



**ΑΕΙ ΠΕΙΡΑΙΑ Τ.Τ.
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ Τ.Ε.**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Ανάπτυξη συστήματος μέτρησης του
μονοξειδίου του άνθρακα και καταχώρηση
των δεδομένων μέσω datalogger.**

Σταμάτης Ευστάθιος ΑΜ: 38350

Εισηγητές: Γιαννακόπουλος Παναγιώτης, Καθηγητής
Νικολόπουλος Δημήτρης, Αν. Καθηγητής

ΑΙΓΑΛΕΩ ΙΑΝΟΥΑΡΙΟΣ, 2016

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Ανάπτυξη συστήματος μέτρησης του μονοξειδίου του άνθρακα και
καταχώρηση των δεδομένων σε datalogger.**

Σταμάτης Ευστάθιος

A.M. 38350

Εισηγητές:

Γιαννακόπουλος Παναγιώτης, Καθηγητής

Νικολόπουλος Δημήτρης, Αν. Καθηγητής

Εξεταστική επιτροπή: Γιαννακόπουλος Παναγιώτης, Καθηγητής

Νικολόπουλος Δημήτρης, Αν. Καθηγητής

Αθανάσιος Γ. Παλιατσός, Καθηγητής

Ημερομηνία εξέτασης: 8 Ιανουαρίου 2016

ΕΥΧΑΡΙΣΤΙΕΣ

Για την ολοκλήρωση της παρούσας πτυχιακής εργασίας θα ήθελα να ευχαριστήσω αρχικά τους επιβλέποντες καθηγητές μου, Κο Γιαννακόπουλο Παναγιώτη και Κο Νικολόπουλο Δημήτριο για την καθοδήγησή τους ως προς την οργάνωση και την εκπόνηση της πτυχιακής μου εργασίας, αλλά και ως προς την εκπαιδευτική μου εμπειρία.

Θα ήθελα επίσης να ευχαριστήσω, όλους εκείνους στους οποίους απευθύνθηκα, προκειμένου να λάβω τις απαραίτητες πληροφορίες και υλικό, για την περαιτέρω διεξαγωγή της εργασίας μου τόσο ως προς το πρακτικό όσο και ως προς το θεωρητικό της μέρος.

ΠΕΡΙΛΗΨΗ

Η παρούσα διπλωματική εργασία, μέσω του πρακτικού αλλά και του θεωρητικού της μέρους, πραγματεύεται την ανάπτυξη συστήματος μέτρησης του μονοξειδίου του άνθρακα καθώς και την καταχώρηση των δεδομένων μέσω καταχωρητή δεδομένων data logger. Αρχικά, γίνεται αναφορά σε σημαντικές πληροφορίες, όπως είναι η ιστορία, τα χαρακτηριστικά αλλά και η λειτουργία των μικροελεγκτών και της πλακέτας Arduino που χρησιμοποιήθηκε στο πρακτικό μέρος της εργασίας. Στη συνέχεια, επισημαίνεται λεπτομερώς ο ορισμός, οι ιδιότητες καθώς και τα χαρακτηριστικά του μονοξειδίου του άνθρακα. Τέλος, δίνεται σημασία στην εικόνα του κυκλώματος αλλά και στην αναφορά και καταγραφή του υλικού μέρους που χρησιμοποιήθηκε για το πρακτικό μέρος της πτυχιακής εργασίας, όπως είναι ο καταχωρητής δεδομένων, ο αισθητήρας μονοξειδίου του άνθρακα, η οθόνη καθώς και η αναφορά του κώδικα του προγράμματος το οποίο φορτώθηκε στο μικροελεγκτή της πλακέτας Arduino.

ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ: Προγραμματισμός Μικροελεγκτών

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Μικροελεγκτές, Arduino, Μονοξείδιο του Άνθρακα, Data logger

ABSTRACT

This thesis, in its practical and theoretical part, deals with the development of a carbon monoxide measurement system as well as the upload of data through a data logger. Firstly, it refers to important information, such as the history, the features and the function of microcontrollers and the Arduino board which was used in the practical part. Then, it presents the definition, the properties along with the features of carbon monoxide. Lastly, importance is given to the illustration of the circuit as well as to the reference and recording of the hardware used for the practical part of the thesis, that is the data logger, the carbon monoxide sensor, the screen and also we refer to the program code which was uploaded on the microcontroller of the Arduino board.

SCIENTIFIC AREA: Microcontrollers Programming

KEYWORDS: Microcontrollers, Arduino, Carbon Monoxide, Data logger

ΠΕΡΙΕΧΟΜΕΝΑ

Ευχαριστίες.....	5
Περίληψη-Abstract.....	7
Περιεχόμενα.....	9
Κατάλογος σχημάτων.....	12
Κατάλογος Πινάκων.....	14
Συνομογραφίες.....	15

Κεφάλαιο 1. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

1.1 Μικροελεγκτές.....	18
1.1.1 Ιστορία των μικροεπεξεργαστών-μικροελεγκτών.....	18
1.1.2 Η εξέλιξη των μικροεπεξεργαστών.....	20
1.1.3 Περιγραφή του μικροελεγκτή.....	25
1.1.4 Η λειτουργία του μικροελεγκτή.....	26
1.1.5 Βασικά χαρακτηριστικά που καθιστούν προτιμότερο τον μικροελεγκτή.....	27
1.1.6 Διαφορές μεταξύ μικροελεγκτή και μικροεπεξεργαστή.....	29
1.1.7 Διαδεδομένες κατηγορίες μικροελεγκτών.....	30
1.1.8 Εργαλεία ανάπτυξης και γλώσσες προγραμματισμού.....	32
1.1.9 Κατασκευαστές μικροελεγκτών.....	32
1.2 Arduino.....	33
1.2.1 Ιστορική αναδρομή.....	33

1.2.2 Τι είναι το Arduino.....	34
1.2.3 Υλικό – Hardware.....	34
1.2.4 ATmega328 – Η καρδιά του Arduino.....	35
1.2.5 Μοντέλα Arduino.....	36
1.2.6 Arduino Shields.....	37
1.2.7 Τεχνικά χαρακτηριστικά του Arduino UNO.....	38
1.2.8 Είσοδοι – Έξοδοι.....	39
1.2.9 Τροφοδοσία.....	43
1.2.10 Ενσωματωμένα κουμπιά και LED.....	45
1.2.11 Λογισμικό - Arduino IDE και σύνδεση με τον υπολογιστή.....	46
1.2.12 Γλώσσα προγραμματισμού.....	49
1.3 Το μονοξείδιο του άνθρακα.....	55
1.3.1 Γενικά.....	55
1.3.2 Επιπτώσεις στην υγεία.....	59
<u>ΚΕΦΑΛΑΙΟ 2.</u> ΥΛΙΚΟ ΜΕΡΟΣ ΣΥΣΤΗΜΑΤΟΣ.....	62
2.1 SD Datalogger Shield.....	64
2.1.1 Τι είναι ο καταχωρητής δεδομένων (Data logger).....	64
2.1.2 Adafruit Assembled Data Logging shield for Arduino.....	65
2.1.2.1 Τροφοδοσία.....	71
2.1.2.2 Το Ρολόι Πραγματικού Χρόνου (Real Time Clock).....	72
2.1.2.3 Η διεπαφή της κάρτας SD.....	73

2.1.3 Οι ακροδέκτες του Arduino που χρησιμοποιεί το datalogger shield.....	74
2.2 Ο αισθητήρας μονοξειδίου του άνθρακα MQ-7.....	74
2.3 Η οθόνη LCD.....	81
2.3.1 Σύνδεση της οθόνης μέσω του πρωτοκόλλου I2C.....	84
ΠΑΡΑΡΤΗΜΑ Α'.....	87
ΠΑΡΑΡΤΗΜΑ Β'.....	100
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	111

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

- Εικόνα 1.1: Intel 4004, 8080, 8086/8088, 80286, 80386, 80486, Pentium, II, III, IV.....25
- Εικόνα 1.2: Οι ακροδέκτες των μικροελεγκτών ATmega168/328.....36
- Εικόνα 1.3: Μοντέλα μικροελεγκτών Arduino.....37
- Εικόνα 1.4: Περιγραφή των ακροδεκτών και των εξαρτημάτων του Arduino.....40
- Εικόνα 1.5: Οι ψηφιακοί ακροδέκτες του Arduino UNO.....42
- Εικόνα 1.6: Οι αναλογικοί ακροδέκτες του Arduino UNO.....43
- Εικόνα 1.7: Εξωτερικές πηγές τροφοδοσίας για το Arduino.....44
- Εικόνα 1.8: Είσοδοι-Εξοδοί τροφοδοσίας του Arduino.....44
- Εικόνα 1.9: Το περιβάλλον ανάπτυξης του Arduino IDE.....47
- Εικόνα 1.10: Οι λειτουργίες των εικονιδίων του Arduino IDE.....57
- Εικόνα 1.11: Εγκατάσταση των drivers της σειριακής θύρας του Arduino.....49
- Εικόνα 1.12: Σχηματικό του Arduino UNO.....55
- Εικόνα 2.1: Σχέδιο κυκλώματος Arduino - εξαρτημάτων.....67
- Εικόνα 2.2: Adafruit data logging shield.....69
- Εικόνα 2.3: Περιγραφή των εξαρτημάτων του datalogger shield.....71
- Εικόνα 2.4: Πρόσθετα led και ακροδέκτες του shield.....72
- Εικόνα 2.5: Το σχηματικό του datalogger shield.....74
- Εικόνα 2.6: Το σχηματικό του κυκλώματος της τροφοδοσίας του shield.....74
- Εικόνα 2.7: Το σχηματικό του κυκλώματος για το ρολόι πραγματικού χρόνου.....75
- Εικόνα 2.8: Το σχηματικό του κυκλώματος της διεπαφής της κάρτας SD.....86
- Εικόνα 2.9: Ο αισθητήρας MQ-7.....78
- Εικόνα 2.10: Πληροφορίες κατασκευής MQ-7.....79

- Εικόνα 2.11: Βασικό κύκλωμα MQ-7.....80
- Εικόνα 2.12: Σταθεροποιητής τάσης KA278RA05C.....82
- Εικόνα 2.13: Solid state relay LH1546.....83
- Εικόνα 2.14: Adafruit LCD RGB 16x2.....85
- Εικόνα 2.15: Σχηματικό της οθόνης LCD.....86

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

- Πίνακας 1.1: Οι δεκαετίες εξέλιξης των μικροεπεξεργαστών.....20
- Πίνακας 1.2: Arduino Shields.....38
- Πίνακας 1.3: Τεχνικά χαρακτηριστικά του Arduino UNO.....39
- Πίνακας 1.4: Σύνοψη περιγραφή των λειτουργιών των ακροδεκτών του Arduino.....41
- Πίνακας 1.5: Βασικές εντολές και συναρτήσεις της γλώσσας του Arduino.....50
- Πίνακας 1.6: Αναλογία μίγματος μονοξειδίου του άνθρακα σε διάφορες περιοχές.....63
- Πίνακας 2.1: Τεχνικά χαρακτηριστικά MQ-7.....81

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

ΗΠΑ Ηνωμένες Πολιτείες της Αμερικής

ΚΜΕ Κεντρική Μονάδα Επεξεργασίας

AC Alternating Current

ADC Analog to Digital Converter

ALU Arithmetic Logic Unit

AREF Analog Reference

CAN Controller Area Network

CPU Central Processing Unit

CO Carbon Monoxide

CD Card Detect

CS Chip Select

D Debye

DC Direct Current

DIP Dual In-line Package

E Enable

EEPROM Electrically Erasable Programmable Read Only Memory

FAT16, FAT32 File Allocation Table

FTDI Future Technology Devices International

GND Ground

GCC GNU Compiler Collection

I / O Input / Output

I²C Inter-Integrated Circuit

IC Integrated Circuit

IDE Integrated Development Environment

ILP Instruction Level Parallelism

IPC Instructions Per Cycle

ISA Instruction Set Architecture

ISP Instruction Set Processors

LCD Liquid Crystal Display

LED Light Emitting Diode

MIPS Microprocessor without Interlocked Pipeline Stages

MMC Multi Media Card

MLP Memory Level Parallelism

MMX Multi Media eXtension, Multiple Math eXtension, Matrix Math eXtension

OSHA Occupational Safety & Health Administration

PWM Pulse Width Modulation

PPM Parts Per Million

PC Personal Computer

PROM Programmable Read Only Memory

RAM Random Access Memory

RGB Red Green Blue

RISC Reduced Instruction Set Computing

ROM Read Only Memory

RS Register Select

RS-232 Recommended Standard 232

RT-Linux Real Time Linux

RTC Real Time Clock

RW Read Write

RX Receive

SCL Serial Clock

SD Secure Digital

SDA Serial Data

SPI Serial Peripheral Interface

SRAM Static Random Access Memory

SQ Square wave

TLP Thread Level Parallelism

TTL Transistor Transistor Logic

TX Transmit

UART Universal Asynchronous Receiver Transmitter

USB Universal Serial Bus

WP Write Protect

ΚΕΦΑΛΑΙΟ 1 – ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

1.1 Μικροελεγκτές

1.1.1 Ιστορία των μικροεπεξεργαστών-μικροελεγκτών

Η επιθυμία των κατασκευαστικών συστημάτων να δημιουργήσουν συστήματα με περισσότερες δυνατότητες και με μικρότερο μέγεθος οδήγησε στην ανάγκη για ενσωμάτωση όλων των λειτουργιών ενός υπολογιστή σε ένα ή μερικά ολοκληρωμένα κυκλώματα. Η ανάπτυξη της τεχνολογίας που χρησιμοποιήθηκε για την κατασκευή των κυκλωμάτων αυτών είχε σαν αποτέλεσμα την ανάπτυξη του μικροεπεξεργαστή. Στην σχετικά σύντομη διάρκεια ζωής του, ο μικροεπεξεργαστής (microprocessor) έχει κάνει τεράστιες προόδους.

Η απόδοσή του έχει βελτιωθεί αισθητά αφού διπλασιάζεται περίπου κάθε 18 μήνες. Οι μικροεπεξεργαστές είναι υπεύθυνοι για την έμπνευση και τη δημιουργία μερικών από τις μεγαλύτερες καινοτομίες στα συστήματα υπολογιστών. Αυτές οι καινοτομίες περιλαμβάνουν τους embedded μικροελεγκτές, τους προσωπικούς υπολογιστές, τους σύγχρονους σταθμούς εργασίας, συσκευές χειρός και κινητές συσκευές (όπως τους επεξεργαστές των κινητών τηλεφώνων), servers (εξυπηρετητές) εφαρμογών και αρχείων, web servers για το internet, υπερυπολογιστές χαμηλού κόστους και ευρείας κλίμακας δίκτυα υπολογιστών.

Σήμερα πωλούνται κάθε χρόνο πάνω από 100 εκατομμύρια μικροεπεξεργαστές στις αγορές κινητής τηλεφωνίας, επεξεργαστές για εργασίες γραφείου και server. Αν συνυπολογίσουμε και τους embedded μικροεπεξεργαστές και μικροελεγκτές τότε ο συνολικός αριθμός των μικροεπεξεργαστών που πωλούνται κάθε χρόνο υπερβαίνει κατά πολύ το ένα δισεκατομμύριο.

Οι μικροεπεξεργαστές είναι επεξεργαστές συνόλου εντολών (Instruction Set Processors, ISPs). Ένας ISP εκτελεί εντολές ενός προκαθορισμένου συνόλου εντολών. Η λειτουργικότητά του εξαρτάται ανάλογα από το σύνολο εντολών που είναι ικανός να εκτελέσει ο μικροεπεξεργαστής. Σε αυτό το σύνολο των εντολών κωδικοποιούνται όλα τα προγράμματα που τρέχουν σε έναν μικροεπεξεργαστή. Αυτό το προκαθορισμένο σύνολο εντολών ονομάζεται επίσης αρχιτεκτονική συνόλου εντολών (instruction set architecture, ISA).

Το ISA χρησιμεύει ως μία διασύνδεση ανάμεσα στο λογισμικό (software) και το υλικό (hardware), δηλαδή ανάμεσα στα προγράμματα και τους επεξεργαστές. Σε ορολογία της μεθοδολογίας σχεδίασης επεξεργαστών, το ISA είναι ο καθορισμός της σχεδίασης ενώ ο επεξεργαστής ή ο ISP είναι η υλοποίηση της σχεδίασης. Με τον όρο μικροελεγκτή αναφερόμαστε σε έναν τύπο επεξεργαστή ο οποίος μπορεί να λειτουργήσει με ελάχιστα εξωτερικά εξαρτήματα λόγω των πολλών ενσωματωμένων υποσυστημάτων που διαθέτει. Για τον λόγο αυτό θεωρείτε ως παραλλαγή ενός μικροεπεξεργαστή.

Ένας μικροελεγκτής είναι ένα ενσωματωμένο τσιπ (ολοκληρωμένο κύκλωμα) που αποτελεί συχνά μέρος ενός συστήματος. Όπως και ένας απλός τυπικός υπολογιστής έτσι και ο μικροελεγκτής περιλαμβάνει CPU, RAM, ROM για αποθήκευση δεδομένων και λογισμικού αντίστοιχα, μνήμη flash για μόνιμη αποθήκευση, θύρες εισόδου/εξόδου, μετατροπέα αναλογικού σε ψηφιακό σήμα και το αντίστροφο και τέλος timers. Καθώς είναι σχεδιασμένος να εκτελεί μόνο μία συγκεκριμένη εργασία για τον έλεγχο ενός απλού συστήματος, είναι πολύ μικρότερος και απλούστερα σχεδιασμένος ώστε να μπορεί να περιλαμβάνει όλες τις λειτουργίες που απαιτούνται σε ένα μόνο ολοκληρωμένο κύκλωμα.

Χρησιμοποιείται ευρέως σε όλα τα ενσωματωμένα συστήματα ελέγχου χαμηλού και μεσαίου κόστους όπως για παράδειγμα αυτά που χρησιμοποιούνται σε αυτοματισμούς, ηλεκτρονικά καταναλωτικά προϊόντα (από ψηφιακές φωτογραφικές μηχανές έως παιχνίδια), ηλεκτρικές συσκευές και κάθε είδους αυτοκινούμενα τροχοφόρα οχήματα. Στις μέρες μας δεν θα ήταν υπερβολή να πούμε ότι η χρήση μικροελεγκτών είναι καθολική για το

λόγο ότι κάθε προϊόν το οποίο αλληλεπιδρά με ένα χρήστη περιλαμβάνει ένα μικροελεγκτή, ο οποίος παίζει το ρόλο του «εγκεφάλου» των ηλεκτρονικών κυκλωμάτων.

1.1.2 Η εξέλιξη των μικροεπεξεργαστών

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

Το έτος 2011 σήμανε την τεσσαρακοστή επέτειο της γέννησης των μικροεπεξεργαστών. Μικροεπεξεργαστές υψηλών επιδόσεων (high-end microprocessors), οι οποίοι αποτελούνται από εκατοντάδες εκατομμύρια τρανζίστορ με συχνότητα ρολογιού που αγγίζει τα 4 GHz και πολλαπλούς πυρήνες, και είναι οι δομικές μονάδες για τα συστήματα υπερυπολογιστών και πανίσχυρων συστημάτων πελάτη-εξυπηρετητή (client-server systems) που υπάρχουν στο Internet. Στη σημερινή εποχή η συχνότητα ρολογιού των μικροεπεξεργαστών υπερβαίνει τα 3 GHz και τους τέσσερις πυρήνες και αποτελείται από αρκετές δεκάδες εκατομμύρια τρανζίστορ.

Οι τέσσερις δεκαετίες της ιστορίας των μικροεπεξεργαστών λένε μία πραγματικά αξιοπρόσεκτη ιστορία όσον αφορά την τεχνολογική πρόοδο στην βιομηχανία υπολογιστών. Αυτή η πρόοδος φαίνεται καλύτερα στον πίνακα 1 που ακολουθεί:

Πίνακας 1.1: Οι δεκαετίες εξέλιξης των μικροεπεξεργαστών.

	1971-1981	1981-1991	1991-2001	2001-2011
Πλήθος τρανζίστορ (transistor count)	2K-100K	100K-1M	1M-100M	100M-2B
Συχνότητα ρολογιού (clock frequency)	0.1-3 MHz	3-30 MHz	30 MHz-1GHz	1-5 GHz
Εντολές/κύκλο (Instructions/cycle, IPC)	0.1	0.1-0.9	0.9-1.9	1.9-2.9

Η εξέλιξη των μικροεπεξεργαστών έχει κατά βάση ακολουθήσει τον περίφημο νόμο του Moore (Moore's law), που παρατηρήθηκε από τον Gordon Moore το 1965 και σύμφωνα με τον οποίο ο αριθμός των συσκευών που μπορούν να ολοκληρωθούν (με την έννοια της ολοκλήρωσης στην ορολογία της σχεδίασης ολοκληρωμένων κυκλωμάτων) σε ένα απλό κομμάτι πυριτίου διπλασιάζεται κάθε 18-24 μήνες. Σε λιγότερο από 30 χρόνια μάλιστα, ο αριθμός των τρανζίστορ σε ένα chip μικροεπεξεργαστή έχει αυξηθεί κατά πάνω από 4 τάξεις μεγέθους όπως φαίνεται και στον πίνακα. Στην ίδια χρονική περίοδο, η απόδοση του μικροεπεξεργαστή έχει αυξηθεί κατά πάνω από 5 τάξεις μεγέθους. Επίσης, στις τελευταίες δύο δεκαετίες η απόδοση του μικροεπεξεργαστή διπλασιάζεται κάθε 18 μήνες, ή διαφορετικά εκατονταπλασιάζεται (x100) σε κάθε δεκαετία. Αυτή η εντυπωσιακή βελτίωση της απόδοσης των μικροεπεξεργαστών αποτελεί μοναδικό φαινόμενο και δεν έχει συναντηθεί σε καμία άλλη βιομηχανία.

Κατά τη διάρκεια της πρώτης δεκαετίας η έλευση του 4bit μικροεπεξεργαστή οδήγησε γρήγορα στην παρουσίαση του 8bit μικροεπεξεργαστή. Αυτοί εξελίχθηκαν σε μικροελεγκτές που χρησιμοποιήθηκαν σε embedded εφαρμογές, από μηχανές πλυσίματος μέχρι ανελκυστήρες και μηχανές jet. Ο 8bit μικροεπεξεργαστής έγινε επίσης η καρδιά μίας νέας διάσημης υπολογιστικής πλατφόρμας, γνωστής ως προσωπικός υπολογιστής (Personal Computer, PC) και εισήγαγε την εποχή των PCs.

Η δεκαετία του 1980 υπήρξε μάρτυρας πολλών σημαντικών αλλαγών στην αρχιτεκτονική και την μικροαρχιτεκτονική των 32bit μικροεπεξεργαστών. Τα προβλήματα που αφορούν τη σχεδίαση του συνόλου εντολών έγιναν το σημαντικότερο αντικείμενο των ακαδημαϊκών και βιομηχανικών ερευνών. Το pipelining των εντολών και οι γρήγορες κρυφές μνήμες (cache memories) έγιναν αναντικατάστατες τεχνικές μικρο-αρχιτεκτονικής.

Κατά τη διάρκεια της δεκαετίας του 1990 οι μικροεπεξεργαστές έγιναν η πιο πανίσχυρη και δημοφιλής μορφή υπολογιστών. Η συχνότητα ρολογιού των ταχύτερων μικροεπεξεργαστών ξεπέρασαν τις συχνότητες ρολογιού των ταχύτερων υπερυπολογιστών. Οι προσωπικοί υπολογιστές και οι σταθμοί εργασίας έγιναν σημαντικά και αναντικατάστατα εργαλεία της παραγωγικότητας και των επικοινωνιών. Χρησιμοποιήθηκαν ιδιαίτερα επιθετικές τεχνικές μικροαρχιτεκτονικής για την επίτευξη απόδοσης μικροεπεξεργαστών σε επίπεδα που δεν είχαν επιτευχθεί ποτέ μέχρι τότε. Επίσης έγιναν δημοφιλείς βαθιά pipelined μηχανές (δηλαδή μηχανές με σωληνώσεις πολλών σταδίων), ικανές να επιτύχουν εξαιρετικά υψηλές συχνότητες ρολογιού και να εκτελέσουν πολλές εντολές ανά κύκλο ρολογιού. Ακόμη χρησιμοποιήθηκαν πολύ επιθετικές τεχνικές πρόβλεψης διακλαδώσεων όπως επίσης και η out-of-order εκτέλεση των εντολών ώστε να μειωθούν στο ελάχιστο οι χαμένοι κύκλοι ρολογιού λόγω καθυστερήσεων του pipeline (pipeline stalls).

