρμογές. Η σχεδίαση και υλοποίηση ενός αυτόνομου κυκλώματος αισθητήρων με διαφορετικό μικροελεγκτή περισσοτέρων δυνατοτήτων, όπως ATmega4809 ή nRF52840 ή ATmega2560, για ποιο σύνθετα και απαιτητικά προγράμματα, αποτελούν ορισμένα από τα σημαντικά βήμα για μελλοντική εργασία. Αξίζει να σημειωθεί ότι όταν δίνεται η ευκαιρία σε φοιτητές να ασχοληθούν με μικροελεγκτές και τη δημιουργία PCB που προϋποθέτουν προγραμματιστικές και σχεδιαστικές δεξιότητες, οι φοιτητές αυτοί αποκτούν συγκριτικό πλεονέκτημα στην εξέλιξη του αντικειμένου σε επαγγελματικό επίπεδο.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [2] D. Myridakis, P. Myridakis, and A. Kakarountas, "Intrusion Detection and Botnet Prevention Circuit for IoT Devices," in *2020 5th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, 2020, pp. 1-4.
- [3] J. F. Wakerly, "John F. Wakerly. Ψηφιακή Σχεδίαση Αρχές & Πρακτικές. Κλειδάριθμος. ISBN 960-209-728-0."
- [4] M. Paraskevas, G. Asimakopoulos, V. Triantafyllou, M. Παρασκευάς, Γ. Ασημακόπουλος, and B. Τριανταφύλλου, "Information Society," 2015.
- [5] R. H. Sudhan, M. G. Kumar, A. U. Prakash, S. A. R. Devi, and S. P, "Arduino Atmega-328 Microcontroller," *Ijireeice*, vol. 3, pp. 27-29, 2015.
- [7] Δ. Καλοφωλιάς, "Ο προγραμματισμός του μικροελεγκτή AVR ATMega328 με τη χρήση της πλατφόρμας Arduino, Εκδόσεις Τζιόλα, ISBN: 9789604186594," 2017.
- [24] D. Myridakis, G. Spathoulas, and A. Kakarountas, "Supply Current Monitoring for Anomaly Detection on IoT Devices," in *Proceedings of the 21st Pan-Hellenic Conference on Informatics*, 2017, pp. 1-2.
- [25] D. Myridakis, G. Spathoulas, A. Kakarountas, D. Schoinianakis, and J. Lueken, "Anomaly detection in iot devices via monitoring of supply current," in *2018 IEEE 8th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*, 2018, pp. 1-4.
- [26] D. Myridakis, G. Spathoulas, A. Kakarountas, D. Schinianakis, and J. Lueken, "Monitoring Supply Current Thresholds for Smart Device's Security Enhancement," in *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2019, pp. 224-227.
- [27] D. Myridakis, G. Spathoulas, A. Kakarountas, D. Schoinianakis, and J. Lueken, "Mimicking Biometrics on Smart Devices and Its Application in IoT Security for Health Systems," in *IoT and ICT for Healthcare Applications*, ed: Springer, 2020, pp. 175-189.

ΗΛΕΚΤΡΟΝΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Microchip. (2019, 10-12). *Microchip.com*. Available: <https://www.microchip.com>
- [6] Arduino. (2019, 10-10). *Arduino.cc*. Available: <https://www.arduino.cc>
- [8] Raspberrypi. (2019, 10-11). *Raspberrypi.org*. Available: <https://www.raspberrypi.org>
- [9] Sparkfun. (2019, 10-10). *Sparkfun.com*. Available: <https://www.sparkfun.com>
- [10] OrangePi. (2019, 10-11). *OrangePi.org*. Available: <http://www.orangePi.org>
- [11] Alldatasheet. (2019, 10-14). *Electronic Components Datasheet*. Available: <https://www.alldatasheet.com>
- [12] fritzing. (2019, 9-9). *fritzing.org*. Available: <https://fritzing.org>
- [13] kicad. (2019, 11-12). *kicad.org*. Available: <https://kicad.org/>
- [14] geda. (2019, 11-24). *geda-project.org*. Available: <http://www.geda-project.org>
- [15] 4pcb. (2019, 10-10). *4pcb.com*. Available: <https://www.4pcb.com/free-pcb-layout-software>
- [16] tinycad. (2019, 10-24). *tinycad.net*. Available: <https://www.tinycad.net>
- [17] bayareacircuits. (2019, 10-27). *bayareacircuits.com*. Available: <https://bayareacircuits.com/pcb-design-layout-software-custom>
- [18] multisim. (2019, 10-27). *multisim.com*. Available: <https://www.multisim.com>
- [19] ni. (2019, 10-16). *ni.com*. Available: <https://www.ni.com>
- [20] autodesk. (2020, 8-5). *autodesk.com*. Available: <https://www.autodesk.com>
- [21] Alltransistors. (2019, 10-15). *Data sheets for transistors*. Available: <https://alltransistors.com>
- [22] Datasheets360. (2019, 10-14). *Source of Electronic Component Datasheets*. Available: <https://www.datasheets360.com>
- [23] FTDIchip. (2019, 10-15). *FTDIchip.com*. Available: <https://www.ftdichip.com>