Σήμερα φαίνεται ότι η ροπή ανάπτυξης των μικροεπεξεργαστών δεν δείχνει σημάδια ελάττωσης. Οι περισσότεροι τεχνολόγοι συμφωνούν ότι ο νόμος του Moore θα συνεχίσει να ισχύει για τουλάχιστον 10 με 15 χρόνια. Η εστίαση που τις πρώτες τρεις δεκαετίες ήταν μόνο στον παραλληλισμό στο επίπεδο της εντολής (Instruction-level parallelism, ILP) έχει ήδη περάσει στον παραλληλισμό στο επίπεδο του νήματος (Thread-level parallelism, TLP) και στον παραλληλισμό στο επίπεδο της μνήμης (Memory-level parallelism, MLP). Επίσης πολλά ζητήματα που αφορούσαν παραδοσιακά την «μακροαρχιτεκτονική» γίνονται ζητήματα που αφορούν τη μικροαρχιτεκτονική (δηλαδή χαρακτηριστικά που αφορούσαν μεγάλα συστήματα τώρα υλοποιούνται πάνω σε ένα chip). Η κατανάλωση ισχύος έχει γίνει επίσης ένας πολύ σημαντικός παράγοντας της απόδοσης και απαιτούνται νέες λύσεις σε όλα τα επίπεδα της ιεραρχίας σχεδίασης, που περιλαμβάνουν την διαδικασία παραγωγής, τη λογική σχεδίαση, τη σχεδίαση σε επίπεδο μικροαρχιτεκτονικής και το run-time περιβάλλον του λογισμικού, έτσι ώστε να διατηρηθούν τα ίδια επίπεδα βελτίωσης της απόδοσης που επιτεύχθηκαν κατά τις προηγούμενες δεκαετίες.

Παρακάτω θα αναφερθούν οι σημαντικότερες χρονολογίες-σταθμοί στην ιστορία των μικροεπεξεργαστών:

- Στις 15 Νοεμβρίου του έτους 1971 η Intel παρουσιάζει τον πρώτο μικροεπεξεργαστή, τον Intel 4004 από τον Ted Hoff και τον συνεργάτη του Stan Mazor. Σχεδιάστηκε σε μορφή dual in-line package (DIP) και ήταν ο πρώτος εμπορικά διαθέσιμος επεξεργαστής υπολογιστών. Ο Intel 4004 ήταν ένας 4bit επεξεργαστής ο οποίος αποτελούνταν από 2.300 τρανζίστορ με συχνότητα ρολογιού 108KHz, εκτελούσε 60.000 πράξεις το δευτερόλεπτο και μπορούσε να αναγνωρίσει 640 bytes μνήμης. Αρχικά εφαρμόστηκε για τη δημιουργία αριθμομηχανών ενώ τον Νοέμβριο του 1971 η Intel ανακοίνωσε τον πρώτο μικροϋπολογιστή. Μέσα στην επόμενη χρονιά εμφανίζεται ο διάδοχός του ο 8008.

- Το 1974 έχουμε την εμφάνιση του 8bit μικροεπεξεργαστή Intel 8080 που ήταν αποτέλεσμα εξέλιξης του 8008. Αποτελούνταν από 6.000 τρανζίστορ με συχνότητα λειτουργίας 2MHz. Την ίδια περίοδο η Motorola παρουσίασε τον 6800 που χρησιμοποιήθηκε σε υπολογιστές, σε όλα τα pinball-παιχνίδια καθώς και σε βιομηχανικές συσκευές ελέγχου. Είχε 4.000 τρανζίστορ, 78 εντολές, σήμα χρονισμού στα 1 ή 2 MHz και 16bit πλάτος διαύλου διευθύνσεων.

- Το 1975 η Zilog φτιάχνει τον Z80 έναν 8bit μικροεπεξεργαστή βασισμένο στον 8080 του οποίου η γλώσσα μηχανής είναι υπερσύνολο αυτής του Intel 8080. Το σήμα χρονισμού του ήταν στα 3.5 MHz με 16 bit πλάτος διευθύνσεων ενώ μπορούσε να αναγνωρίσει 64Kbytes μνήμης. Κατά την δεκαετία του 80 είχε την μεγαλύτερη δημοτικότητα καθώς επικεντρώθηκε στο χαμηλό κόστος σε συνδυασμό με τη μικρή συσκευασία, τις χαμηλές απαιτήσεις και τον συνυπολογισμό των στοιχείων κυκλώματος που κανονικά θα έπρεπε να παρασχεθούν σε ένα χωριστό τσιπ.

- Το 1978 εμφανίζονται οι πρώτοι 16bit μικροεπεξεργαστές. Η Intel παρουσιάζει τον 8086/8088 του οποίου η συχνότητα λειτουργίας έχει ανέβει στα 10MHz και περιέχει 29.000 τρανζίστορ. Αντίστοιχα η Motorola εμφανίζει τον 68000 με συχνότητα λειτουργίας 8MHz και περιέχει 68.000

τρανζίστορ. Ο συνδυασμός της υψηλής ταχύτητας, του μεγάλου χώρου αποθήκευσης (16Mbytes) και του αρκετά χαμηλού κόστους, τον έκανε τον δημοφιλέστερο μικροεπεξεργαστή με αποτέλεσμα να χρησιμοποιηθεί στους υπολογιστές Apple Lisa και Macintosh.

- Το 1982 η Intel δημιούργησε τον 80286 με συχνότητα λειτουργίας αρχικά στα 6 και έπειτα 12.5MHz και περιέχει 134.000 τρανζίστορ. Περιλάμβανε δίαυλο δεδομένων 16bit, δίαυλο διευθύνσεων 24bit και μπορούσε να αναγνωρίσει μέχρι 16Mbytes μνήμης. Είχε τη δυνατότητα να λειτουργεί στην κατάσταση protected mode (προστατευμένη κατάσταση λειτουργίας). Μετά την επιτυχία της Motorola με τον 68000, οδηγήθηκε στον σχεδιασμό του MC68010 με αποτέλεσμα την προσθήκη υποστήριξης της εικονικής μνήμης.

- Το 1985 εμφανίζονται οι πρώτοι 32bit μικροεπεξεργαστές. Από τη μια ο Intel 80386 όπου περιέχει 275.000 τρανζίστορ και συχνότητα λειτουργίας 33 MHz. Η μνήμη που δεχόταν ήταν μέχρι 4 Gbytes. Από την άλλη ο MC86020 της Motorola είχε 200.000 τρανζίστορ και συχνότητα λειτουργίας στα 16MHz. Έγινε ιδιαίτερα δημοφιλής στη microcomputer Unix αγορά ενώ πολλές μικρές επιχειρήσεις παρήγαγαν τα συστήματα desktop.

- Το 1989 κάνει την εμφάνισή του ο 32bit μικροεπεξεργαστής Intel 80486 με 1.200.000 τρανζίστορ και 50MHz συχνότητα λειτουργίας.

- Το 1993 εμφανίζεται ο Intel Pentium της οικογένειας P5 ο οποίος περιέχει 3.100.000 τρανζίστορ και λειτουργούσε στα 60 και 66MHz. Την ίδια περίοδο η Digital παρουσιάζει τον πρώτο 64bit μικροεπεξεργαστή Alpha.

- Το 1997 η Intel εισήγαγε τον μικροεπεξεργαστή Pentium II με τεχνολογία MMX για την υποστήριξη πολυμέσων. Είχε 7.500.000 τρανζίστορ και η συχνότητα λειτουργίας του ήταν στα 300 MHz.

- Το 1999 η Intel ανακοίνωσε τον Pentium III με 9.500.000 τρανζίστορ και η συχνότητα λειτουργίας του βρισκόταν στα 450MHz (και σήμερα έχει φτάσει στα 1.13GHz)

■ Το 2000 εμφανίστηκε ο Pentium IV όπου ήταν σχεδιασμένος σύμφωνα με την μικροαρχιτεκτονική NetBurst. Η συχνότητα λειτουργίας του έφτασε στο 1GHz αλλά χωρίς αυτό να παρέχει σημαντική αύξηση της ταχύτητας επεξεργασίας σε σχέση με τον Pentium III.

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



Εικόνα 1.1: Intel 4004, 8080, 8086/8088, 80286, 80386, 80486, Pentium, II, III, IV

1.1.3 Περιγραφή του μικροελεγκτή

Ο μικροελεγκτής είναι ένας μικρός υπολογιστής σε ένα ολοκληρωμένο κύκλωμα το οποίο περιέχει επεξεργαστή, μνήμη, διάφορα περιφερειακά κυκλώματα καθώς επίσης και θύρες εισόδου/εξόδου για επικοινωνία με εξωτερικές συσκευές. Η μνήμη, όπου φορτώνεται κάποιο πρόγραμμα, της οποίας ο τύπος ποικίλει ανάλογα με τον μικροελεγκτή (Flash, ROM, PROM και λοιπές), είναι ενσωματωμένη στο ολοκληρωμένο όπως επίσης και ένα μικρό ποσό μνήμης RAM. Οι μικροελεγκτές είναι σχεδιασμένοι για ενσωματωμένες εφαρμογές σε αντίθεση με τους μικροεπεξεργαστές που χρησιμοποιούνται σε υπολογιστές ή για άλλες εφαρμογές γενικής χρήσης.

Οι μικροελεγκτές χρησιμοποιούνται σε προϊόντα και συσκευές που ελέγχονται αυτόματα, όπως σε συστήματα ελέγχου μηχανής αυτοκινήτου, σε εμφυτεύσιμες ιατρικές συσκευές, τηλεχειριστήρια, μηχανήματα γραφείου, ηλεκτρικές συσκευές και εργαλεία, παιχνίδια και άλλα ενσωματωμένα συστήματα. Με τη μείωση του μεγέθους και του κόστους σε σχέση με μια σχεδίαση που χρησιμοποιεί ξεχωριστά μικροεπεξεργαστή, μνήμη και συσκευές εισόδου/εξόδου, οι μικροελεγκτές είναι πιο οικονομικοί αν θέλει κανείς να ελέγξει ψηφιακά ακόμα περισσότερες συσκευές και λειτουργίες. Οι mixed signal μικροελεγκτές είναι συνηθισμένοι επίσης, και ολοκληρώνουν αναλογικά εξαρτήματα που χρειάζονται για τον έλεγχο μη ψηφιακών ηλεκτρονικών συστημάτων.

Κάποιοι μικροελεγκτές μπορεί να χρησιμοποιούν 4bit λέξεις και να λειτουργούν σε συχνότητες ρολογιού μέχρι και μόλις 4 kHz, για χαμηλή κατανάλωση ρεύματος (μονοψήφια milliwatts ή microwatts). Γενικά έχουν την δυνατότητα να διατηρήσουν τη λειτουργία τους μέχρι να γίνει διακοπή από πάτημα κουμπιού ή από άλλη ενέργεια. Η κατανάλωση ρεύματος κατά την λειτουργία «stand by»(το ρολόι της cpu και τα περισσότερα περιφερειακά απενεργοποιημένα) μπορεί να είναι μόλις κάποια nanowatts, κάτι που κάνει πολλούς μικροελεγκτές κατάλληλους για συσκευές μακράς διάρκειας μπαταρίας. Άλλοι μικροελεγκτές μπορεί να λειτουργούν ως δείκτες αποδοτικότητας, όπου χρειάζεται να λειτουργούν περισσότερο σαν επεξεργαστές ψηφιακού σήματος, με μεγαλύτερης ταχύτητας ρολόι και κατανάλωση ρεύματος.

1.1.4 Η λειτουργία του μικροελεγκτή

Υπάρχουν πολλοί μικροελεγκτές και ακόμα περισσότερα προγράμματα που διατίθενται για μικροελεγκτές, παρόλο αυτά οι περισσότεροι από αυτούς έχουν πολλά πράγματα από κοινού. Έτσι, αν μπορέσουμε να μάθουμε να χειριζόμαστε έναν από αυτούς τότε θα μπορούμε να χειριστούμε και τους άλλους επίσης. Ο μικροελεγκτής είναι μια γρήγορη συσκευή, όχι όσο ο

υπολογιστής, έτσι ώστε κάθε εντολή που εκτελείται σε αυτόν να γίνεται με πολύ γρήγορη ταχύτητα. Η λειτουργία του δίνεται παρακάτω:

Όταν ενεργοποιείται η τροφοδοσία, το Μητρώο Λογικού Ελέγχου ενεργοποιεί με τη σειρά του τον ταλαντωτή χαλαζία. Στα πρώτα λίγα χιλιοστά του δευτερολέπτου, ενώ οι πρώτες προετοιμασίες βρίσκονται σε εξέλιξη, οι πυκνωτές παράσιτα φορτίζονται. Όταν η στάθμη της τάσης φτάνει στη μέγιστη τιμή και η συχνότητα του ταλαντωτή χαλαζία γίνεται σταθερή, η διαδικασία της γραφής bits σε ειδικά λειτουργικά μητρώα (SFrs) ξεκινάει. Τα πάντα λαμβάνουν χώρα σύμφωνα με το ρολόι του ταλαντωτή και πάνω από όλα τα ηλεκτρονικά αρχίζουν να δουλεύουν. Όλα αυτά γίνονται σε πολύ λίγα νάνο δευτερόλεπτα. Το PC ή ο μετρητής προγράμματος μηδενίζει τη διεύθυνση της μνήμης προγράμματος. Στη συνέχεια, η διεύθυνση αποστέλλει τις οδηγίες στον αποκωδικοποιητή ο οποίος αποκωδικοποιεί τις εντολές και έτσι τις εκτελεί. Μετά την εκτέλεση μιας εντολής, η διεύθυνση του απαριθμητή προγράμματος αυξάνεται κατά 1 και ως εκ τούτου στέλνει τη διεύθυνση της επόμενης εντολής στον αποκωδικοποιητή εντολών και εκτελεί τις επόμενες οδηγίες.

1.1.5 Βασικά χαρακτηριστικά που καθιστούν προτιμότερο τον μικροελεγκτή

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

- Χαμηλό κόστος. Είναι ένα από τα βασικότερα χαρακτηριστικά που κάποιος σχεδιαστής λαμβάνει υπόψη. Η συνεχής απελευθέρωση στην αγορά μικροελεγκτών από διάφορες εταιρίες βελτίωσαν την ποιότητα αυτών και μείωσαν τις τιμές λόγω του ανταγωνισμού.

- Μικρότερο μέγεθος. Η ολοκλήρωση των βασικών στοιχείων από τα οποία απαρτίζεται μείωσε τις διαστάσεις σε σχέση με τη χρήση των επιμέρους στοιχείων ως σύνολο.
- Χαμηλή κατανάλωση ισχύος. Το γεγονός ότι οι μικροελεγκτές λειτουργούν σε συγκριτικά χαμηλές συχνότητες που φτάνουν τα 32KHz, οδηγεί στην κατανάλωση μικρών ποσών ισχύος της τάξης των mW ακόμα και KW. Επιπλέον έχουν τη δυνατότητα να εισέρχονται σε κατάσταση αναμονής -sleep mode - καταστέλλουν προσωρινά τη λειτουργία της κεντρικής μονάδας επεξεργασίας (ΚΜΕ) και των περιφερειακών, οπότε αυτό μπορεί να γίνει μειώνοντας κατά πολύ την κατανάλωση ισχύος του μικροελεγκτή. Έτσι μπορούν να χρησιμοποιηθούν σε εφαρμογές με αυστηρές απαιτήσεις ως προς αυτήν την παράμετρο.
- Αυτονομία. Αυτό επιτυγχάνεται μέσω της ενσωμάτωσης σύνθετων περιφερειακών υποσυστημάτων όπως μνήμες και θύρες επικοινωνίας. Έτσι πολλοί μικροελεγκτές δεν χρειάζονται κανένα άλλο ολοκληρωμένο κύκλωμα για να λειτουργήσουν.
- Επίτευξη ελέγχου ή μετρήσεων σε πραγματικό χρόνο. Ενώ οι ηλεκτρονικοί υπολογιστές πρέπει να τρέχουν λειτουργικά συστήματα πραγματικού χρόνου (όπως RT-Linux, QNX κ.ά.) για να το επιτύχουν, οι μικροελεγκτές δεν απαιτούν επιπλέον λογισμικό.
- Μειωμένες εκπομπές ηλεκτρομαγνητικών παρεμβολών και μειωμένη ευαισθησία σε αντίστοιχες παρεμβολές από άλλες ηλεκτρικές και ηλεκτρονικές συσκευές. Το πλεονέκτημα αυτό προκύπτει από το μικρότερο αριθμό και μήκος εξωτερικών διασυνδέσεων καθώς και των χαμηλότερων ταχυτήτων λειτουργίας.
- Περισσότεροι διαθέσιμοι ακροδέκτες για ψηφιακές εισόδους-εξόδους (για δεδομένο μέγεθος ολοκληρωμένου κυκλώματος) λόγω της μη δέσμευσής τους για τη σύνδεση εξωτερικών περιφερειακών.
- Η βασική αρχιτεκτονική των μικροελεγκτών δεν διαφέρει από αυτή των κοινών μικροεπεξεργαστών, αν και στους πρώτους συναντάται συχνά η αρχιτεκτονική μνήμης τύπου Harvard, η οποία χρησιμοποιεί διαφορετικές αρτηρίες σύνδεσης της μνήμης προγράμματος και της

μνήμης δεδομένων (π.χ. οι σειρές AVR από την Atmel και PIC από την Microchip). Στους κοινούς μικροεπεξεργαστές συνηθίζεται η ενιαία διάταξη μνήμης τύπου von-Neumann.

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

1.1.6 Διαφορές μεταξύ μικροελεγκτή και μικροεπεξεργαστή

Ο μικροεπεξεργαστής είναι ένα ολοκληρωμένο κύκλωμα (IC) που έχει μόνο τη CPU στο εσωτερικό τους δηλαδή μόνο τις εξουσίες επεξεργασίας, όπως το Pentium I,n,III,IV της Intel κλπ. Αυτοί οι μικροεπεξεργαστές δεν έχουν μνήμη RAM, ROM , και άλλα περιφερειακά στο τσιπ. Για να γίνει λειτουργικός ένας μικροεπεξεργαστής, ο σχεδιαστής του συστήματος πρέπει να τα προσθέσει στο εξωτερικό του.

Εφαρμογές του μικροεπεξεργαστή περιλαμβάνουν τα επιτραπέζια PC, οι φορητοί υπολογιστές, τα σημειωματάρια κ.λπ. Αλλά αυτό δεν ισχύει στην περίπτωση με τους μικροελεγκτές. Ο μικροελεγκτής έχει CPU, επιπροσθέτως με ένα σταθερό ποσό της μνήμης RAM, ROM και άλλα περιφερειακά, όλα ενσωματωμένα σε ένα μόνο chip . Μερικές φορές ονομάζεται ως μίνι υπολογιστής ή υπολογιστής με ένα μόνο chip. Σήμερα διαφορετικοί κατασκευαστές παράγουν μικροελεγκτές με ένα ευρύ φάσμα δυνατοτήτων που είναι διαθέσιμοι σε διαφορετικές εκδόσεις. Ορισμένοι κατασκευαστές είναι οι ATMEL, Microchip, Philips, Motorola κ.λπ.

Οι μικροελεγκτές είναι σχεδιασμένοι για να εκτελούν συγκεκριμένες διεργασίες. Αυτό σημαίνει εφαρμογές, όπου η σχέση των εισροών και εκροών είναι καθορισμένες. Ανάλογα με την εισροή, πρέπει να γίνει η διεργασία ώστε να έχουμε εκροή. Για παράδειγμα, πληκτρολόγιο, ποντίκια, πλυντήριο ρούχων, φούρνος μικροκυμάτων, αυτοκίνητα , ποδήλατα , τηλέφωνο, κινητά τηλέφωνα , ρολόγια , κλπ. Δεδομένου ότι οι εφαρμογές είναι πολύ συγκεκριμένες , χρειαζόμαστε μικρούς πόρους, όπως μνήμη RAM, ROM,

θύρες I / O κλπ και ως εκ τούτου να μπορούν να ενσωματωθούν σε ένα μόνο chip . Αυτό με τη σειρά του μειώνει το μέγεθος και το κόστος.

Οι μικροεπεξεργαστές βρίσκουν εφαρμογές όπου τα καθήκοντα είναι αόριστα όπως η ανάπτυξη λογισμικού , παιχνίδια , ιστοσελίδες , επεξεργασία φωτογραφιών , δημιουργία εγγράφων κλπ. Σε τέτοιες περιπτώσεις, η σχέση μεταξύ εισόδου και εξόδου δεν ορίζεται. Χρειάζονται υψηλό ποσό πόρων, όπως RAM , ROM , θύρες I / O κλπ. Η ταχύτητα ρολογιού του μικροεπεξεργαστή είναι αρκετά υψηλή σε σύγκριση με του μικροελεγκτή. Ενώ οι μικροελεγκτές λειτουργούν από μερικά MHz έως 30 με 50 MHz, οι σημερινοί μικροεπεξεργαστές λειτουργούν πάνω από 1GHz , δεδομένου ότι εκτελούν πολύπλοκα καθήκοντα. Η σύγκριση του μικροελεγκτή και του μικροεπεξεργαστή όσον αφορά το κόστος δεν είναι δικαιολογημένη. Αναμφίβολα ένας μικροελεγκτής είναι πολύ φθηνότερος από ένα μικροεπεξεργαστή. Ωστόσο ένας μικροελεγκτής δεν μπορεί να χρησιμοποιηθεί στη θέση ενός μικροεπεξεργαστή και αντίστοιχα η χρήση ενός μικροεπεξεργαστή δεν συνιστάται στη θέση ενός μικροελεγκτή, καθώς κάνει την εφαρμογή αρκετά δαπανηρή. Ο μικροεπεξεργαστής δεν μπορεί να χρησιμοποιηθεί αυτόνομα. Χρειάζεται άλλα περιφερειακά όπως RAM, ROM , buffer, θύρες I / O κλπ και ως εκ τούτου ένα σύστημα σχεδιασμένο γύρω από έναν μικροεπεξεργαστή, είναι αρκετά δαπανηρό.

1.1.7 Διαδεδομένες κατηγορίες μικροελεγκτών

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

- Μικροελεγκτές -σπάνια 4-bit αλλά συνήθως 8-bit- πολύ χαμηλού κόστους, γενικής χρήσης, με πολύ μικρό αριθμό ακροδεκτών (ακόμη και λιγότερους από 8). Για να μη μπορεί να αντιγραφεί εύκολα το εσωτερικό λογισμικό τους σχεδιάζονται με έμφαση στη χαμηλή

κατανάλωση ισχύος και την αυτάρκεια. Απουσιάζει η δυνατότητα επέκτασης της μνήμης τους. Μερικά μοντέλα είναι ευρέως γνωστά στους ερασιτέχνες ηλεκτρονικούς, όπως πχ οι περισσότεροι μικροελεγκτές των σειρών PIC (Microchip), AVR (Atmel) και 8051 (Intel, Atmel, Dallas κα).

- Μικροελεγκτές -συνήθως 8-bit αλλά και 16 ή 32-bit- χαμηλού κόστους, γενικής χρήσης, με μέτριο έως σχετικά μεγάλο αριθμό ακροδεκτών. Διαθέτουν μεγάλο αριθμό κοινών περιφερειακών, όπως θύρες UART, I²C, SPI ή CAN, μετατροπείς αναλογικού σε ψηφιακό και ψηφιακού σε αναλογικό. Στους κατασκευαστές της Άπω Ανατολής (Ιαπωνία, Κορέα), συνηθίζεται η ενσωμάτωση ελεγκτών οθόνης υγρών κρυστάλλων και πληκτρολογίου. Μερικές φορές παρέχουν δυνατότητα εξωτερικής επέκτασης της μνήμης τους.
- Μικροελεγκτές -κυρίως 32-bit- μέσου κόστους, γενικής χρήσης, με μεγάλο αριθμό ακροδεκτών. Χαρακτηρίζονται από έμφαση στην ταχύτητα εκτέλεσης εντολών, υψηλή αυτάρκεια περιφερειακών και μεγάλες δυνατότητες εσωτερικής ή εξωτερικής μνήμης προγράμματος (FLASH) και RAM. Στο χώρο αυτό έχουν ισχυρή παρουσία οι αρχιτεκτονικές με υψηλή μεταφερσιμότητα λογισμικού (portability) από τον ένα στον άλλο κατασκευαστή. Πχ μεταξύ των μικροελεγκτών τύπου ARM ή MIPS, το σύνολο των βασικών εντολών που αναγνωρίζει η ALU είναι ακριβώς το ίδιο, μειώνοντας έτσι τις μεγάλες αλλαγές στο λογισμικό, όταν στο μέλλον ο πελάτης υιοθετήσει ένα μικροελεγκτή άλλου κατασκευαστή αρκεί, φυσικά, να υποστηρίζει κι αυτός το σύνολο εντολών ARM ή MIPS, αντίστοιχα.
- Μικροελεγκτές εξειδικευμένων εφαρμογών, οι οποίοι ενσωματώνουν συνήθως κάποιο εξειδικευμένο πρωτόκολλο επικοινωνίας το οποίο υλοποιείται πάντοτε σε hardware. Τέτοιοι μικροελεγκτές χρησιμοποιούνται σε τηλεπικοινωνιακές συσκευές όπως τα μόντεμ.

Η μεγάλη μερίδα πωλήσεων των μικροελεγκτών εξακολουθεί να αφορά αυτούς των 8-bit, καθώς είναι η κατηγορία με το χαμηλότερο κόστος και το μικρότερο μέγεθος λογισμικού για το ίδιο αποτέλεσμα, ιδίως επειδή οι

σύγχρονες οικογένειες μικροελεγκτών 8-bit έχουν πολύ βελτιωμένες επιδόσεις σε σχέση με το παρελθόν.

1.1.8 Εργαλεία ανάπτυξης και γλώσσες προγραμματισμού

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

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

1.1.9 Κατασκευαστές μικροελεγκτών

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

- ARM (δεν κατασκευάζει αλλά παραχωρεί δικαιώματα χρήσης του πυρήνα)
- Atmel

- Epson
- Freescale Semiconductor (πρώην Motorola)
- Hitachi
- Maxim (μετά την εξαγορά της Dallas)
- Microchip
- NEC
- Toshiba
- Texas Instruments
- Intel
- Analog Devices

1.2 Arduino

1.2.1 Ιστορική αναδρομή

Το 2005 ξεκίνησε ένα σχέδιο ώστε να κατασκευαστεί μια συσκευή για τον έλεγχο προγραμμάτων διαδραστικών σχεδίων από μαθητές, η οποία θα ήταν πιο οικονομική από άλλα πρωτότυπα διαθέσιμα συστήματα εκείνη την περίοδο. Οι ιδρυτές αυτού του σχεδίου είναι οι Massimo Banzì και David Cueartielles και ονόμασαν το σχέδια εμπνευσμένοι από Arduino της Inrea. Ξεκίνησαν να παράγουν πλακέτες σε ένα μικρό εργοστάσιο στην Ιβρέα. Η Ιβρέα είναι κωμόπολη της επαρχίας του Τορίνο, στην περιοχή Πεδεμόντιο της βορειοδυτικής Ιταλίας, όπου στην ίδια περιοχή στεγαζόταν η εταιρία υπολογιστών Olivetti. Το σχέδιο Arduino είναι μια διακλάδωση της πλατφόρμας Wiring για λογισμικό ανοιχτού κώδικα και προγραμματίζεται χρησιμοποιώντας μια γλώσσα βασισμένη στο Wiring (σύνταξη και βιβλιοθήκες), παρόμοια με τη C++ με απλοποιήσεις και αλλαγές, καθώς και ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE). Το πρόγραμμα Arduino έλαβε τιμητική μνεία στην κατηγορία Digital Communities στο Prix Ars Electronica το 2006.

1.2.2 Τι είναι το Arduino

Το Arduino είναι μια υπολογιστική πλατφόρμα βασισμένη σε μια απλή μητρική πλακέτα με ενσωματωμένο μικροελεγκτή και διαθέσιμες εισόδους και εξόδους, και η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring (ουσιαστικά πρόκειται για τη γλώσσα προγραμματισμού C++ και ένα σύνολο από βιβλιοθήκες, υλοποιημένες επίσης στην C++). Το Arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε Processing, Max/MSP, Pure Data, SuperCollider. Οι περισσότερες εκδόσεις του Arduino μπορούν να αγοραστούν προ-συναρμολογημένες• το διάγραμμα και πληροφορίες για το υλικό είναι ελεύθερα διαθέσιμα για αυτούς που θέλουν να συναρμολογήσουν το Arduino μόνοι τους.

1.2.3 Υλικό – Hardware

Μία πλακέτα Arduino αποτελείται από ένα μικροελεγκτή Atmel AVR (ATmega328 και ATmega168 στις νεότερες εκδόσεις, ATmega8 στις παλαιότερες) και συμπληρωματικά εξαρτήματα για την διευκόλυνση του χρήστη στον προγραμματισμό και την ενσωμάτωση του σε άλλα κυκλώματα. Όλες οι πλακέτες περιλαμβάνουν ένα γραμμικό ρυθμιστή τάσης 5V και έναν κρυσταλλικό ταλαντωτή 16MHz (ή κεραμικό αντηχητή σε κάποιες παραλλαγές). Ο μικροελεγκτής είναι από κατασκευής προγραμματισμένος με ένα bootloader, έτσι ώστε να μην χρειάζεται εξωτερικός προγραμματιστής.

Σε εννοιολογικό επίπεδο, στην χρήση του Arduino software stack, όλα τα boards προγραμματίζονται με μία RS-232 σειριακή σύνδεση, αλλά ο τρόπος που επιτυγχάνεται αυτό διαφέρει σε κάθε hardware εκδοχή. Οι σειριακές πλάκες Arduino περιέχουν ένα απλό level shifter κύκλωμα για να μετατρέπει μεταξύ σήματος επιπέδου RS-232 και TTL. Τα τωρινά Arduino προγραμματίζονται μέσω USB, αυτό καθιστάται δυνατό μέσω της εφαρμογής προσαρμοστικών chip USB-to-Serial όπως το FTDI FT232. Κάποιες παραλλαγές, όπως το Arduino mini και το ανεπίσημο Boarduino, χρησιμοποιούν ένα αφαιρούμενο USB-to-Serial καλώδιο ή board, Bluetooth ή άλλες μεθόδους. Η πλακέτα Arduino διαθέτει τα περισσότερα microcontroller

I/O pins για χρήση από άλλα κυκλώματα. Τα Diecimila, Duemilanove και το τρέχον Uno παρέχουν 14 ψηφιακά I/O pins, έξι από τα οποία μπορούν να παράγουν pulse-width διαμορφωμένα σήματα (PWM), και έξι αναλογικά δεδομένα.

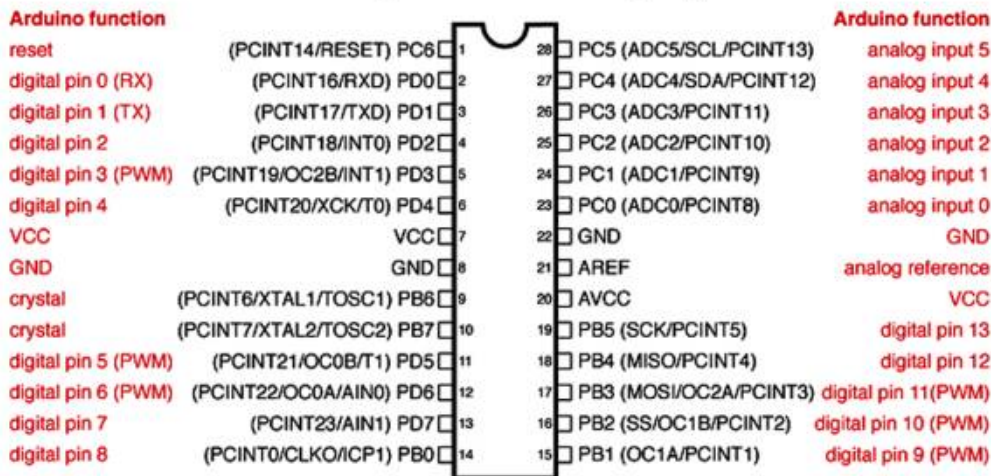
1.2.4 ATmega328 – Η καρδιά του Arduino

Το Arduino βασίζεται στον ATmega328, ο οποίος είναι ένας 8-bit RISC μικροελεγκτής και χρονίζει στα 16MHz. Διαθέτει μνήμη SRAM μεγέθους 2KB η οποία είναι η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν τα προγράμματα για να αποθηκεύουν μεταβλητές, πίνακες κ.λπ. κατά το runtime. Όπως και σε έναν υπολογιστή, αυτή η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο Arduino σταματήσει ή αν γίνει reset.

Επίσης διαθέτει μνήμη EEPROM μεγέθους 1KB, η οποία μπορεί να χρησιμοποιηθεί για «ωμή» εγγραφή/ανάγνωση δεδομένων (χωρίς datatype) ανά byte από τα προγράμματα κατά το runtime. Σε αντίθεση με την SRAM, η EEPROM δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset οπότε είναι το ανάλογο του σκληρού δίσκου.

Τέλος ο μικροελεγκτής ATmega328 διαθέτει μνήμη Flash μεγέθους 32KB, από τα οποία 0.5KB χρησιμοποιείται από το firmware του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware αυτό που στην ορολογία του Arduino ονομάζεται bootloader είναι αναγκαίο για την εγκατάσταση των προγραμμάτων στον μικροελεγκτή μέσω της θύρας USB, χωρίς δηλαδή να χρειάζεται εξωτερικός hardware programmer. Τα υπόλοιπα 31.5KB της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash, όπως και η EEPROM δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset. Επίσης, ενώ η μνήμη Flash υπό κανονικές συνθήκες δεν προορίζεται για χρήση runtime μέσα από τα προγράμματά σας, λόγω της μικρής συνολικής μνήμης που είναι διαθέσιμη σε αυτά (2KB SRAM + 1KB EEPROM), έχει σχεδιαστεί μια βιβλιοθήκη που επιτρέπει την χρήση όσου χώρου περισσεύει.

ATmega168/328 Pin Mapping



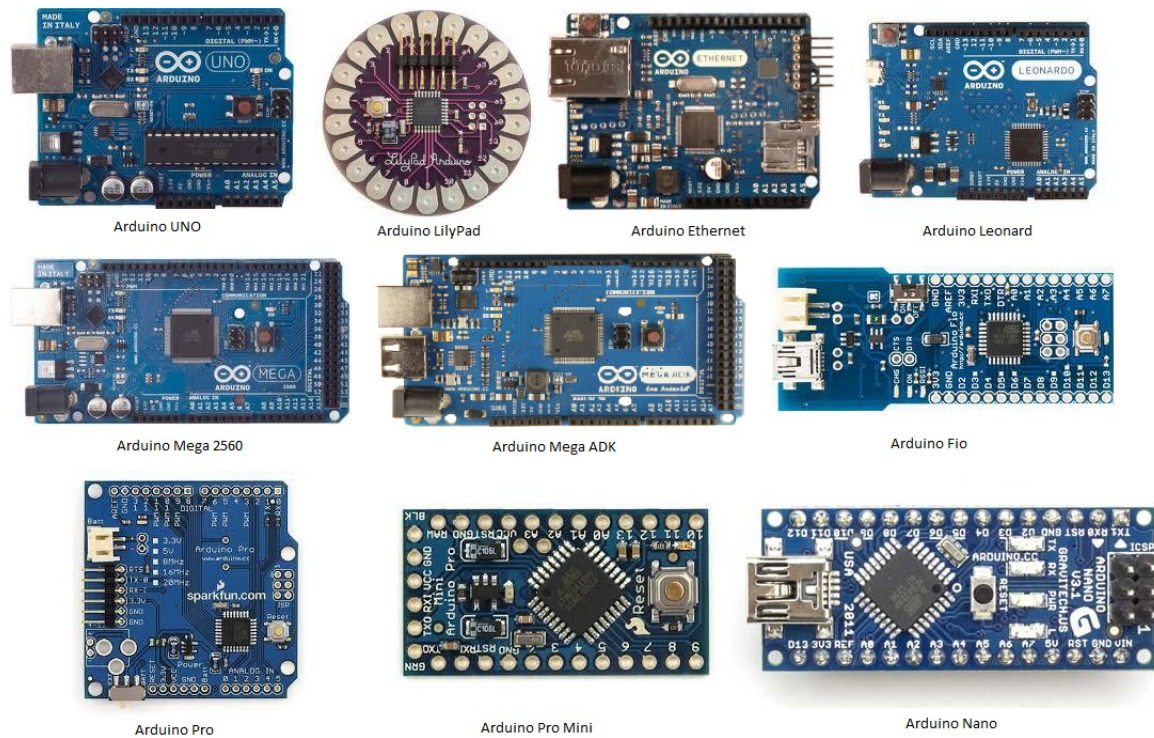
Digital Pins 11, 12 & 13 are used by the ICSP header for MISO, MOSI, SCK connections (Atmega168 pins 17, 18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Mapping ATmega168/328 to Arduino pins

Εικόνα 1.2: Οι ακροδέκτες των μικροελεγκτών ATmega168/328

1.2.5 Μοντέλα Arduino

Υπάρχουν διάφορα μοντέλα Arduino στην αγορά. Αυτό έχει ως αποτέλεσμα ο χρήστης να μπορεί να επιλέξει ανάλογα με τις δυνατότητες που του παρέχει. Για την υλοποίηση της εργασίας χρησιμοποιήθηκε το Arduino UNO διότι καλύπτει όλες τις ανάγκες της εργασίας καθώς τα οι ακροδέκτες που διαθέτει επαρκούν για όλες τις λειτουργίες και η μνήμη του μικροελεγκτή αλλά και η υπολογιστική δύναμή του αρκούνε ώστε να αποθηκευθεί και να εκτελεστεί με επιτυχία το πρόγραμμα που υλοποιήθηκε.



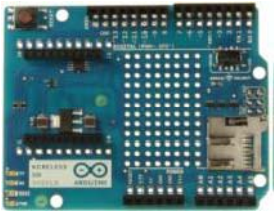
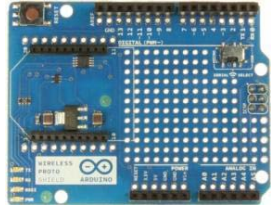
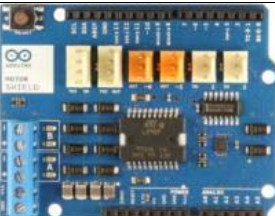


Εικόνα 1.3: Μοντέλα μικροελεγκτών Arduino

1.2.6 Arduino Shields

Τα shields είναι πλακέτες που μπορούν να συνδεθούν στην κορυφή ενός μικροελεγκτή Arduino, επεκτείνοντας τις δυνατότητές του. Επίσης διαθέτουν header pins ώστε να υπάρχει η δυνατότητα πάνω σε ένα shield να συνδέεται και πρόσθετο shield κ.ο.κ. Τα διαφορετικά shields, ακολουθούν την ίδια φιλοσοφία, δηλαδή την ευκολία στην τοποθέτηση και το μικρό κόστος στην παραγωγή τους. Μερικά από αυτά φαίνονται στον ακόλουθο πίνακα.

Πίνακας 1.2: *Arduino Shields*

<p>Arduino WiFi Shield</p>		<p>Το Arduino WiFi Shield συνδέει το Arduino στο διαδίκτυο ασύρματα.</p>
<p>Arduino Ethernet Shield</p>		<p>Το Arduino Ethernet Shield συνδέει το Arduino στο διαδίκτυο με ένα RJ45 καλώδιο.</p>
<p>Wireless SD Shield</p>		<p>Το Wireless SD Shield επιτρέπει σε μια πλακέτα Arduino να επικοινωνεί ασύρματα με μια ασύρματη μονάδα. Η μονάδα μπορεί να επικοινωνήσει έως και 100 πόδια σε εσωτερικούς χώρους ή σε εξωτερικούς χώρους ως 300 πόδια. Η μονάδα περιλαμβάνει μια θύρα υποδοχής SD.</p>
<p>Wireless Proto Shield</p>		<p>Το Wireless Proto Shield επιτρέπει στο Arduino να επικοινωνεί ασύρματα με μια ασύρματη μονάδα. Η μονάδα μπορεί να επικοινωνήσει έως και 100 πόδια σε εσωτερικούς χώρους ή σε εξωτερικούς χώρους ως 300 πόδια. Η μονάδα δεν περιλαμβάνει θύρα υποδοχής SD.</p>
<p>Arduino Motor Shield</p>		<p>Το Arduino Motor Shield επιτρέπει την οδήγηση δύο DC κινητήρων, ελέγχοντας την ταχύτητα και την κατεύθυνση του καθενός ξεχωριστά.</p>

1.2.7 Τεχνικά χαρακτηριστικά του Arduino UNO

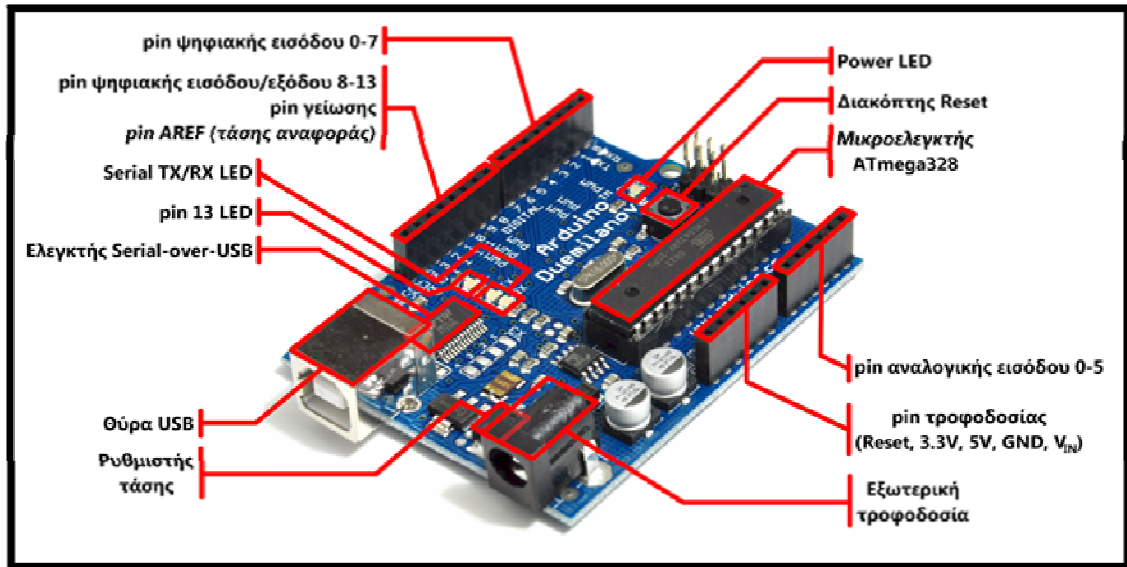
Τα τεχνικά χαρακτηριστικά του Arduino UNO αναφέρονται στον πίνακα που ακολουθεί.

Πινάκας 1.3: Τεχνικά χαρακτηριστικά του Arduino UNO

Μικροελεγκτής	ATmega328
Τάση λειτουργίας	5V
Προτεινόμενη Τάση εισόδου	7-12V
Όρια τάσης εισόδου	6-20V
Ψηφιακοί ακροδέκτες I/O	14, (6 PWM έξοδοι)
Αναλογικοί ακροδέκτες εισόδου	6
Συνεχές ρεύμα ανά ακροδέκτη	40mA
Συνεχές ρεύμα για τον ακροδέκτη τάσης 3.3V	50mA
Μνήμη flash	32KB (ATMEGA328)
Μνήμη SRAM	2KB (ATMEGA328)
Μνήμη EEPROM	1KB (ATMEGA328)
Ταχύτητα ρολογιού	16MHz
Μήκος	68,6mm
Πλάτος	53,4mm
Βάρος	25g

1.2.8 Είσοδοι – Έξοδοι

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



Εικόνα 1.4: Περιγραφή των ακροδεκτών και των εξαρτημάτων του Arduino.

Επιπλέον, στην πάνω πλευρά του Arduino βρίσκονται 14 θηλυκά pin, αριθμημένα από 0 ως 13, που μπορούν να λειτουργήσουν ως ψηφιακές εισοδοί και έξοδοι. Ο ακροδέκτης που προγραμματίζεται σαν INPUT με την εντολή `pinMode()` τίθεται σε κατάσταση υψηλής αντίστασης. Είναι σαν να υπάρχει μια αντίσταση 100 Megohms μπροστά από τον ακροδέκτη. Αυτός ο τρόπος είναι χρήσιμος για την ανάγνωση αισθητήρων αλλά δεν χρησιμοποιείται για την οδήγηση LED. Ο ακροδέκτης που προγραμματίζεται σαν OUTPUT με την εντολή `pinMode()` παρουσιάζει χαμηλή αντίσταση. Αυτό σημαίνει ότι μπορεί να διαθέσει αρκετό ρεύμα για να οδηγήσει άλλα κυκλώματα. Οι ακροδέκτες του Atmega μπορούν να δώσουν (source) ή να πάρουν (sink) ρεύμα μέχρι 40 mA. Επομένως, μπορούν να οδηγήσουν LED αλλά δεν χρησιμοποιούνται για την ανάγνωση αισθητήρων. Οι ακροδέκτες που προγραμματίζονται σαν έξοδοι μπορούν να καταστραφούν εάν συνδεθούν στη γείωση ή την τάση. Το παραπάνω ρεύμα δεν είναι αρκετό για την οδήγηση ρελαί ή μοτέρ και γι' αυτόν το λόγο χρειάζονται ενδιάμεσα κυκλώματα σε αυτές τις περιπτώσεις.

Πίνακας 1.4: Σύντομη περιγραφή των λειτουργιών των ακροδεκτών του Arduino.

MODE	DIGITAL PINS BEHAVIOR
Input – pinMode(pin,INPUT)	Το digitalWrite(pin) επιστρέφει HIGH εάν η τάση εισόδου είναι $\geq 3V$ και LOW εάν είναι $\leq 2V$
	Εάν ένας ακροδέκτης εισόδου γίνει HIGH με το digitalWrite(pin,HIGH), η εσωτερική pull-up αντίσταση των 20 KΩ ενεργοποιείται. Τότε, διαβάζοντας αυτόν τον ακροδέκτη με digitalWrite(pin) θα επιστρέψει HIGH εκτός εάν γίνεται LOW από το εξωτερικό κύκλωμα. Αυτό χρησιμοποιείται για να έχουμε κάποιον ακροδέκτη σε γνωστή κατάσταση και όχι στον αέρα.
Output – pinMode(pin,OUTPUT)	Το digitalWrite(pin,HIGH) θα δώσει έξοδο 5V και θα δώσει ρεύμα 40 mA (source).
	Το digitalWrite(pin,LOW) θα δώσει έξοδο 0V και θα τραβήξει ρεύμα 40 mA (sink).

Ως ψηφιακή έξοδος, ένα από αυτά τα pin μπορεί να τεθεί από το πρόγραμμα σε κατάσταση HIGH ή LOW, οπότε το Arduino θα ξέρει αν πρέπει να διοχετεύσει ή όχι ρεύμα στο συγκεκριμένο pin. Με αυτόν τον τρόπο υπάρχει η δυνατότητα για παράδειγμα να ανάψει και να σβήσει ένα LED που έχει συνδεθεί στο συγκεκριμένο pin. Αν πάλι ρυθμιστεί ένα από αυτά τα pin ως ψηφιακή είσοδος μέσα από το πρόγραμμά, μπορεί με την εντολή digitalWrite(pin) να διαβαστεί η κατάστασή του (HIGH ή LOW) ανάλογα με το αν η εξωτερική συσκευή που έχει συνδεθεί σε αυτό το pin διοχετεύει ή όχι ρεύμα στο pin (με αυτόν τον τρόπο λόγω χάρη μπορεί να διαβαστεί η κατάσταση ενός διακόπτη).

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

- Τα pin 0 και 1 λειτουργούν ως RX και TX της σειριακής όταν το πρόγραμμά σας ενεργοποιεί την σειριακή θύρα. Έτσι, όταν για παράδειγμα το πρόγραμμα στέλνει δεδομένα στην σειριακή, αυτά προωθούνται και στην θύρα USB μέσω του ελεγκτή Serial-Over-USB αλλά και στο pin 0 για να τα διαβάσει ενδεχομένως μια άλλη συσκευή (π.χ. ένα δεύτερο Arduino στο δικό του pin 1). Αυτό φυσικά σημαίνει ότι

αν στο πρόγραμμα ενεργοποιηθεί το σειριακό interface, χάνονται 2 ψηφιακές εισοδοί/έξοδοι.

- Τα pin 2 και 3 λειτουργούν και ως εξωτερικά interrupt (interrupt 0 και 1 αντίστοιχα). Με άλλα λόγια, μπορεί να ρυθμιστεί μέσα από το πρόγραμμα ώστε να λειτουργούν αποκλειστικά ως ψηφιακές εισοδοί στις οποίες όταν συμβαίνουν συγκεκριμένες αλλαγές, η κανονική ροή του προγράμματος σταματάει άμεσα και εκτελείται μια συγκεκριμένη συνάρτηση. Τα εξωτερικά interrupt είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας.
- Τα pin 3, 5, 6, 9, 10 και 11 μπορούν να λειτουργήσουν και ως ψευδοαναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation), δηλαδή το ίδιο σύστημα που διαθέτουν οι μητρικές των υπολογιστών για να ελέγχουν τις ταχύτητες των ανεμιστήρων. Έτσι, μπορεί να συνδεθεί για παράδειγμα ένα LED σε κάποιο από αυτά τα pin και να ελεγχθεί πλήρως η φωτεινότητά του με ανάλυση 8bit (256 καταστάσεις από 0 όπου θα είναι σβηστό ως 255 όπου θα είναι πλήρως αναμμένο) σε αντίθεση με τις υπόλοιπες ψηφιακές εξόδους που παρέχουν μόνο 2 θέσεις και αυτές είναι σβηστό και πλήρως αναμμένο. Είναι σημαντικό να κατανοηθεί ότι το PWM δεν είναι πραγματικά αναλογικό σύστημα και ότι θέτοντας στην έξοδο την τιμή 127, δεν σημαίνει ότι η έξοδος θα δίνει 2.5V αντί της κανονικής τιμής των 5V, αλλά ότι θα δίνει ένα παλμό που θα εναλλάσσεται με μεγάλη συχνότητα και για ίσους χρόνους μεταξύ των τιμών 0 και 5V.



Εικόνα 1.5: Οι ψηφιακοί ακροδέκτες του Arduino UNO.

Στην κάτω πλευρά του Arduino, με τη σήμανση ANALOG IN, υπάρχει μια ακόμη σειρά από 6 pin, αριθμημένα από το 0 ως το 5. Το καθένα από αυτά λειτουργεί ως αναλογική είσοδος κάνοντας χρήση του ADC (Analog to Digital Converter) που είναι ενσωματωμένο στον μικροελεγκτή. Για παράδειγμα, μπορεί να τροφοδοτηθεί ένα από αυτά με μια τάση η οποία μπορεί να κυμανθεί με ένα ποτενσιόμετρο από 0V ως μια τάση αναφοράς Vref η οποία, αν δεν πραγματοποιηθεί κάποια αλλαγή είναι προρυθμισμένη στα 5V. Τότε, μέσα από το πρόγραμμα διαβάζεται η τιμή του pin ως ένας ακέραιος αριθμός ανάλυσης 10-bit, από 0 (όταν η τάση στο pin είναι 0V) μέχρι 1023 (όταν η τάση στο pin είναι 5V). Η τάση αναφοράς μπορεί να ρυθμιστεί με μια εντολή στο 1.1V, ή σε οποιαδήποτε άλλη τάση μεταξύ 2 και 5V τροφοδοτώντας εξωτερικά με αυτή την τάση το pin με την σήμανση AREF που βρίσκεται στην απέναντι πλευρά της πλακέτας. Έτσι, αν τροφοδοτηθεί το pin AREF με 3.3V και στην συνέχεια διαβαστεί κάποιο pin αναλογικής εισόδου στο οποίο εφαρμοστεί τάση 1.65V, το Arduino θα επιστρέψει την τιμή 512. Τέλος, καθένα από τα 6 αυτά pin, με κατάλληλη εντολή μέσα από το πρόγραμμα μπορεί να μετατραπεί σε ψηφιακό pin εισόδου/εξόδου όπως τα 14 που βρίσκονται στην απέναντι πλευρά και τα οποία περιγράφηκαν πριν. Σε αυτή την περίπτωση τα pin μετονομάζονται από 0~5 σε 14~19 αντίστοιχα.

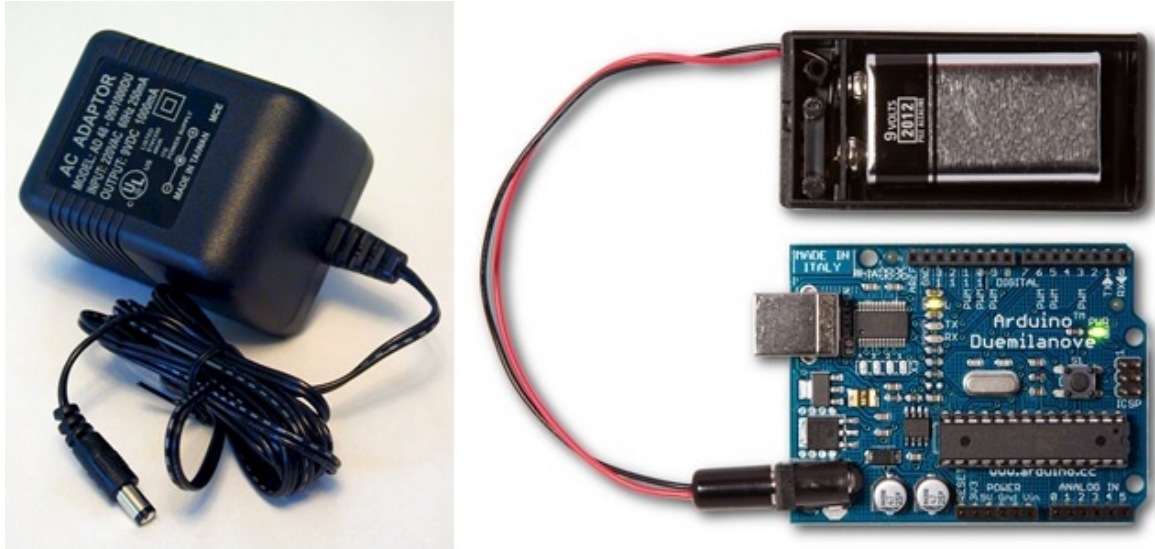


Εικόνα 1.6: Οι αναλογικοί ακροδέκτες του Arduino UNO.

1.2.9 Τροφοδοσία

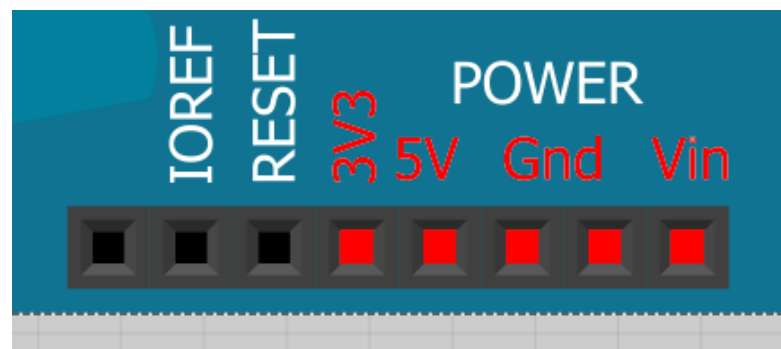
Το Arduino μπορεί να τροφοδοτηθεί με ρεύμα είτε από τον υπολογιστή μέσω της σύνδεσης USB, είτε από εξωτερική τροφοδοσία που παρέχεται μέσω μιας υποδοχής φισ των 2.1mm (θετικός πόλος στο κέντρο) και βρίσκεται στην κάτω-αριστερή γωνία του Arduino. Για να μην υπάρχουν προβλήματα, η

εξωτερική τροφοδοσία πρέπει να είναι από 7 ως 12V και μπορεί να προέρχεται από ένα κοινό μετασχηματιστή του εμπορίου, μπαταρίες ή οποιαδήποτε άλλη πηγή DC.



Εικόνα 1.7: Εξωτερικές πηγές τροφοδοσίας για το Arduino.

Αριστερά από τα pin αναλογικής εισόδου, υπάρχει μια ακόμα συστοιχία από 6 pin με την σήμανση POWER. Η παρακάτω εικόνα παρουσιάζει τις εισόδους και εξόδους τροφοδοσίας του Arduino.



Εικόνα 1.8: Είσοδοι-Εξοδοι τροφοδοσίας του Arduino.

Η λειτουργία του καθενός έχει ως εξής:

- Το πρώτο, με την ένδειξη RESET, όταν γειωθεί (σε οποιοδήποτε από τα 3 pin με την ένδειξη GND που υπάρχουν στο Arduino) έχει ως αποτέλεσμα την επανεκκίνηση του Arduino.

- Το δεύτερο, με την ένδειξη 3.3V, μπορεί να τροφοδοτήσει τα εξαρτήματα με τάση 3.3V. Η τάση αυτή δεν προέρχεται από την εξωτερική τροφοδοσία αλλά παράγεται από τον ελεγκτή Serial-over-USB και έτσι η μέγιστη ένταση που μπορεί να παρέχει είναι μόλις 50mA.
- Το τρίτο, με την ένδειξη 5V, μπορεί να τροφοδοτήσει τα εξαρτήματα με τάση 5V. Ανάλογα με τον τρόπο τροφοδοσίας του ίδιου του Arduino, η τάση αυτή προέρχεται είτε άμεσα από την θύρα USB (που ούτως ή άλλως λειτουργεί στα 5V), είτε από την εξωτερική τροφοδοσία αφού αυτή περάσει από το ρυθμιστή τάσης για να την ψαλιδίσει στα 5V.
- Το τέταρτο και το πέμπτο pin, με την ένδειξη GND, είναι φυσικά γειώσεις.
- Το έκτο και τελευταίο pin, με την ένδειξη Vin έχει διπλό ρόλο. Σε συνδυασμό με το pin γείωσης δίπλα του, μπορεί να λειτουργήσει ως μέθοδος εξωτερικής τροφοδοσίας του Arduino, στην περίπτωση που δεν ενδείκνυται να χρησιμοποιηθεί η υποδοχή του φισ των 2.1mm. Αν όμως έχει ήδη συνδεθεί εξωτερική τροφοδοσία μέσω του φισ, υπάρχει η δυνατότητα να χρησιμοποιηθεί αυτό το pin για να τροφοδοτηθούν εξαρτήματα με την πλήρη τάση της εξωτερικής τροφοδοσίας (7~12V), πριν αυτή περάσει από τον ρυθμιστή τάσης όπως γίνεται με το pin των 5V.

1.2.10 Ενσωματωμένα κουμπιά και LED

Πάνω στην πλακέτα του Arduino υπάρχει ένας διακόπτης micro-switch και 4 μικροσκοπικά LED επιφανειακής στήριξης.

Η λειτουργία του διακόπτη (που έχει την σήμανση RESET) και του ενός LED με την σήμανση POWER είναι μάλλον προφανής.

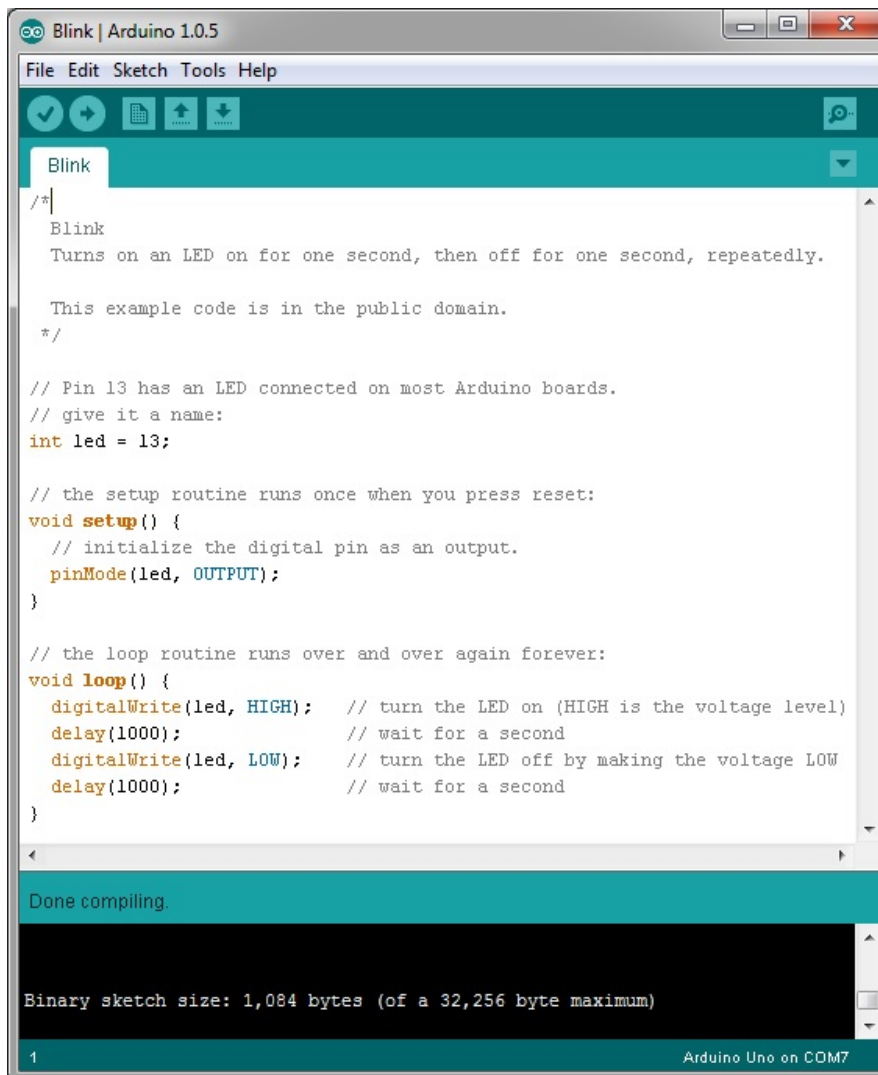
Τα δύο LED με τις σημάνσεις TX και RX, χρησιμοποιούνται ως ένδειξη λειτουργίας του σειριακού interface, καθώς ανάβουν όταν το Arduino στέλνει ή λαμβάνει αντίστοιχα δεδομένα μέσω USB. Να σημειωθεί ότι τα LED αυτά

ελέγχονται από τον ελεγκτή Serial-over-USB και συνεπώς δεν λειτουργούν όταν η σειριακή επικοινωνία γίνεται αποκλειστικά μέσω των ψηφιακών pin 0 και 1.








Τέλος, υπάρχει το LED με την σήμανση L. Η βασική δοκιμή λειτουργίας του Arduino είναι να του ανατεθεί να αναβοσβήνει ένα LED. Για να μπορεί να πραγματοποιηθεί αυτό από την πρώτη στιγμή, χωρίς να συνδεθεί τίποτα πάνω στο Arduino, οι κατασκευαστές του σκέφτηκαν να ενσωματώσουν ένα LED στην πλακέτα, το οποίο σύνδεσαν στο ψηφιακό pin 13. Έτσι, ακόμα και αν δεν έχει συνδεθεί τίποτα πάνω στο φυσικό pin 13, αναθέτοντάς του την τιμή HIGH μέσα από το πρόγραμμα, θα ανάψει αυτό το ενσωματωμένο LED.

1.2.11 Λογισμικό - Arduino IDE και σύνδεση με τον υπολογιστή

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) του Arduino είναι μία εφαρμογή γραμμένη σε Java, που λειτουργεί σε πολλές πλατφόρμες, και προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing και το σχέδιο Wiring. Έχει σχεδιαστεί για να εισαγάγει τον προγραμματισμό στους καλλιτέχνες και τους νέους που δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως είναι η επισήμανση σύνταξης και ο συνδυασμός αγκύλων και είναι επίσης σε θέση να μεταγλωττίζει και να φορτώνει προγράμματα στην πλακέτα με ένα μόνο κλικ. Δεν υπάρχει συνήθως καμία ανάγκη να επεξεργαστούν αρχεία make ή να τρέξουν προγράμματα σε ένα περιβάλλον γραμμής εντολών. Ένα πρόγραμμα ή κώδικας που γράφεται για Arduino ονομάζεται sketch.



Εικόνα 1.9: Το περιβάλλον ανάπτυξης του Arduino IDE.

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

Εικόνα 1.10: Οι λειτουργίες των εικονιδίων του Arduino IDE.

Ό, τι χρειάζεται για τη διαχείριση του Arduino από τον υπολογιστή το παρέχει το Arduino IDE, όπου η πιο πρόσφατη έκδοση του είναι διαθέσιμη για λήψη στο επίσημο site του Arduino για καθένα από τα τρία δημοφιλέστερα λειτουργικά συστήματα.

Το Arduino IDE είναι βασισμένο σε Java και συγκεκριμένα παρέχει:

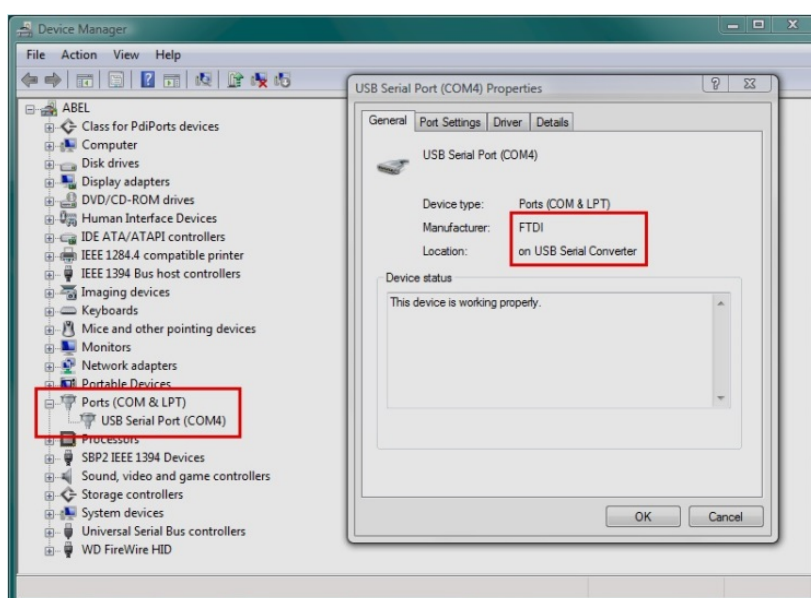
- Ένα πρακτικό περιβάλλον για την συγγραφή των προγραμμάτων (τα οποία ονομάζονται sketch στην ορολογία του Arduino) με συντακτική χρωματική σήμανση.
- Αρκετά έτοιμα παραδείγματα τα οποία βοηθούν στην επίλυση κάποιου προβλήματος.
- Μερικές έτοιμες βιβλιοθήκες για προέκταση της γλώσσας και για πιο εύκολο χειρισμό -μέσα από τον κώδικα- των εξαρτημάτων που συνδέονται στο Arduino.
- Τον compiler για την μεταγλώττιση των sketch.
- Ένα serial monitor που παρακολουθεί τις επικοινωνίες της σειριακής επικοινωνίας (USB), αναλαμβάνει να στείλει αλφαριθμητικούς χαρακτήρες της επιλογής του προγραμματιστή στο Arduino μέσω αυτής και είναι ιδιαίτερα χρήσιμο για το debugging των sketch.
- Τη δυνατότητα να φορτωθεί το μεταγλωττισμένο sketch στο Arduino.

Για τα δύο τελευταία χαρακτηριστικά βέβαια, το Arduino πρέπει να έχει συνδεθεί σε μια από τις θύρες USB του υπολογιστή και, λόγω του ελεγκτή Serial-over-USB, θα πρέπει να αναγνωριστεί από το λειτουργικό σύστημα ως εικονική σειριακή θύρα.

Για την σύνδεση θα χρειαστεί ένα καλώδιο USB από Type A σε Type B, όπως αυτό των εκτυπωτών. Για την αναγνώριση από το λειτουργικό είναι απαραίτητη η εγκατάσταση του driver του FTDI chip (δηλαδή του ελεγκτή Serial-over-USB) ο οποίος υπάρχει στον φάκελο drivers του Arduino IDE που έγινε λήψη από το επίσημο site. Για την τελευταία έκδοση αυτού του οδηγού

μπορεί επίσης να κατεβεί για κάθε λειτουργικό σύστημα από το site της FTDI. Να σημειωθεί ότι στους τελευταίους πυρήνες του Linux υπάρχει εγγενής υποστήριξη του συγκεκριμένου ελεγκτή.

Αν όλα έγιναν σωστά, το κεντρικό παράθυρο του Arduino IDE θα εμφανιστεί όταν εκτελεστεί και στο μενού Tools → Serial Port θα πρέπει να εμφανίζεται η εικονική σειριακή θύρα (συνήθως COM# για τα Windows, /dev/ttyusbserial## για το MacOS και /dev/ttyusb## για το Linux). Θα πρέπει να είναι επιλεγμένη αυτή η εικονική θύρα και στην συνέχεια ο τύπος του Arduino (Arduino Uno) από το μενού Tools → Board. Το Arduino είναι πλέον έτοιμο να δεχτεί τα sketch του προγραμματιστή.



Εικόνα 1.11: Εγκατάσταση των drivers της σειριακής θύρας του Arduino.

1.2.12 Γλώσσα προγραμματισμού

Η γλώσσα του Arduino βασίζεται στη γλώσσα Wiring, μια παραλλαγή C/C++ για μικροελεγκτές αρχιτεκτονικής AVR όπως ο ATmega, και υποστηρίζει όλες τις βασικές δομές της C καθώς και μερικά χαρακτηριστικά της C++. Για compiler χρησιμοποιείται ο AVR gcc και ως βασική βιβλιοθήκη C χρησιμοποιείται η AVR libc.

Λόγω της καταγωγής της από την γλώσσα C, στη γλώσσα του Arduino υπάρχει η δυνατότητα να χρησιμοποιηθούν ουσιαστικά οι ίδιες βασικές εντολές και συναρτήσεις, με την ίδια σύνταξη, τους ίδιους τύπους δεδομένων και τους ίδιους τελεστές όπως και στη γλώσσα C. Πέρα από αυτές όμως, υπάρχουν κάποιες ειδικές εντολές, συναρτήσεις και σταθερές που βοηθούν για την διαχείριση του ειδικού hardware του Arduino.

Πίνακας 1.5: Βασικές εντολές και συναρτήσεις της γλώσσας του Arduino. [9]

Όρισμα	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	-	(<i>pin, mode</i>)	Καθορίζει αν το συγκεκριμένο ψηφιακό <i>pin</i> θα είναι <i>pin</i> εισόδου ή <i>pin</i> εξόδου ανάλογα με την τιμή που δίνεται στην παράμετρο <i>mode</i> (INPUT ή OUTPUT αντίστοιχα).
digitalWrite	Εντολή	-	(<i>pin, pinstatus</i>)	Θέτει την κατάσταση <i>-pinstatus-</i> (HIGH ή LOW) στο συγκεκριμένο ψηφιακό <i>pin</i> .
digitalRead	Συνάρτηση	int	(<i>pin</i>)	Επιστρέφει την κατάσταση του συγκεκριμένου ψηφιακού <i>-pin-</i> (0 για LOW και 1 για HIGH) εφόσον αυτό είναι <i>pin</i> εισόδου.
analogReference	Εντολή	-	(<i>type</i>)	Δέχεται τις τιμές DEFAULT, INTERNAL ή EXTERNAL

				στην παράμετρο <i>-type-</i> 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	(<i>pin</i>)	Επιστρέφει έναν ακέραιο από 0 έως 1023, ανάλογα με την τάση που τροφοδοτείται το συγκεκριμένο <i>-pin-</i> αναλογικής εισόδου στην κλίμακα 0 ως V_{ref} .
analogWrite	Εντολή	-	(<i>pin, value</i>)	Θέτει το συγκεκριμένο ψηφιακό <i>-pin-</i> σε κατάσταση ψευδοαναλογικής εξόδου (PWM). Η παράμετρος <i>-value-</i> καθορίζει το πλάτος του παλμού σε σχέση με την περίοδο του παραγόμενου σήματος στην κλίμακα από 0 ως 255 (π.χ. με <i>-value-</i> 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. 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	-	(<i>time</i>)	Σταματά προσωρινά την ροή του προγράμματος για <i>time</i> ms. Η παράμετρος <i>time</i> είναι unsigned long (από 0 ως 2^{32}). Να σημειωθεί ότι παρά την προσωρινή παύση, συναρτήσεις των οποίων η εκτέλεση ενεργοποιείται από interrupt θα εκτελεστούν

				κανονικά κατά την διάρκεια μιας delay.
attachInterrupt	Εντολή	-	(<i>interrupt,function,triggermode</i>)	<p>Θέτει σε λειτουργία το συγκεκριμένο <i>interrupt</i>, ώστε να ενεργοποιεί την συνάρτηση <i>function</i>, κάθε φορά που ικανοποιείται η συνθήκη που ορίζεται από την παράμετρο <i>triggermode</i>:</p> <ul style="list-style-type: none"> • LOW (ενεργοποίηση όταν η κατάσταση του pin που αντιστοιχεί στο συγκεκριμένο <i>interrupt</i> γίνει LOW) • RISING (όταν από LOW γίνει HIGH) • FALLING (όταν από HIGH γίνει LOW) • CHANGE (όταν αλλάξει κατάσταση γενικά)
detachInterrupt	Εντολή	-	(<i>interrupt</i>)	Απενεργοποιεί το συγκεκριμένο <i>interrupt</i> .
noInterrupts	Εντολή	-	()	Σταματά προσωρινά την λειτουργία όλων των <i>interrupt</i>
interrupts	Εντολή	-	()	Επαναφέρει την λειτουργία των <i>interrupt</i> που διακόπηκε προσωρινά από μια εντολή <i>noInterrupts</i> .
Serial.begin	Μέθοδος κλάσης	-	(<i>datarate</i>)	Θέτει τον ρυθμό μεταφοράς δεδομένων του σειριακού interface (σε baud)
Serial.println	Μέθοδος κλάσης	-	(<i>data</i>)	Μεταφορά δεδομένων (<i>data</i>) ώστε να αποσταλούν μέσω interface. Τα δεδομένα, είναι αριθμοί ή αλφάριθμοι.

Επιπλέον, στη γλώσσα του Arduino κάθε πρόγραμμα αποτελείται από δύο βασικές ρουτίνες ώστε να έχει την παρακάτω γενική δομή:

```
// Ενσωματώσεις βιβλιοθηκών, δηλώσεις μεταβλητών...

void setup()
{
  // ...
}

void loop()
{
  // ...
}

// Υπόλοιπες συναρτήσεις...
```

Η ρουτίνα setup καλείται κατά την εκκίνηση του προγράμματος και χρησιμοποιείται για αρχικοποίηση (προετοιμασία) μεταβλητών, της λειτουργικότητας των χρησιμοποιούμενων ακροδεκτών (pins), για εκκίνηση χρήσης βιβλιοθηκών και διάφορων άλλων λειτουργιών. Εκτελείται μόνο μια φορά από τη στιγμή που τίθεται σε λειτουργία το Arduino ή μετά από επανεκκίνηση.

Η ρουτίνα loop περιέχει το βασικό κορμό του προγράμματος και κάνει αυτό ακριβώς που αναφέρει τα όνομά της, δηλαδή η εκτέλεσή της επαναλαμβάνεται συνέχεια σαν ένας βρόγχος while (true) (δομή επανάληψης για τη γλώσσα προγραμματισμού C), επιτρέποντας στο πρόγραμμα να αλλάξει και να απαντήσει. Χρησιμοποιείται για τον ενεργό έλεγχο του Arduino.

1.3 Το μονοξείδιο του άνθρακα

1.3.1 Γενικά

Είναι ένας από τους κυριότερους ρύπους της ατμόσφαιρας. Στη φύση, το μονοξείδιο του άνθρακα παράγεται κυρίως από την αντίδραση της ρίζας του υδροξυλίου με το μεθάνιο που παράγεται κατά την αποσύνθεση οργανικών ουσιών, από τη βακτηριακή δράση στους ωκεανούς, κατά τις δασικές πυρκαγιές και δευτερευόντως από τα ηφαίστεια, τις ηλεκτρικές εκκενώσεις που συμβαίνουν κατά τη διάρκεια των καταιγίδων και τη διαφυγή φυσικών αερίων από τη Γη. Υπολογίζεται ότι από την αντίδραση του μεθανίου με τη ρίζα του υδροξυλίου παράγονται περίπου $3 \cdot 10^{12}$ kg/έτος μονοξειδίου του άνθρακα, ενώ από τη βακτηριακή δράση στους ωκεανούς παράγονται περίπου 10^{11} kg/έτος.

Το μονοξείδιο του άνθρακα είναι ανόργανη χημική ένωση, που περιέχει άνθρακα και οξυγόνο, με χημικό τύπο CO. Το καθαρό μονοξείδιο του άνθρακα, στις «συνηθισμένες συνθήκες», δηλαδή σε θερμοκρασία 25°C και υπό πίεση 1 atm, είναι άχρωμο, άοσμο και άγευστο αέριο. Είναι τοξικό για τους ανθρώπους και τα ζώα, ακόμα και σε μικρές συγκεντρώσεις, παρ'όλο ότι παράγεται σε μικρές ποσότητες από τον κανονικό ζωικό μεταβολισμό και θεωρείται ότι συμμετέχει σε κάποιες φυσιολογικές βιολογικές λειτουργίες. Στην ατμόσφαιρα, είναι πολύ ευμετάβλητο και βραχύβιο, έχοντας ένα ρόλο στο σχηματισμό του τροποσφαιρικού όζοντος.

Κύρια πηγή του, στις αστικές περιοχές, είναι τα βενζινοκίνητα αυτοκίνητα, ενώ δευτερεύουσα πηγή είναι η καύση πετρελαίου για κεντρική θέρμανση. Σε παγκόσμια κλίμακα υπολογίζεται ότι από τις ανθρώπινες δραστηριότητες εκλύονται στην ατμόσφαιρα ποσότητες μονοξειδίου του άνθρακα της τάξης των $7 \cdot 10^9$ kg/έτος. Το μονοξείδιο του άνθρακα είναι ο πιο χημικά αδρανής ρύπος στην ατμόσφαιρα και η συγκέντρωσή του διαρκώς ελαττώνεται λόγω της οξειδωσης του σε CO₂, από την ηλιακή ενέργεια, με αποτέλεσμα ο χρόνος παραμονής του στην ατμόσφαιρα να είναι πολύ μικρός.

Το μονοξειδίο του άνθρακα σε αρρύπαντη ατμόσφαιρα βρίσκεται σε μικρές συγκεντρώσεις της τάξης του 0.1 ppm (= 0.1145 mgr/m³), ενώ αντίθετα σε ρυπασμένες περιοχές, όπως σε στενές και χωρίς καλό αερισμό λεωφόρους, οι συγκεντρώσεις του μπορούν να ξεπεράσουν τα 50 ppm (= 57.25 mgr/m³).

Πίνακας 1.6α. Συσχέτιση της συγκέντρωσης του CO στον αναπνεόμενο αέρα (1 ppm = 1145 mgr/m³), σε χώρο παραμονής, με τα εμφανιζόμενα στα άτομα συμπτώματα (Βασιλικιώτης 1989).

CO (ppm)	ΕΜΦΑΝΙΖΟΜΕΝΑ ΣΥΜΠΤΩΜΑΤΑ ΣΕ ΑΤΟΜΑ
<10	Πρώτα συμπτώματα στη συμπεριφορά
10-30	Προσβολή του κεντρικού νευρικού συστήματος, διαταραχές στην όραση και τις κινήσεις
31-60	Καρδιακές και πνευμονικές διαταραχές
61-3000	Πονοκέφαλος, κόπωση, κώμα, αδυναμία αναπνοής, θάνατος

Στον άνθρωπο η επίδραση το μονοξειδίου του άνθρακα γίνεται εμφανής όταν οι συγκεντρώσεις του στην ατμόσφαιρα υπερβούν κάποια όρια. Ειδικότερα, όταν εισπνέεται το μονοξειδίο του άνθρακα, διαχέεται ευκολότατα από τους πνεύμονες στην κυκλοφορία του αίματος και δεσμεύεται στην αιμοσφαιρίνη. Η αιμοσφαιρίνη αποτελεί συστατικό των ερυθρών αιμοσφαιρίων, τα οποία μεταφέρουν το οξυγόνο από τους πνεύμονες στους διάφορους ιστούς. Όταν το μονοξειδίο του άνθρακα δεσμευθεί στην αιμοσφαιρίνη, δεν αποδεσμεύεται. Όταν αυξηθούν τα επίπεδα του μονοξειδίου του άνθρακα στο αίμα, μειώνεται η διαθέσιμη για τη μεταφορά του οξυγόνου αιμοσφαιρίνη. Ο εγκέφαλος και οι άλλοι ιστοί, οι οποίοι χρειάζονται οξυγόνο για τον φυσιολογικό μεταβολισμό τους, αρχίζουν να υπολειτουργούν και τελικά πεθαίνουν.

Γενικά, η έκθεση σε μικρές συγκεντρώσεις μονοξειδίου του άνθρακα, ενώ σε υγιή άτομα δεν προκαλεί τίποτε περισσότερο από έναν απλό πονοκέφαλο, είναι ιδιαίτερα επιβαρυντική στην υγεία όσο πάσχουν από στεφανιαία νοσήματα. Παρόλα αυτά, ακόμη και χαμηλότερα επίπεδα μονοξειδίου του άνθρακα είναι εξίσου επικίνδυνα διότι τα τυπικά συμπτώματα -όπως ο πονοκέφαλος, ο ίλιγγος, η ναυτία και η δύσπνοια (Πίνακας 1.6α)-

ανάλογα και με τα επίπεδα τιμών συγκέντρωσής του σε συνάρτηση με τις θέσης έκλυσης του ρύπου (Πίνακας 1.6β), μπορεί να θεωρηθεί ότι οφείλονται και σε πάρα πολλά άλλα αίτια. Με αυτόν τον τρόπο, αν και η επίδραση του μονοξειδίου του άνθρακα δεν είναι θανατηφόρα, είναι δυνατόν να επιφέρει ανεπανόρθωτες βλάβες στην υγεία και ειδικότερα στην καταστροφή του εγκεφάλου. Έτσι λοιπόν, άτομο που παρουσιάζει εγκεφαλικά συμπτώματα που οφείλονται σε άνω των δύο εβδομάδων έκθεση σε μονοξείδιο του άνθρακα, τότε οι πιθανότητες πλήρους ανάρρωσης είναι ελάχιστες.

Πίνακας 1.6β: *Επίπεδα τιμών συγκέντρωσης CO σε διάφορες περιοχές έκλυσής του [13].*

Θέση έκλυσης	Συγκέντρωση CO
Αστική λεωφόρος, μποτιλιαρισμένη κυκλοφορία	> 44
Καπνός τσιγάρου σε εσωτερικό κλειστού	> 87
Εσωτερικός, μη αεριζόμενος χώρος στάθμευσης	> 100
Τούνελ με έντονη κυκλοφορία	> 200 (μέγιστη ωριαία)
Περιοχή χωρίς καπνό τσιγάρου	< 20 - 50
Ορισμένα επαγγέλματα	> 100

Το μονοξείδιο του άνθρακα υπάρχει οπουδήποτε συμβαίνουν σημαντικές καύσεις. Για παράδειγμα, οι πυρκαγιές δασών παράγουν τεράστιες ποσότητες, αλλά επειδή οι φωτιές είναι ακανόνιστες στην κατανομή, τα αέρια διαλύονται από τους ανέμους που επικρατούν. Οι υψηλότερες συγκεντρώσεις στις οποίες οι άνθρωποι συνήθως εκτίθενται (εκτός από το κάπνισμα) παρατηρούνται σε πυκνοκατοικημένες αστικές περιοχές με μεγάλη κυκλοφοριακή συμφόρηση. Ο αέρας μέσα αλλά και γύρω από τους δρόμους της πόλης και τους αυτοκινητόδρομους περιέχει υψηλά επίπεδα μονοξειδίου του άνθρακα ειδικά τις ώρες κυκλοφοριακής αιχμής. Το κάπνισμα σε κλειστό αυτοκίνητο σε ώρα κυκλοφοριακής αιχμής μπορεί να προκαλέσει επίπεδα συγκέντρωσης μονοξειδίου του άνθρακα διπλάσια από εκείνο του ορίου ασφαλείας για την προστασία την ανθρώπινης υγείας. Εσωτερικός, μη αεριζόμενος χώρος στάθμευσης, κτίρια κατά μήκος αστικών δρόμων και

τούνελ με έντονη κυκλοφορία περιέχουν τα υψηλότερα επίπεδα όπως φαίνεται στον Πίνακα 1.6β.

Η δηλητηρίαση από μονοξείδιο του άνθρακα προκαλεί διάφορα συμπτώματα, τα πιο συχνά, όμως, μοιάζουν με αυτά της τροφικής δηλητηρίασης. Τα παιδιά έχοντας υψηλότερους ρυθμούς μεταβολισμού, έχουν ανάγκη μεγαλύτερων ποσοτήτων οξυγόνου από τους ενήλικες και για το λόγο αυτό, είναι πιο ευπρόσβλητα σε δηλητηρίαση από μονοξείδιο του άνθρακα. Η δηλητηρίαση από μονοξείδιο του άνθρακα στην παιδική ηλικία μπορεί να οδηγήσει σε νευρολογικά προβλήματα, απώλεια μνήμης, μαθησιακές δυσκολίες, προβλήματα σωματικής και πνευματικής ανάπτυξης κ.λπ.

Εξετάζοντας τις πηγές έκλυσης μεγάλων ποσοτήτων μονοξειδίου του άνθρακα γίνεται φανερό πως τα μεγάλα αστικά κέντρα ευθύνονται για την έκλυσή του και ταυτόχρονα υποφέρουν από τις συνέπειες. Στους μεγάλους αυτοκινητόδρομους, στα κέντρα των πόλεων και ιδιαίτερα σε δρόμους με υψηλά κτίρια και χωρίς καλό αερισμό αλλά και κατά τη διάρκεια του χειμώνα με την επίδραση δυσμενών μετεωρολογικών συνθηκών (θερμοκρασιακή αναστροφή) το πρόβλημα των επιπτώσεων είναι εντονότερο και πολύ πιο επικίνδυνο (Omaye 2002).

Πόλεις που υπέφεραν σε πολύ μεγάλο βαθμό από το μονοξείδιο του άνθρακα ήταν η Νέα Υόρκη, η Σαγκάη, το Τόκιο κ.ά. Φυσικά και τα ελληνικά αστικά κέντρα, με έμφαση στην Αθήνα, παρατηρήθηκαν υψηλά επίπεδα συγκέντρωσης μονοξειδίου του άνθρακα (Viras et al. 1996), παρόλα αυτά, από το έτος 1990 και μετά, λόγω των μέτρων ανανέωσης του στόλου των βενζινοκίνητων οχημάτων και αντικατάστασής τους με οχήματα εφοδιασμένα με κινητήρες νέας τεχνολογίας (Viras et al. 1996). Τέλος, από το 2006 δεν σημειώθηκε υπέρβαση της οριακής τιμής του ρύπου (Οδηγία 2000/69/EK).

1.3.2 Επιπτώσεις στην υγεία

Τα πιο συνηθισμένα συμπτώματα δηλητηρίασης από μονοξείδιο του άνθρακα, μπορεί να μοιάζουν με άλλους τύπους δηλητηρίασης και λοίμωξης, αλλά περιλαμβάνουν συμπτώματα όπως πονοκέφαλο, ναυτία, εμετό, ζαλάδα, κούραση και ένα αίσθημα αδυναμίας. Οι οικογένειες που δηλητηριάζονται από μονοξείδιο του άνθρακα συχνά πιστεύουν ότι είναι θύματα τροφικής δηλητηρίασης. Τα βρέφη μπορεί να είναι ευερέθιστα και ανόρεκτα. Τα νευρολογικά σημάδια περιλαμβάνουν σύγχυση, αποπροσανατολισμό, οπτικές διαταραχές, συγκοπή και επιληπτικές κρίσεις.

Το μονοξείδιο του άνθρακα είναι ικανό να επηρεάσει σημαντικά τον αερόβιο μεταβολισμό του ανθρώπου εξαιτίας αντίδρασης του με την αιμοσφαιρίνη, τη χημική ουσία των ερυθρών αιμοσφαιρίων που μεταφέρει το οξυγόνο από τους πνεύμονες και το δίνει στους ιστούς. Η έλξη αυτή είναι από 210-250 φορές ισχυρότερη από την τάση που έχει το οξυγόνο να ενωθεί με την αιμοσφαιρίνη με αποτέλεσμα, αντί να σχηματίζεται η οξυ – αιμοσφαιρίνη, να σχηματίζεται η καρβοξυ – αιμοσφαιρίνη.

Σύμφωνα με τον Κωτσοβίνο (1992), η απορρόφηση από τον οργανισμό του μονοξειδίου του άνθρακα αυξάνεται μέσω της αύξησης της συγκέντρωσης του στην ατμόσφαιρα, της αύξησης της διάρκειας έκθεσης του ανθρώπου σε αυτό, όπως και της αύξησης του ρυθμού αναπνοής. Έχει παρατηρηθεί ότι η απορρόφηση του μονοξειδίου του άνθρακα από τον ανθρώπινο οργανισμό επηρεάζει σημαντικά το σύστημα των ενζύμων και την οξυγόνωση των ιστών. Επίσης επηρεάζει την ανθρώπινη συμπεριφορά και την κίνηση και σχετίζεται με καρδιαγγειακές παθήσεις. Σε πολύ υψηλές συγκεντρώσεις του στην ατμόσφαιρα μπορεί να προκαλέσει πονοκεφάλους, ζαλάδες, ίλιγγους, ασφυξία μέχρι και θάνατο. Σε μικρότερες συγκεντρώσεις του μια πιθανή αλλαγή της ανθρώπινης συμπεριφοράς, η οποία μπορεί να προέρχεται από τη μεταβολή σημαντικών λειτουργιών του οργανισμού, εμφανίζεται είτε με διαταραχές στο νευρικό σύστημα, είτε με επιδείνωση στο χρονικό διάστημα διάκρισης, οπτικής άμβλυωσης και ικανότητας αντίληψης κάποιων καταστάσεων. Για παράδειγμα, υψηλές συγκεντρώσεις μονοξειδίου του άνθρακα σε αυτοκινητόδρομους, μπορεί να επηρεάσουν τόσο πολύ τις

αντιδράσεις ενός οδηγού με αποτέλεσμα μέχρι και την πρόκληση τροχαίου ατυχήματος (Καρβούνης και Γεωργακέλλος 2003).

Από την Αμερικανική Υπηρεσία προστασίας του Περιβάλλοντος (USEPA) έχουν διατυπωθεί κριτήρια ποιότητας του ατμοσφαιρικού αέρα που σχετίζονται με συμπτώματα (Πίνακας 1.6γ) που προκαλούνται στον άνθρωπο σαν συνέπεια έκθεσης σε μονοξείδιο του άνθρακα και σχετίζονται άμεσα με το ποσοστό καρβοξυ-αιμοσφαιρίνης στο αίμα (Boubel et al. 1994).

Πίνακας 1.6γ: Σχέση μεταξύ ποσοστού καρβοξυ-αιμοσφαιρίνης στο ανθρώπινο αίμα και εμφανιζόμενων συμπτωμάτων σαν συνέπεια έκθεσης σε μονοξείδιο του άνθρακα (Boubel et al. 1994).

Ποσοστό στο ανθρώπινο αίμα	Εμφανιζόμενα συμπτώματα
3-6	Επιδείνωση καρδιαγγειακών βλαβών, μείωση εργασιακής ικανότητας σε ασθενείς με στεφανιαίο σύνδρομο όπως και περιφερειακά φαινόμενα αρτηριοσκλήρυνσης
4	Μείωση ετοιμότητας εργασιακής απόδοσης σε υγιείς αλλά και σε χρονίως πάσχοντες από καρδιοπνευμονικά προβλήματα
5	Μείωση οπτικού φάσματος, των επιδεξιοτήτων, εκτέλεση περιέργων αισθητικοκινητικών αντιδράσεων
20	Καρδιαγγειακές παθήσεις
30	Πονοκέφαλος, κόπωση
40	Κατάρρευση δυνάμεων, σύγχυση
60	Απώλεια συνείδησης, θάνατος (σε περίπτωση διατήρησης του επιπέδου του ποσοστού)
80	Θάνατος

Ιδιαίτερα επιβαρημένη είναι η κατάσταση ανάμεσα σε ασθενείς με χρόνιες καρδιοπνευμονικές παθήσεις, οι οποίοι εκτείνονται ακόμα και σε μικρές συγκεντρώσεις μονοξειδίου του άνθρακα. Αυτό εξηγείται από το γεγονός ότι η καρδιά του ανθρώπου είναι πιο ευαίσθητη στην έλλειψη οξυγόνου, διότι ενώ οι περιφερειακοί ιστοί του σώματος παίρνουν περίπου το

25% από το οξυγόνο του αίματος των αρτηριών, ο καρδιακός μυς χρειάζεται τριπλάσια ποσότητα για τη σωστή λειτουργία του. Επιδημιολογικές μελέτες έδειξαν μια σημαντική εμφάνιση μονίμων καρδιακών παθήσεων από παρατεταμένες εκθέσεις σε υψηλές συγκεντρώσεις μονοξειδίου του άνθρακα το οποίο δείχνει να ευθύνεται κατά παρόμοιο τρόπο με τη χοληστερόλη για τη δημιουργία ενός λιπαρού στρώματος πάνω στα αιμοφόρα αγγεία (αθηρωματική πλάκα) και την τάση συγκόλλησης των αιμοπεταλίων μεταξύ τους. Ωστόσο, υπάρχουν επιδημιολογικά στοιχεία τα οποία δείχνουν συσχέτιση ανάμεσα σε εισαγωγές ατόμων που εισήχθησαν σε νοσοκομεία από έμφραγμα του μυοκαρδίου και στην έκθεση τους σε μέσες εβδομαδιαίες συγκεντρώσεις μονοξειδίου του άνθρακα από 9-16 mg/m³ (8-14ppm) (Γεντεκάκης 2003).

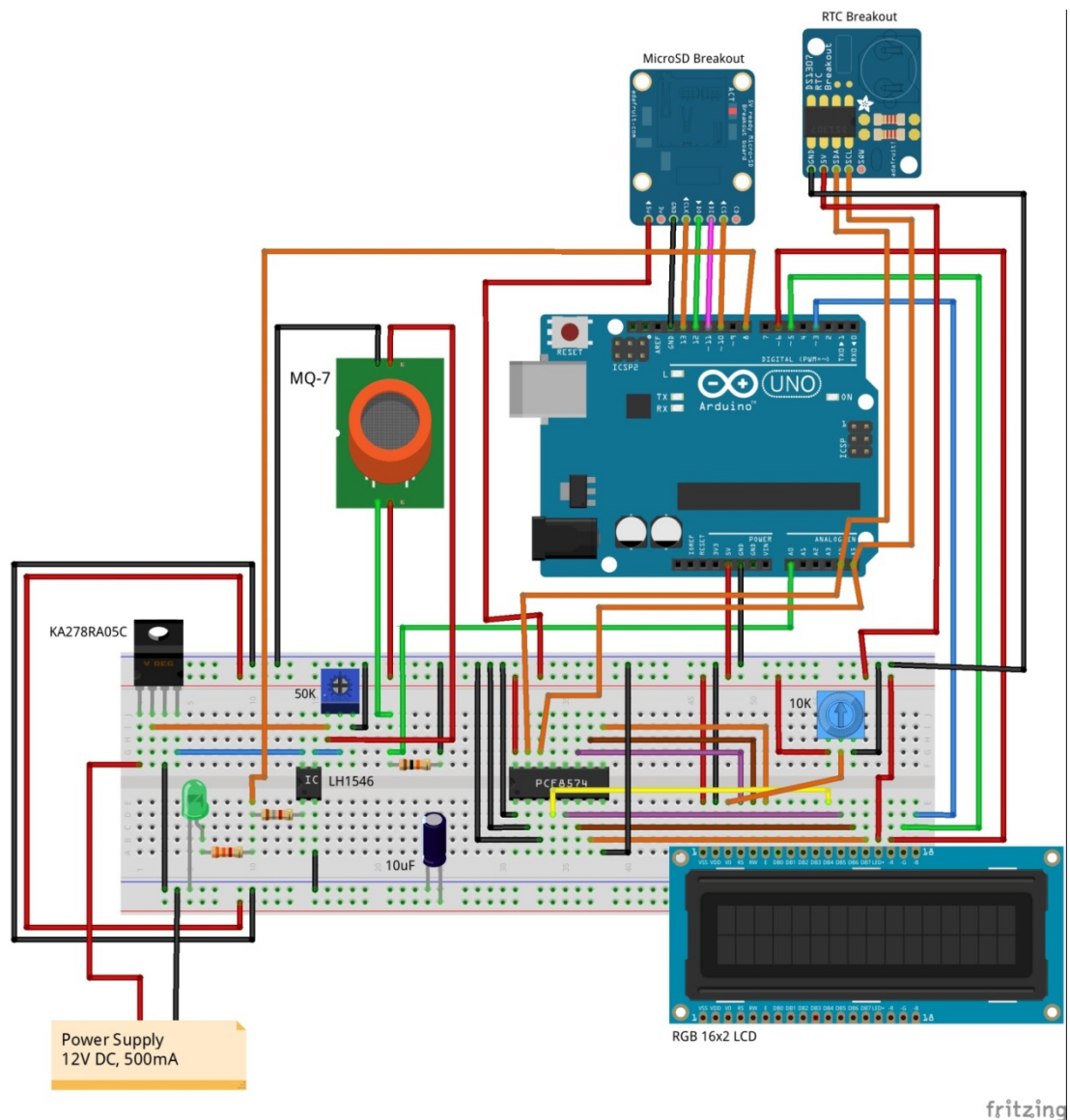
ΚΕΦΑΛΑΙΟ 2 – ΥΛΙΚΟ ΜΕΡΟΣ ΣΥΣΤΗΜΑΤΟΣ

Η υλοποίηση του συστήματος μέτρησης του μονοξειδίου του άνθρακα βασίστηκε στην πλακέτα του Arduino UNO και στήθηκε γύρω από αυτό. Η αρχιτεκτονική Arduino επιτρέπει τη σύνδεση διαφόρων εξαρτημάτων όπως αισθητήρες, led, συσκευές διαφόρων λειτουργιών κτλ. τα οποία σε συνδυασμό με τις κατάλληλες προγραμματιστικές εντολές εκτελούν συγκεκριμένες λειτουργίες. Τα εξαρτήματα που είναι συνδεδεμένα στο Arduino UNO αναφέρονται παρακάτω:

- Αισθητήρας μονοξειδίου του άνθρακα MQ-7
- SD Datalogger Shield
- LCD RGB οθόνη 16x2 μέσω ενός I²C Expander PCF8574A

Αυτό που υλοποιεί ο μικροελεγκτής Arduino UNO είναι να συλλέγει την τιμή του μονοξειδίου του άνθρακα στο περιβάλλον που μετρήθηκε από τον αισθητήρα MQ-7. Στη συνέχεια το Arduino στέλνει δεδομένα στην οθόνη LCD ώστε να προβάλλεται η τιμή μέτρησης του μονοξειδίου του άνθρακα εκείνη τη χρονική στιγμή. Επίσης σε συγκεκριμένα χρονικά διαστήματα το Arduino δίνει εντολή στον SD Datalogger να καταγράψει στην καρτα SD την τιμή του μονοξειδίου του άνθρακα καθώς και άλλες πληροφορίες όπως ώρα και ημερομηνία της στιγμής που έγινε η μέτρηση.

Στη συνέχεια απεικονίζεται το σχέδιο του κυκλώματος που υλοποιήθηκε ώστε να πραγματοποιηθούν οι λειτουργίες που αναφέρθηκαν παραπάνω.



Εικόνα 2.1: Σχέδιο κυκλώματος Arduino - εξαρτημάτων.

Το σχέδιο κυκλώματος υλοποιήθηκε με το πρόγραμμα σχεδίασης και απεικόνισης κυκλωμάτων με την ονομασία Fritzing. Επειδή δεν υπήρχε η δυνατότητα να απεικονιστεί γραφικά το SD Datalogger shield χρησιμοποιήθηκαν για την απεικόνιση το MicroSD Breakout και το RTC Breakout τα οποία εξαρτήματα ισοδυναμούν με το Datalogger shield. Επίσης οι ακροδέκτες που χρησιμοποιούν από το Arduino είναι ακριβώς οι ίδιοι με αυτούς που δεσμεύει το SD Datalogger shield.

2.1 SD Datalogger Shield

2.1.1 Τι είναι ο καταχωρητής δεδομένων (Data logger)

Ο καταχωρητής δεδομένων είναι μια ηλεκτρονική συσκευή που καταγράφει δεδομένα σε συνάρτηση με το χρόνο ή και σε σχέση με την τοποθεσία είτε με μια ενσωματωμένη συσκευή, είτε με αισθητήρα, είτε μέσω εξωτερικών οργάνων και αισθητήρων. Με το πέρασμα του χρόνου βασίζονται όλο και περισσότερο σε έναν ψηφιακό επεξεργαστή ή κάποιον ηλεκτρονικό υπολογιστή. Συνήθως είναι μικροί, λειτουργούν με μπαταρία, είναι φορητοί και εξοπλισμένοι με μικροεπεξεργαστή, εσωτερική ή αποσπώμενη μνήμη (κάρτες μνήμης κλπ) για αποθήκευση δεδομένων και αισθητήρες για μέτρηση και ανάλυση διαφόρων τιμών. Ορισμένοι καταχωρητές δεδομένων έχουν τη δυνατότητα να συνδεθούν σε ηλεκτρονικό υπολογιστή και να χρησιμοποιήσουν λογισμικό που ενεργοποιεί τον καταχωρητή για να προβληθούν και να αναλυθούν τα δεδομένα που συλλέχτηκαν, ενώ άλλοι καταγραφείς, έχουν μια τοπική συσκευή διεπαφής, όπως πληκτρολόγιο και οθόνη lcd, και μπορούν να χρησιμοποιηθούν ως αυτόνομες συσκευές.

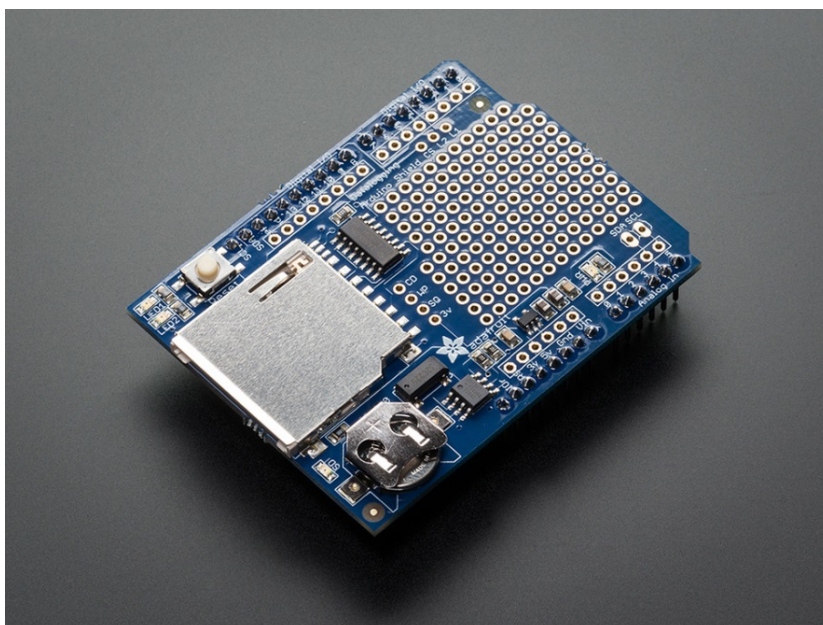
Οι καταχωρητές δεδομένων ποικίλλουν από γενικής χρήσης με ένα εύρος εφαρμογών μέτρησης μέχρι εξειδικευμένες συσκευές για μέτρηση κάποιας τιμής σε συγκεκριμένο περιβάλλον και συνθήκες ή συγκεκριμένο τύπο εφαρμογής. Οι καταχωρητές δεδομένων γενικής χρήσης συνήθως είναι προγραμματιζόμενοι, ωστόσο, πολλοί παραμένουν στατικοί με λίγες έως καθόλου μεταβλητές παραμέτρους. Οι ηλεκτρονικοί καταχωρητές δεδομένων έχουν αντικαταστήσει σε πολλές εφαρμογές τους καταγραφείς με πίνακα.

Ένα από τα πλεονεκτήματα της χρήσης του καταχωρητή δεδομένων είναι η δυνατότητα να συλλέγονται δεδομένα αυτόματα σε 24ωρη βάση. Μετά την ενεργοποίησή τους, οι καταγραφείς δεδομένων μπαίνουν σε λειτουργία χωρίς ανάγκη επιτήρησης και καταγράφουν πληροφορίες καθ' όλη τη διάρκεια της καθορισμένης περιόδου καταγραφής. Έτσι παρέχεται μια κατανοητή και ακριβής εικόνα των περιβαλλοντολογικών συνθηκών που παρατηρούνται, όπως η θερμοκρασία του αέρα και η σχετική υγρασία.

Το κόστος των καταχωρητών δεδομένων έχει μειωθεί με τα χρόνια καθώς η τεχνολογία εξελίσσεται και οι τιμές πέφτουν. Οι απλοί καταγραφείς δεδομένων κοστίζουν γύρω στα 25 ευρώ, ενώ πιο εξειδικευμένοι καταχωρητές μπορεί να κοστίζουν εκατοντάδες ή και χιλιάδες ευρώ.

2.1.2 Adafruit Assembled Data Logging shield for Arduino

Για την υλοποίηση της εργασίας χρησιμοποιήθηκε το data logging shield της Adafruit.



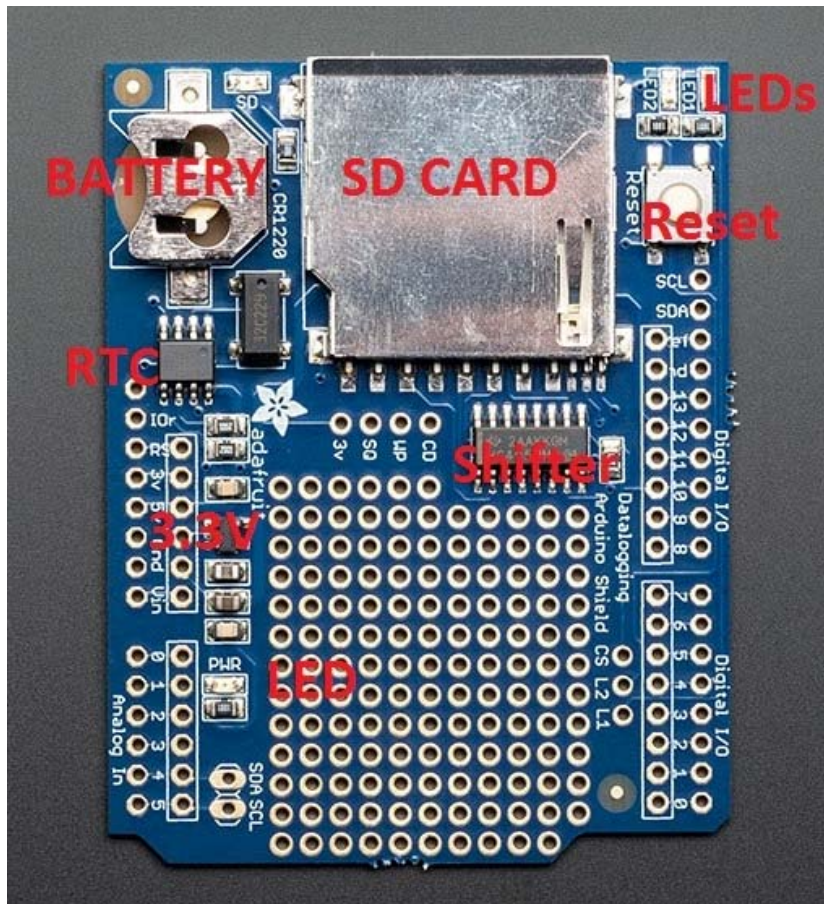
Εικόνα 2.2: *Adafruit data logging shield.*

Το συγκεκριμένο shield καταχωρεί δεδομένα σε αρχεία και τα αποθηκεύει σε κάρτα μνήμης SD. Στη συνέχεια μέσω ενός SD Card Reader μπορούν να έχουν πρόσβαση στα αρχεία της κάρτας SD σε διάφορα προγράμματα, όπως προγράμματα διαγραμμάτων, υπολογιστικών φύλλων, ανάλυσης στοιχείων και λοιπών λειτουργιών. Επίσης διαθέτει Ρολόι Πραγματικού Χρόνου (Real Time Clock) ώστε να υπάρχει η δυνατότητα να αποθηκευθεί η ακριβής ώρα και ημερομηνία όπου καταχωρήθηκαν τα δεδομένα με αποτέλεσμα ο χρήστης να γνωρίζει ποια χρονική στιγμή έγινε

κάποια ενέργεια. Το data logging shield έχει αξιόπιστο και έξυπνο σχεδιασμό και έχει υποστήριξη από πολλές βιβλιοθήκες. Συνδέεται εύκολα πάνω στο Arduino χωρίς επιπλέον τροφοδοσία καθώς χρησιμοποιεί την τροφοδοσία του Arduino. Μερικά από τα χαρακτηριστικά του shield αναφέρονται παρακάτω:

- Η διεπαφή (interface) της SD λειτουργεί με κάρτες με format FAT16 και FAT32. Το κύκλωμα του level shifter που λειτουργεί στα 3,3V προστατεύει την κάρτα από το να καταστραφεί.
- Το ρολόι πραγματικού χρόνου (RTC) συνεχίζει να κρατάει την ώρα ακόμα και αν το shield είναι αποσυνδεδεμένο και αυτό γίνεται επειδή μια μπαταρία 3V τροφοδοτεί το RTC όταν δεν τροφοδοτείται από το Arduino.
- Υπάρχει μια περιοχή στο shield όπου μπορούν να συνδεθούν κυκλώματα και αισθητήρες.
- Περιλαμβάνονται ρυθμιζόμενα leds για διάφορες ενδείξεις.
- Ο σταθεροποιητής τάσης στα 3,3V που υπάρχει στο κύκλωμα μπορεί να χρησιμοποιηθεί και ως τάση αναφοράς και επίσης βοηθάει στο να λειτουργούν σωστά οι κάρτες SD που συνδέονται, οι οποίες απαιτούν πολύ ισχύ για να τη λειτουργία τους.

Παρακάτω θα γίνει περεταίρω ανάλυση του datalogger shield.



Εικόνα 2.3: Περιγραφή των εξαρτημάτων του datalogger shield.

Πάνω αριστερά βρίσκεται το ρολόι πραγματικού χρόνου (RTC) στο οποίο είναι συνδεδεμένα ένας κρύσταλλος και μια εφεδρική μπαταρία μακράς διάρκειας η οποία τροφοδοτεί το RTC όταν το shield είναι αποσυνδεδεμένο από το Arduino.

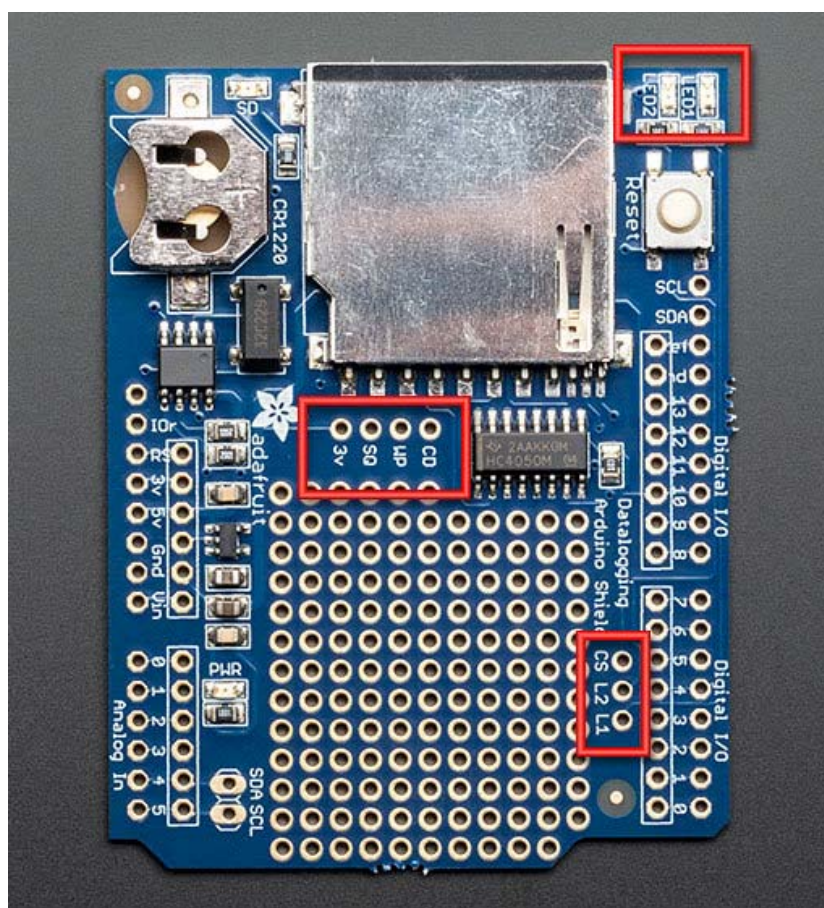
Στη μέση και αριστερά βρίσκεται ένας ενσωματωμένος σταθεροποιητής τάσης στα 3,3V ώστε τα εξαρτήματα του shield να λειτουργούν σταθερά και ομαλά. Υπάρχει επίσης και ένα πράσινο led το οποίο είναι ανοιχτό όταν βρίσκεται σε λειτουργία το shield.

Πάνω βρίσκεται η υποδοχή για την κάρτα SD στην οποία είναι συμβατή οποιαδήποτε SD/MMC χωρητικότητας μέχρι 32GB και το ελάχιστο μέγεθος

είναι 32MB. Μια microSD είναι επίσης συμβατή αρκεί να υπάρχει ένας microSD adapter ώστε να έχει το μέγεθος μιας κανονικής SD κάρτας.

Πάνω δεξιά υπάρχουν δυο ρυθμιζόμενα LED τα οποία μπορούν να συνδεθούν μέσω καλωδίου από τα L1 και L2 σε οποιοδήποτε ψηφιακό ακροδέκτη του Arduino. Επίσης ακριβώς από κάτω βρίσκεται το κουμπί Reset το οποίο επανακινεί το Arduino.

Στη μέση βρίσκεται ένας level shifter ο οποίος προστατεύει την κάρτα SD από σήματα του Arduino της τάξης των 5V τα οποία θα μπορούσαν να την καταστρέψουν. Λειτουργεί επίσης και με σήματα της τάξης των 3V.

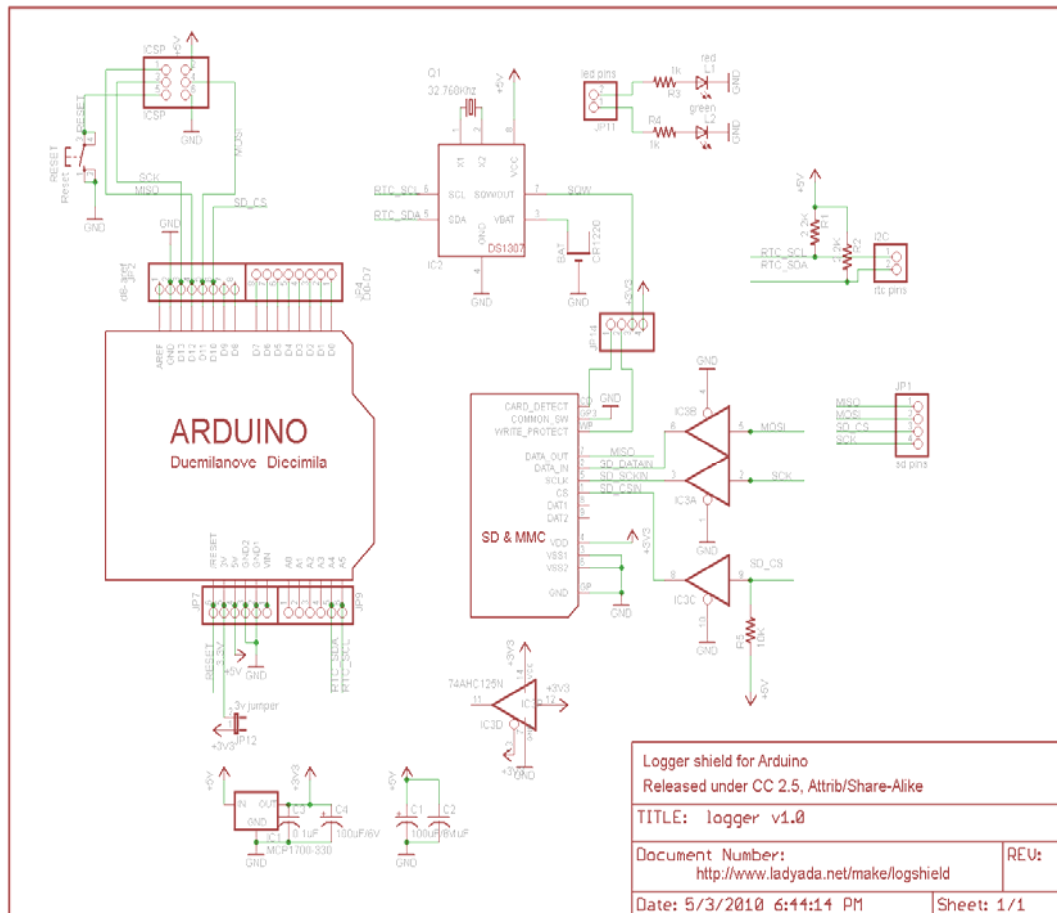


Εικόνα 2.4: Πρόσθετα led και ακροδέκτες του shield.

Εκτός από τα παραπάνω υπάρχουν και κάποιοι πρόσθετοι ακροδέκτες τους οποίους θα αναφέρουμε παρακάτω:

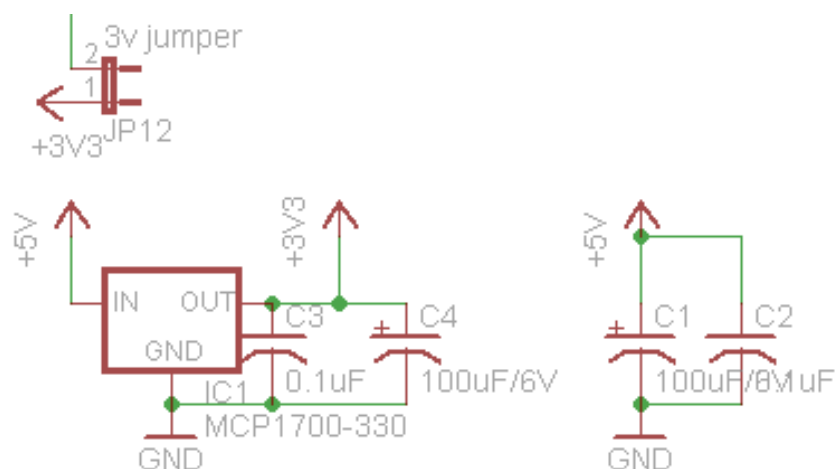
- 3V - Ο συγκεκριμένος ακροδέκτης είναι η 3V έξοδος του σταθεροποιητή τάσης. Είναι μια ποιοτική 3.3V τάση αναφοράς η οποία μπορεί να τροφοδοτήσει αισθητήρες και άλλα εξαρτήματα και παρέχει ρεύμα μέχρι 50mA.
- SQ – Ο ακροδέκτης αυτός είναι μια έξοδος τετραγωνικού παλμού προερχόμενος από το RTC. Με αυτόν τον τρόπο μπορούμε να πάρουμε τετραγωνικό παλμό ακριβείας ο οποίος χρησιμοποιείται κυρίως για διάφορες δοκιμές.
- WP – Αυτός ο ακροδέκτης χρησιμεύει στο να εντοπιστεί αν υπάρχει προστασία εγγραφής στην κάρτα SD ελέγχοντας την έξοδο του.
- CD – Είναι ο ακροδέκτης που ελέγχει αν έχει εισαχθεί μια κάρτα SD στην υποδοχή. Όταν ο ακροδέκτης CD είναι συνδεδεμένος στη γείωση σημαίνει ότι έχει εισαχθεί μια SD.
- CS – Αυτός είναι ο ακροδέκτης Chip Select για την κάρτα SD. Κανονικά το Chip Select είναι συνδεδεμένο με το ψηφιακό pin 10 του Arduino αλλά σε περίπτωση που το pin 10 είναι δεσμευμένο για κάποια άλλη λειτουργία υπάρχει η δυνατότητα μέσω του CS να συνδεθεί σε κάποιο άλλο ψηφιακό pin.
- L1 και L2 – Είναι οι ακροδέκτες των προαιρετικών led LED1 και LED2 αντίστοιχα. Ανάμεσα στα led και τους ακροδέκτες υπάρχει ήδη συνδεδεμένη σε σειρά από έναν αντιστάτη στα 470Ω.

Στη συνέχεια απεικονίζεται το σχηματικό του shield και αναλύεται το κάθε κομμάτι ξεχωριστά.



Εικόνα 2.5: Το σχηματικό του datalogger shield.

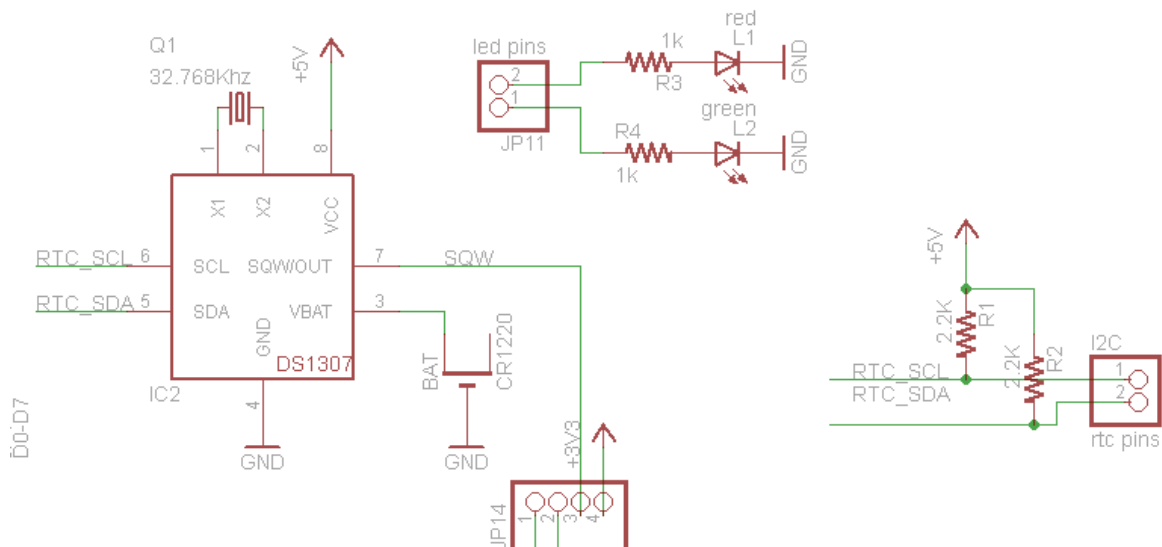
2.1.2.1 Τροφοδοσία



Εικόνα 2.6: Το σχηματικό του κυκλώματος της τροφοδοσίας του shield.

Στο shield υπάρχει μια πηγή τροφοδοσίας ώστε να παράγει 3,3V με ρεύμα στα 250mA. Δεν χρησιμοποιείται ο ενσωματωμένος στο Arduino σταθεροποιητής τάσης στα 3,3V επειδή παρέχει ρεύμα μέχρι 50mA και μερικές κάρτες SD χρειάζονται αρκετό ρεύμα κατά τη διάρκεια εγγραφής. Η συγκεκριμένη πηγή είναι αρκετά σταθερή και μπορεί να χρησιμοποιηθεί και ως τάση αναφοράς. Επίσης, είναι συνδεδεμένοι στην πηγή δυο σετ πυκνωτών παράκαμψης (bypass) και σκοπό έχουν τη σωστή λειτουργία των πηγών 3,3V και 5V. Οι πυκνωτές των 100µF απαλείφουν τους θορύβους χαμηλής συχνότητας ενώ αυτοί των 0,1µF απαλείφουν τους θορύβους υψηλής συχνότητας.

2.1.2.2 Το Ρολόι Πραγματικού Χρόνου (Real Time Clock):

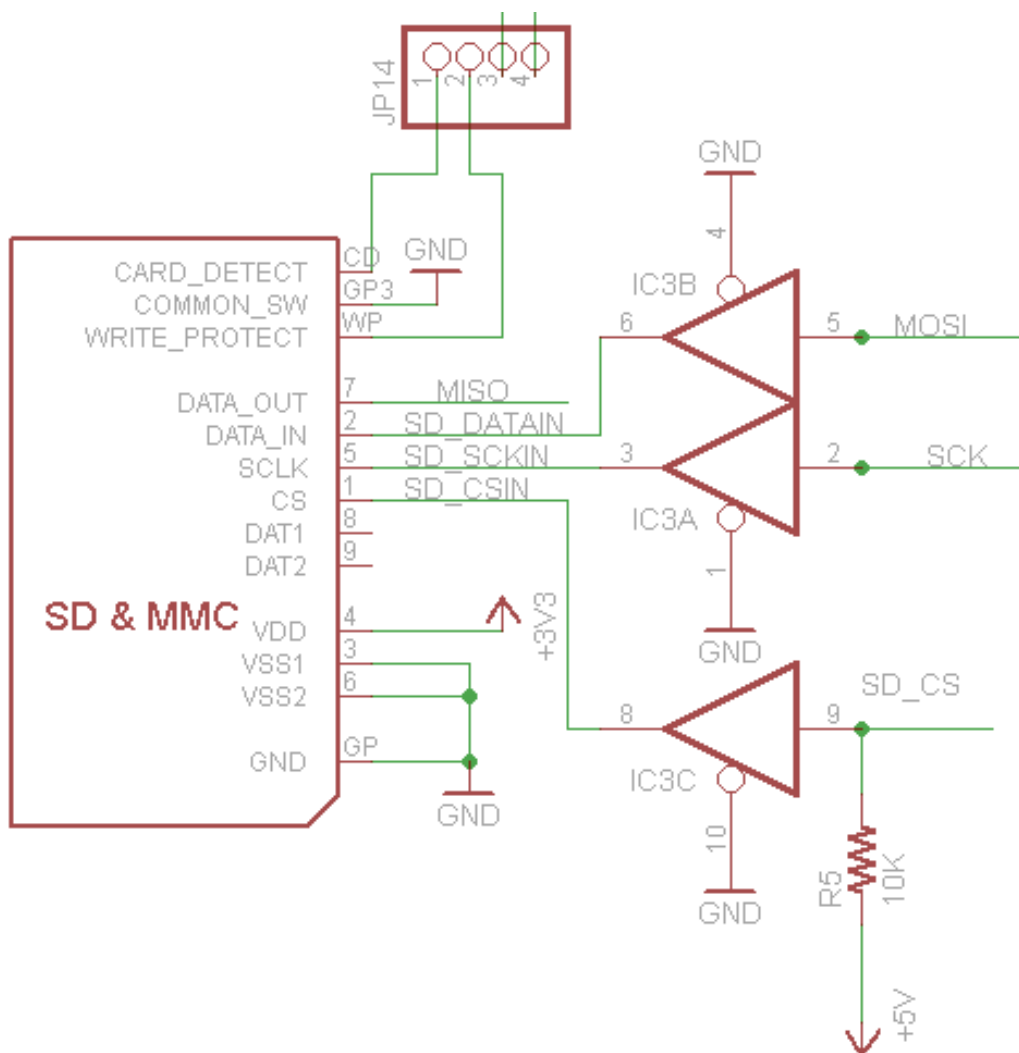


Εικόνα 2.7: Το σχηματικό του κυκλώματος για το ρολόι πραγματικού χρόνου.

Το ρολόι πραγματικού χρόνου είναι το ολοκληρωμένο DS1307 από την Maxim, στο οποίο είναι συνδεδεμένη μια εφεδρική μπαταρία (CR1220), και επικοινωνεί με το Arduino μέσω του πρωτοκόλλου i2c (οι αγωγοί SCL και SDA). Το πρωτόκολλο i2c απαιτεί pullup αντιστάτες στους αγωγούς του (ρολογιού και δεδομένων) και γι αυτό είναι συνδεδεμένοι από ένας αντιστάτης των 2,2KΩ σε κάθε αγωγό (R1 και R2). Για το RTC απαιτείται επίσης και ένας κρύσταλλος χωρητικότητας 12,5pF χρονισμένος στα 32,768 KHz (Q1).

Υπάρχουν επίσης και δυο led τα οποία μπορούν να ρυθμιστούν και να προγραμματιστούν όπως θέλει ο χρήστης.

2.1.2.3 Η διεπαφή της κάρτας SD:



Εικόνα 2.8: Το σχηματικό του κυκλώματος της διεπαφής της κάρτας SD.

Η υποδοχή της κάρτας SD είναι συνδεδεμένη στο Arduino μέσω του buffer IC3. Ο buffer είναι ένας level shifter και μετατρέπει τα σήματα των 5V σε σήματα των 3,3V τα οποία είναι πιο ασφαλή για τις SD, σε σχέση με αυτά των 5V. Στον ακροδέκτη CS είναι συνδεδεμένος ένας pull up αντιστάτης ώστε να προστατεύει την κάρτα SD σε περίπτωση που το Arduino προγραμματίζεται από έναν ISP προγραμματιστή. Στην κάρτα SD υπάρχουν

και άλλοι δυο ακροδέκτες: Ο ακροδέκτης Card Detect ο οποίος συνδέεται στη γείωση όταν έχει εισαχθεί μια κάρτα SD και ο ακροδέκτης Write Protect ο οποίος συνδέεται στη γείωση όταν εντοπιστεί ότι στην κάρτα SD που έχει εισαχθεί υπάρχει προστασία εγγραφής.

2.1.3 Οι ακροδέκτες του Arduino που χρησιμοποιεί το datalogger shield.

Το datalogger shield χρησιμοποιεί 6 ακροδέκτες του Arduino. Οι ακροδέκτες αναλογικής εισόδου 4 και 5 (A4 και A5) είναι αυτοί που χρησιμοποιούνται για το i2c πρωτόκολλο. Η κάρτα SD χρησιμοποιεί τα ψηφιακά pin 10, 11, 12 και 13. Οι αγωγοί που συνδέονται στα pin 11, 12 και 13 δε γίνεται να συνδεθούν σε κάποιο άλλο ακροδέκτη. Για τον αγωγό που συνδέεται στο pin 10 όμως υπάρχει η δυνατότητα να συνδεθεί σε κάποιον άλλο ψηφιακό ακροδέκτη, σε περίπτωση που το pin 10 χρειάζεται για κάποια άλλη λειτουργία, αρκεί να ενημερωθεί το αρχείο με το Header στη βιβλιοθήκη. Επίσης ακροδέκτης 10 θα πρέπει να προγραμματιστεί ως έξοδος διότι σε άλλη περίπτωση η διεπαφή της κάρτας SD δε θα λειτουργήσει.

2.2 Ο αισθητήρας μονοξειδίου του άνθρακα MQ-7

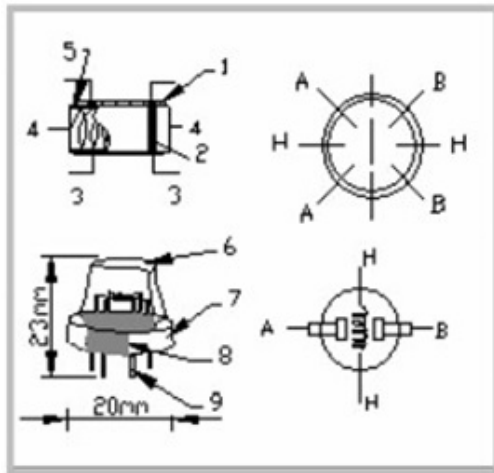
Για την υλοποίηση της εργασίας χρησιμοποιήθηκε ο αισθητήρας MQ-7. Ο MQ-7, που μπορεί να εντοπίσει συγκεντρώσεις μονοξειδίου του άνθρακα οπουδήποτε και οι μετρήσεις που κάνει είναι από 20 μέχρι 2000ppm. Είναι απλός στη χρήση και είναι κατάλληλος για μέτρηση των συγκεντρώσεων στον αέρα.

Ο MQ-7 έχει υψηλή ευαισθησία και γρήγορο χρόνο απόκρισης. Η έξοδος του αισθητήρα είναι μια αναλογική αντίσταση.



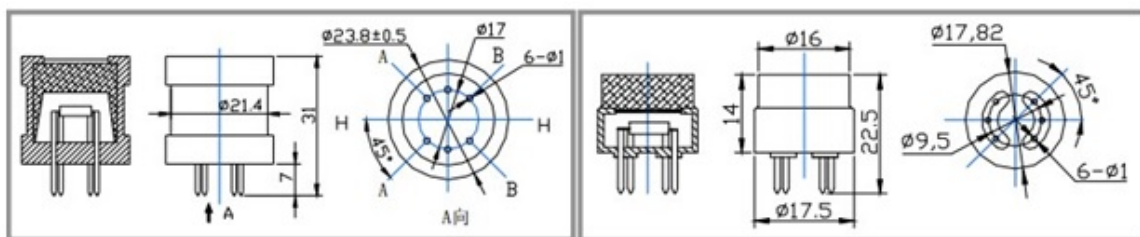
Εικόνα 2.9: Ο αισθητήρας MQ-7.

Ο αισθητήρας αποτελείται από ένα micro ceramic tube οξειδίου του αργιλίου (AL_2O_3), μια λεπτή στρώση διοξειδίου του κασσίτερου (SnO_2), ηλεκτρόδιο μέτρησης και ένα heater. Όλα αυτά είναι συγκεντρωμένα σε μια μάζα φτιαγμένη από πλαστικό και διχτάκι από ανοξείδωτο ασάλι. Ο heater παρέχει κατάλληλες συνθήκες λειτουργίας ώστε να λειτουργούν σωστά τα ευαίσθητα εξαρτήματα. Το περίβλημα του MQ-7 έχει 6 ακροδέκτες: 4 από αυτούς χρησιμοποιούνται για παροχή σημάτων και οι άλλοι 2 χρησιμοποιούνται για την τροφοδοσία του heater. Η παρακάτω εικόνα απεικονίζει τις βασικές πληροφορίες της κατασκευής του MQ-7 όπως τις διαστάσεις, τους ακροδέκτες καθώς και τη θέση του κάθε εξαρτήματος στο εσωτερικό του αισθητήρα.



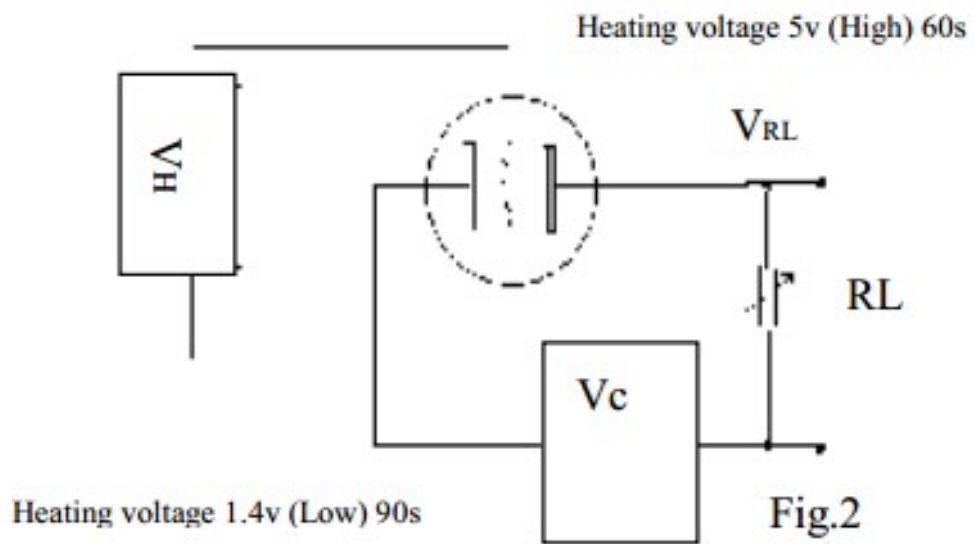
	Parts	Materials
1	Gas sensing layer	SnO ₂
2	Electrode	Au
3	Electrode line	Pt
4	Heater coil	Ni-Cr alloy
5	Tubular ceramic	Al ₂ O ₃
6	Anti-explosion network	Stainless steel gauze (SUS316 100-mesh)
7	Clamp ring	Copper plating Ni
8	Resin base	Bakelite
9	Tube Pin	Copper plating Ni

Fig.1



Εικόνα 2.10: Πληροφορίες κατασκευής MQ-7.

Τα βασικό κύκλωμα του MQ-7 με τα εξαρτήματα υψηλής ευαισθησίας που συμβάλλουν στη μέτρηση του μονοξειδίου του άνθρακα, αποτελείται από 2 μέρη. Το ένα, είναι το κύκλωμα του heater το οποίο λειτουργεί με συνάρτηση που ελέγχει το χρόνο καθώς η υψηλή και η χαμηλή τάση του heater εναλλάσσονται κυκλικά. Το δεύτερο μέρος, είναι το κύκλωμα της αναλογικής εξόδου του αισθητήρα και μπορεί να έχει ακριβή απόκριση στις αλλαγές της επιφάνειας της αντίστασης του αισθητήρα. Η παρακάτω εικόνα απεικονίζει τα μέρη του κυκλώματος που αναφέρθηκαν:



Εικόνα 2.11: Βασικό κύκλωμα MQ-7.

Παρακάτω θα αναφερθούν τα τεχνικά χαρακτηριστικά του αισθητήρα MQ-7:

Πίνακας 2.1: Τεχνικά χαρακτηριστικά MQ-7.

A. Standard work condition			
Symbol	Parameter name	Technical condition	Remark
V _c	circuit voltage	5V±0.1	Ac or Dc
V _H (H)	Heating voltage (high)	5V±0.1	Ac or Dc
V _H (L)	Heating voltage (low)	1.4V±0.1	Ac or Dc
R _L	Load resistance	Can adjust	
R _H	Heating resistance	33 Ω± 5 %	Room temperature
T _H (H)	Heating time (high)	60 ± 1 seconds	
T _H (L)	Heating time (low)	90 ± 1 seconds	
PH	Heating consumption	About 350mW	

B. Environment conditions			
Symbol	Parameters	Technical conditions	Remark
Tao	Using temperature	-20°C-50°C	
Tas	Storage temperature	-20°C-50°C	Advice using scope
RH	Relative humidity	Less than 95%RH	
O ₂	Oxygen concentration	21%(stand condition) the oxygen concentration can affect the sensitivity characteristic	Minimum value is over 2%

C. Sensitivity characteristic			
symbol	Parameters	Technical parameters	Remark
R _s	Surface resistance Of sensitive body	2-20k	In 100ppm Carbon Monoxide
a (300/100ppm)	Concentration slope rate	Less than 0.5	R _s (300ppm)/R _s (100ppm)
Standard working condition	Temperature -20°C± 2 °C relative humidity 65% ± 5% RL: 10K Ω± 5% V _c :5V±0.1V V _H :5V±0.1V V _H :1.4V±0.1V		
Preheat time	No less than 48 hours	Detecting range: 20ppm-2000ppm carbon monoxide	

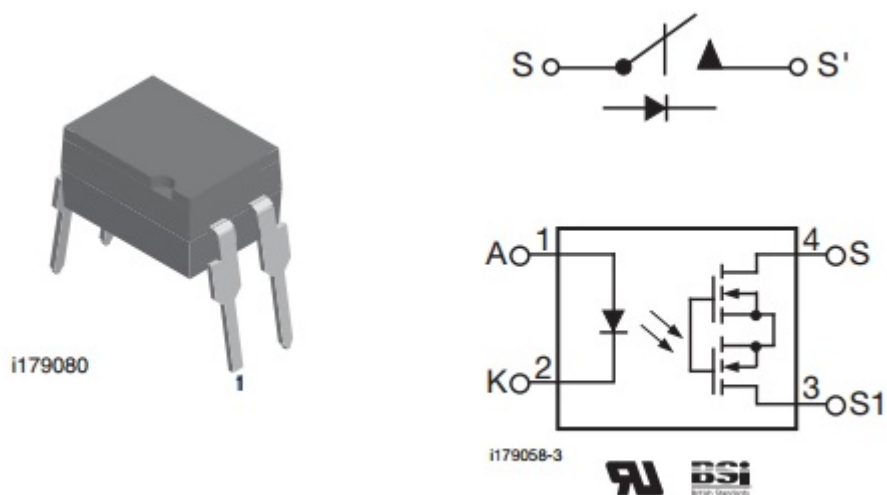
Το κύκλωμα του heater του MQ-7 λειτουργεί σε τάση 5V για 60sec και σε τάση 1.4V για 90sec και οι τάσεις εναλλάσσονται κυκλικά για όση χρονική διάρκεια βρίσκεται σε λειτουργία ο αισθητήρας. Επίσης, στην υψηλή τάση του heater (5V), απαιτείται ρεύμα της τάξης των 150mA, το οποίο σημαίνει ότι ξεπερνά το ρεύμα που μπορεί να παρέχει το Arduino (50mA). Γι αυτό το λόγο, χρησιμοποιήθηκε ένας ρυθμιζόμενος σταθεροποιητής τάσης με κωδική ονομασία KA278RA05C, ο οποίος μπορεί να παρέχει ρεύμα μέχρι 2A το

οποίο είναι υπεραρκετό. Για την τροφοδοσία του KA278RA05C χρησιμοποιήθηκε ένας μετασχηματιστής με τάση 12V και ρεύμα 500mA.



Εικόνα 2.12: Σταθεροποιητής τάσης KA278RA05C.

Η προεπιλεγμένη τάση του KA278RA05C με το pin V_{adj} αποσυνδεδεμένο είναι 5V, η οποία είναι η high τάση του heater. Για τη low τάση του heater χρησιμοποιήθηκε ένα ποτενσιόμετρο 50KΩ ώστε να ρυθμιστεί η τάση στο 1,4V. Για να είναι δυνατή η αλλαγή τάσης από τα 5V στο 1,4V χρησιμοποιήθηκε ένα solid state ρελέ το LH1546. Το LH1546 συνδέεται στον ψηφιακό ακροδέκτη 8 του Arduino. Όταν το pin 8 βρίσκεται σε κατάσταση HIGH το ρελέ ενεργοποιείται και ο σταθεροποιητής τάσης τροφοδοτεί το heater με τάση 1,4V. Όταν το pin 8 βρίσκεται σε κατάσταση LOW τότε το ρελέ απενεργοποιείται και το heater τροφοδοτείται από το KA278RA05C με τάση 5V.



Εικόνα 2.13: Solid state relay LH1546.

Παρακάτω, θα αναφερθεί η συνδεσμολογία του αισθητήρα MQ-7:

- Οι ακροδέκτες A, συνδέονται και οι δυο στον ακροδέκτη 5V του Arduino.
- Ο ακροδέκτης H+ (είναι στην τριάδα με τους A,) συνδέεται στο μεσαίο pin του ποτενσιόμετρου 50KΩ.
- Ο ακροδέκτης H- (είναι στην τριάδα με τους B), συνδέεται στη γείωση του Arduino.
- Οι ακροδέκτες B, συνδέονται μαζί στον αναλογικό ακροδέκτη A0 του Arduino. Επίσης στην ένωση των ακροδεκτών B, είναι συνδεδεμένος ένας αντιστάτης 10KΩ παράλληλα στη γείωση του Arduino.

Η συνδεσμολογία του σταθεροποιητή τάσης KA278RA05C είναι η εξής:

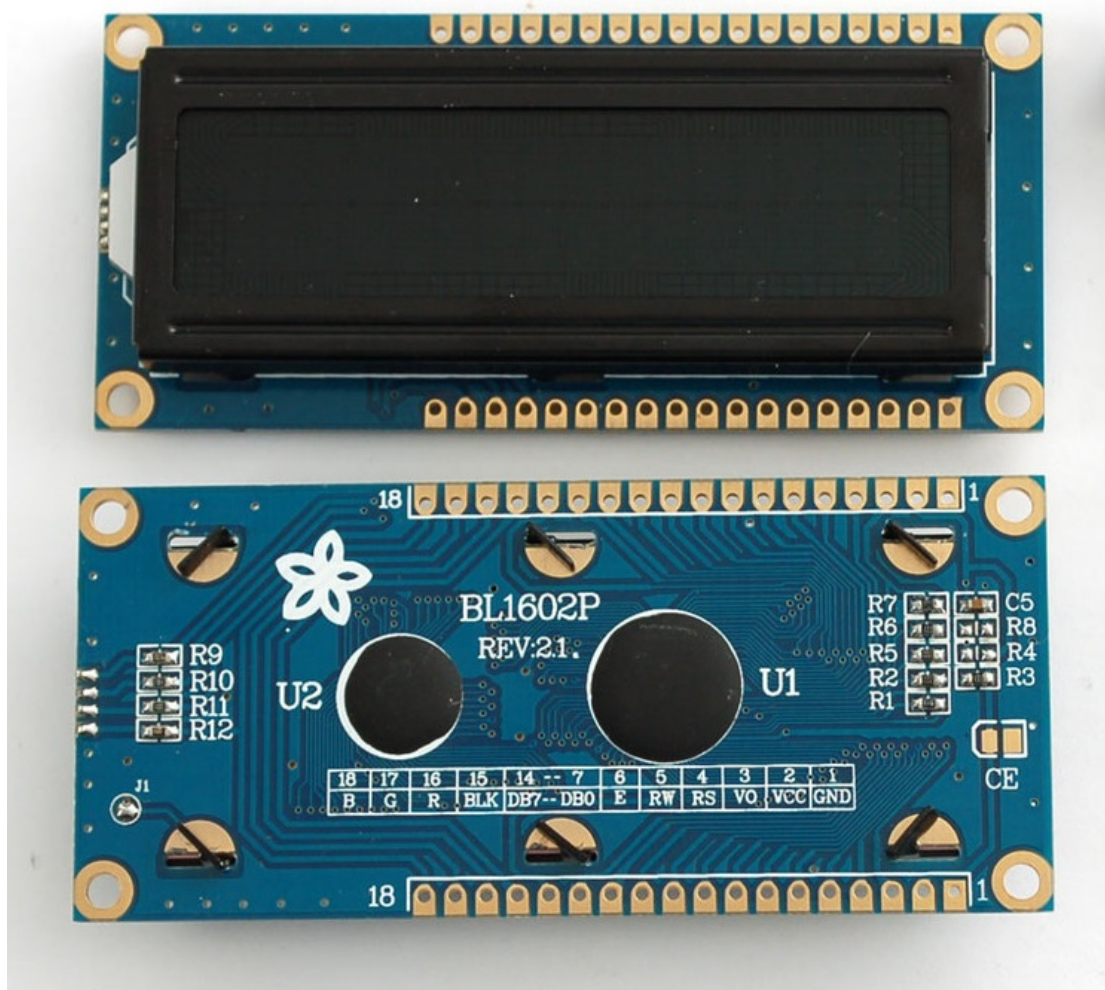
- Ο ακροδέκτης V_{IN} , συνδέεται στον θετικό ακροδέκτη της πηγής 12V.
- Ο ακροδέκτης V_O , συνδέεται σε ένα ακραίο pin του ποτενσιόμετρου 50KΩ (δεν έχει σημασία σε ποιο). Το άλλο ακραίο pin του ποτενσιόμετρου 50KΩ, συνδέεται στη γείωση του Arduino.
- Ο ακροδέκτης GND, συνδέεται στη γείωση του Arduino.
- Ο ακροδέκτης V_{adj} , συνδέεται στον ακροδέκτη S του LH1546.

Η συνδεσμολογία του solid state μελέ LH1546 αναφέρεται παρακάτω:

- Ο ακροδέκτης A, συνδέεται στο ψηφιακό pin 8 του Arduino και ανάμεσα τους είναι συνδεδεμένος σε σειρά ένας αντιστάτης 1,8ΚΩ.
- Ο ακροδέκτης K, συνδέεται στη γείωση του Arduino.
- Ο ακροδέκτης S1, συνδέεται στο μεσαίο pin του ποτενσιόμετρου 50ΚΩ.
- Ο ακροδέκτης S, συνδέεται στο pin V_{adj} του KA278RA05C.

2.3 Η οθόνη LCD

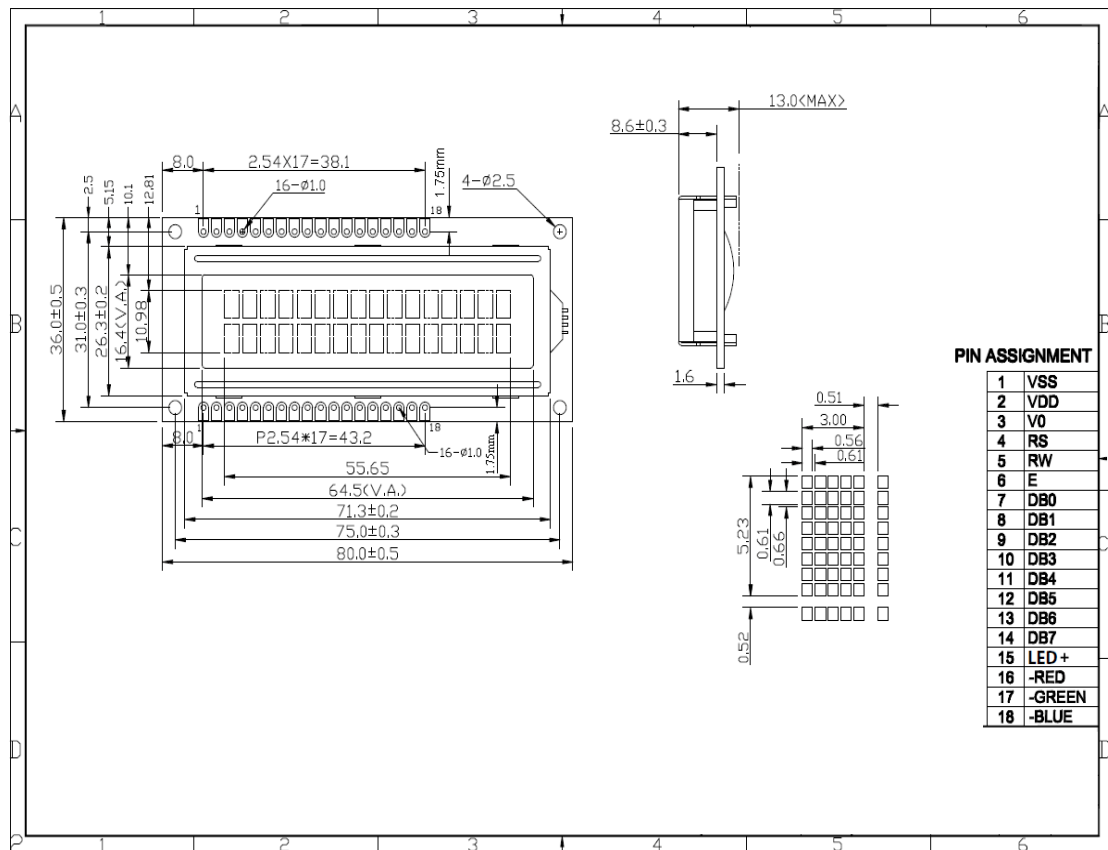
Ένα χαρακτηριστικό που θεωρήθηκε απαραίτητο για την υλοποίηση της εργασίας είναι η αυτονομία της ηλεκτρονικής κατασκευής. Για αυτό το λόγο, χρησιμοποιήθηκε μια LCD οθόνη για την προβολή μηνυμάτων και τιμών ώστε να μην χρειάζεται ένας ηλεκτρονικός υπολογιστής για την προβολή τους. Μια 16x2 LCD είναι μια πολύ συνηθισμένη οθόνη και η χρήση της είναι πολύ συνήθης σε διάφορες συσκευές και κυκλώματα. Οι συγκεκριμένες οθόνες προτιμώνται από άλλες οθόνες διότι είναι οικονομικές, προγραμματίζονται εύκολα και δεν έχουν περιορισμούς στην προβολή χαρακτήρων καθώς μπορούν να προβληθούν και custom χαρακτήρες. Η οθόνη που χρησιμοποιήθηκε είναι μια 16x2 LCD της Adafruit με RGB Backlight negative. Αυτό σημαίνει ότι το χρώμα των χαρακτήρων της οθόνης μπορεί να αλλάξει σε οποιοδήποτε επιθυμητό. Χρησιμοποιεί το chipset HD44780 το οποίο είναι αρκετά συνηθισμένο για οδήγηση οθόνων. Η οθόνη έχει 18 ακροδέκτες σε αντίθεση με τις απλές 16x2 LCD οι οποίες έχουν 16. Τα δυο πρόσθετα pin (17 και 18) είναι για το πράσινο και το μπλε led αντίστοιχα. Υπάρχει ένα RGB led backlight για όλη την οθόνη.



Εικόνα 2.14: Adafruit LCD RGB 16x2.

Τεχνικά χαρακτηριστικά:

- 16 χαρακτήρες ανά γραμμή, 2 γραμμές
- Πολλαπλή επιλογή χρώματος χαρακτήρων σε σκούρο φόντο
- Τα R, G και B led έχουν το καθένα από έναν αντιστάτη 200Ω συνδεδεμένο σε σειρά και έτσι οι ακροδέκτες του μπορούν να συνδεθούν απευθείας σε μια πηγή 3 ή 5V.
- Το ενσωματωμένο σετ χαρακτήρων υποστηρίζει τους χαρακτήρες του αγγλικού και ιαπωνικού αλφαβήτου.
- Μπορούν να σχηματιστούν μέχρι και 8 επιπλέον χαρακτήρες για τη δημιουργία γραμμάτων άλλης γλώσσας καθώς και τη δημιουργία συμβόλων.



Εικόνα 2.15: Σχηματικό της οθόνης LCD.

Περιγραφή των ακροδεκτών:

1. – VSS: Η γείωση της τροφοδοσίας της οθόνης.
2. – VDD: +5V της τροφοδοσίας της οθόνης.
3. – V0: Ρύθμιση της αντίθεσης της οθόνης μέσω ενός ποτενσιόμετρου 10K.
4. – RS: Όταν βρίσκεται σε κατάσταση LOW είναι επιλεγμένος ο καταχωρητής εντολών ώστε να σταλεί στην οθόνη μια εντολή και όταν είναι σε HIGH είναι επιλεγμένος ο καταχωρητής δεδομένων ώστε να σταλούν δεδομένα για να προβληθούν στην οθόνη.
5. – RW: Εγγραφή στην οθόνη όταν βρίσκεται σε κατάσταση LOW, διάβασμα απ την οθόνη όταν βρίσκεται σε κατάσταση HIGH.
6. – E: Ενεργοποιεί τη διαδικασία εγγραφής ώστε να προβληθούν δεδομένα στην οθόνη.

7 - 14. – DB0 - DB7: Είναι οι 8-bit ακροδέκτες δεδομένων οι οποίοι λαμβάνουν τα δεδομένα ώστε να προβληθούν στην οθόνη

15. – LED+: Η άνοδος του RGB backlight led.

16. – -RED: Η κάθοδος του κόκκινου led από το RGB backlight led.

17. – -GREEN: Η κάθοδος του πράσινου led από το RGB backlight led.

18. – -BLUE: Η κάθοδος του μπλε led από το RGB backlight led.

2.3.1 Σύνδεση της οθόνης μέσω του πρωτοκόλλου I2C.

Η LCD οθόνη χρησιμοποιεί 7 ψηφιακούς ακροδέκτες του Arduino. Για εξοικονόμηση των pin όμως, η οθόνη συνδέθηκε έτσι ώστε να επικοινωνεί μέσω του πρωτοκόλλου I2C. Για την υλοποίηση αυτής της ενεργείας χρησιμοποίησα το ολοκληρωμένο PCF8574A το οποίο είναι ένας I2C Expander. Οι ακροδέκτες της οθόνης που θα συνδεόντουσαν στα ψηφιακά pin του Arduino συνδέονται πλέον στο PCF8574A και από αυτό συνδέονται μόλις δυο ακροδέκτες στο Arduino και συγκεκριμένα στον δίαυλο επικοινωνίας I2C, δηλαδή στους αναλογικούς ακροδέκτες A4 και A5 (SDA και SCL αντίστοιχα). Οπότε, εξοικονομήθηκαν 5 ψηφιακοί ακροδέκτες του Arduino. Επίσης, επειδή η LCD οθόνη έχει RGB backlight led, οι κάθοδοι του κόκκινου, του πράσινου και του μπλε led είναι συνδεδεμένοι στα ψηφιακά PWM pin 6 , 5 και 3 αντίστοιχα ώστε με την κατάλληλη τάση στην καθεμία να δημιουργηθεί συγκεκριμένο συνδυασμό χρώματος.

Η συνδεσμολογία με την οποία η οθόνη συνδέεται στο Arduino μέσω του PCF8574A αναφέρεται παρακάτω:

- Lcd Pin 1 (VSS) → Arduino Ground
- Lcd Pin 2 (VDD) → Arduino +5V
- Lcd Pin 3 (V0) → Middle Pin of Potentiometer 10K
- Lcd Pin 4 (RS) → PCF8574A Pin 11
- Lcd Pin 5 (RW) → PCF8574A Pin 10

- Lcd Pin 6 (E) → PCF8574A Pin 9
- Lcd Pin 7-10 (DB0-DB3) → N/C Lcd Pin 11 (DB4) → PCF8574A Pin 4
- Lcd Pin 12 (DB5) → PCF8574A Pin 5
- Lcd Pin 13 (DB6) → PCF8574A Pin 6
- Lcd Pin 14 (DB7) → PCF8574A Pin 7
- Lcd Pin 15 (LED+) → Arduino +5V
- Lcd Pin 17 (-GREEN) → Arduino Pin 5 (PWM)
- Lcd Pin 17 (-GREEN) → Arduino Pin 5 (PWM)
- Lcd Pin 18 (-BLUE) → Arduino Pin 3 (PWM)
- Οι άλλοι δυο ακροδέκτες του ποτενσιόμετρου συνδέονται στο Arduino Ground και Arduino +5V.

Στη συνέχεια αναφέρεται παρακάτω η συνδεσμολογία με την οποία συνδέεται το PCF8574A στο Arduino:

- Pin 1 (A0) → Arduino Ground
- Pin 2 (A1) → Arduino Ground
- Pin 3 (A2) → Arduino Ground
- Pin 4 (P0) → Lcd Pin 11
- Pin 5 (P1) → Lcd Pin 12
- Pin 6 (P2) → Lcd Pin 13
- Pin 7 (P3) → Lcd Pin 14
- Pin 8 (VSS) → Arduino Ground
- Pin 9 (P4) → Lcd Pin 6
- Pin 10 (P5) → Lcd Pin 5

- Pin 11 (P6) → Lcd Pin 4
- Pin 12-13 → N/C
- Pin 14 (SCL) → Arduino Pin A5
- Pin 15 (SDA) → Arduino Pin A4
- Pin 16 (VDD) → Arduino +5V

ΠΑΡΑΡΤΗΜΑ Α΄

Στο παράρτημα αυτό παρατίθεται ένα δείγμα μετρήσεων που λήφθηκαν από το κύκλωμα.

ID	Mode	CO (ppm)	Time	Date
1	Cleaning Sensing Elements	509	21:40:00	17/6/2015
2	Cleaning Sensing Elements	315	21:40:10	17/6/2015
3	Cleaning Sensing Elements	248	21:40:20	17/6/2015
4	Cleaning Sensing Elements	209	21:40:30	17/6/2015
5	Cleaning Sensing Elements	187	21:40:40	17/6/2015
6	Taking Readings	164	21:40:50	17/6/2015
7	Taking Readings	55	21:41:00	17/6/2015
8	Taking Readings	26	21:41:10	17/6/2015
9	Taking Readings	19	21:41:20	17/6/2015
10	Taking Readings	18	21:41:30	17/6/2015
11	Taking Readings	20	21:41:40	17/6/2015
12	Taking Readings	20	21:41:50	17/6/2015
13	Taking Readings	22	21:42:00	17/6/2015
14	Taking Readings	24	21:42:10	17/6/2015
15	Cleaning Sensing Elements	42	21:42:20	17/6/2015
16	Cleaning Sensing Elements	347	21:42:30	17/6/2015
17	Cleaning Sensing Elements	226	21:42:40	17/6/2015
18	Cleaning Sensing Elements	186	21:42:50	17/6/2015
19	Cleaning Sensing Elements	169	21:43:00	17/6/2015
20	Cleaning Sensing Elements	160	21:43:10	17/6/2015
21	Taking Readings	147	21:43:20	17/6/2015
22	Taking Readings	47	21:43:30	17/6/2015
23	Taking Readings	21	21:43:40	17/6/2015
24	Taking Readings	17	21:43:50	17/6/2015
25	Taking Readings	17	21:44:00	17/6/2015
26	Taking Readings	18	21:44:10	17/6/2015
27	Taking Readings	18	21:44:20	17/6/2015
28	Taking Readings	19	21:44:30	17/6/2015
29	Taking Readings	22	21:44:40	17/6/2015
30	Cleaning Sensing Elements	31	21:44:50	17/6/2015
31	Cleaning Sensing Elements	338	21:45:00	17/6/2015
32	Cleaning Sensing Elements	228	21:45:10	17/6/2015
33	Cleaning Sensing Elements	189	21:45:20	17/6/2015
34	Cleaning Sensing Elements	171	21:45:30	17/6/2015
35	Cleaning Sensing Elements	163	21:45:40	17/6/2015
36	Taking Readings	153	21:45:50	17/6/2015
37	Taking Readings	38	21:46:00	17/6/2015
38	Taking Readings	15	21:46:10	17/6/2015

39	Taking Readings	526	21:46:20	17/6/2015
40	Taking Readings	795	21:46:30	17/6/2015
41	Taking Readings	876	21:46:40	17/6/2015
42	Taking Readings	895	21:46:50	17/6/2015
43	Taking Readings	927	21:47:00	17/6/2015
44	Taking Readings	930	21:47:10	17/6/2015
45	Cleaning Sensing Elements	925	21:47:20	17/6/2015
46	Cleaning Sensing Elements	867	21:47:30	17/6/2015
47	Cleaning Sensing Elements	705	21:47:40	17/6/2015
48	Cleaning Sensing Elements	578	21:47:50	17/6/2015
49	Cleaning Sensing Elements	490	21:48:00	17/6/2015
50	Cleaning Sensing Elements	420	21:48:10	17/6/2015
51	Taking Readings	364	21:48:20	17/6/2015
52	Taking Readings	181	21:48:30	17/6/2015
53	Taking Readings	101	21:48:40	17/6/2015
54	Taking Readings	87	21:48:50	17/6/2015
55	Taking Readings	85	21:49:00	17/6/2015
56	Taking Readings	88	21:49:10	17/6/2015
57	Taking Readings	102	21:49:20	17/6/2015
58	Taking Readings	108	21:49:30	17/6/2015
59	Taking Readings	112	21:49:40	17/6/2015
60	Taking Readings	117	21:49:50	17/6/2015
61	Cleaning Sensing Elements	552	21:50:00	17/6/2015
62	Cleaning Sensing Elements	391	21:50:10	17/6/2015
63	Cleaning Sensing Elements	329	21:50:20	17/6/2015
64	Cleaning Sensing Elements	289	21:50:30	17/6/2015
65	Cleaning Sensing Elements	270	21:50:40	17/6/2015
66	Cleaning Sensing Elements	246	21:50:50	17/6/2015
67	Taking Readings	96	21:51:00	17/6/2015
68	Taking Readings	47	21:51:10	17/6/2015
69	Taking Readings	35	21:51:20	17/6/2015
70	Taking Readings	36	21:51:30	17/6/2015
71	Taking Readings	38	21:51:40	17/6/2015
72	Taking Readings	40	21:51:50	17/6/2015
73	Taking Readings	42	21:52:00	17/6/2015
74	Taking Readings	45	21:52:10	17/6/2015
75	Taking Readings	47	21:52:20	17/6/2015
76	Cleaning Sensing Elements	439	21:52:30	17/6/2015
77	Cleaning Sensing Elements	298	21:52:40	17/6/2015
78	Cleaning Sensing Elements	250	21:52:50	17/6/2015
79	Cleaning Sensing Elements	225	21:53:00	17/6/2015
80	Cleaning Sensing Elements	210	21:53:10	17/6/2015
81	Cleaning Sensing Elements	200	21:53:20	17/6/2015
82	Taking Readings	64	21:53:30	17/6/2015
83	Taking Readings	28	21:53:40	17/6/2015

84	Taking Readings	19	21:53:50	17/6/2015
85	Taking Readings	19	21:54:00	17/6/2015
86	Taking Readings	20	21:54:10	17/6/2015
87	Taking Readings	22	21:54:20	17/6/2015
88	Taking Readings	24	21:54:30	17/6/2015
89	Taking Readings	26	21:54:40	17/6/2015
90	Taking Readings	27	21:54:50	17/6/2015
91	Cleaning Sensing Elements	382	21:55:00	17/6/2015
92	Cleaning Sensing Elements	259	21:55:10	17/6/2015
93	Cleaning Sensing Elements	216	21:55:20	17/6/2015
94	Cleaning Sensing Elements	198	21:55:30	17/6/2015
95	Cleaning Sensing Elements	187	21:55:40	17/6/2015
96	Cleaning Sensing Elements	179	21:55:50	17/6/2015
97	Taking Readings	55	21:56:00	17/6/2015
98	Taking Readings	21	21:56:10	17/6/2015
99	Taking Readings	15	21:56:20	17/6/2015
100	Taking Readings	15	21:56:30	17/6/2015
101	Taking Readings	16	21:56:40	17/6/2015
102	Taking Readings	17	21:56:50	17/6/2015
103	Taking Readings	18	21:57:00	17/6/2015
104	Taking Readings	19	21:57:10	17/6/2015
105	Taking Readings	20	21:57:20	17/6/2015
106	Cleaning Sensing Elements	350	21:57:30	17/6/2015
107	Cleaning Sensing Elements	240	21:57:40	17/6/2015
108	Cleaning Sensing Elements	200	21:57:50	17/6/2015
109	Cleaning Sensing Elements	184	21:58:00	17/6/2015
110	Cleaning Sensing Elements	174	21:58:10	17/6/2015
111	Cleaning Sensing Elements	167	21:58:20	17/6/2015
112	Taking Readings	49	21:58:30	17/6/2015
113	Taking Readings	17	21:58:40	17/6/2015
114	Taking Readings	12	21:58:50	17/6/2015
115	Taking Readings	12	21:59:00	17/6/2015
116	Taking Readings	14	21:59:10	17/6/2015
117	Taking Readings	14	21:59:20	17/6/2015
118	Taking Readings	15	21:59:30	17/6/2015
119	Taking Readings	16	21:59:40	17/6/2015
120	Taking Readings	17	21:59:50	17/6/2015
121	Cleaning Sensing Elements	336	22:00:00	17/6/2015
122	Cleaning Sensing Elements	229	22:00:10	17/6/2015
123	Cleaning Sensing Elements	189	22:00:20	17/6/2015
124	Cleaning Sensing Elements	172	22:00:30	17/6/2015
125	Cleaning Sensing Elements	165	22:00:40	17/6/2015
126	Cleaning Sensing Elements	159	22:00:50	17/6/2015
127	Taking Readings	47	22:01:00	17/6/2015
128	Taking Readings	16	22:01:10	17/6/2015

129	Taking Readings	11	22:01:20	17/6/2015
130	Taking Readings	10	22:01:30	17/6/2015
131	Taking Readings	11	22:01:40	17/6/2015
132	Taking Readings	11	22:01:50	17/6/2015
133	Taking Readings	13	22:02:00	17/6/2015
134	Taking Readings	13	22:02:10	17/6/2015
135	Taking Readings	15	22:02:20	17/6/2015
136	Cleaning Sensing Elements	318	22:02:30	17/6/2015
137	Cleaning Sensing Elements	220	22:02:40	17/6/2015
138	Cleaning Sensing Elements	183	22:02:50	17/6/2015
139	Cleaning Sensing Elements	167	22:03:00	17/6/2015
140	Cleaning Sensing Elements	159	22:03:10	17/6/2015
141	Cleaning Sensing Elements	154	22:03:20	17/6/2015
142	Taking Readings	46	22:03:30	17/6/2015
143	Taking Readings	13	22:03:40	17/6/2015
144	Taking Readings	9	22:03:50	17/6/2015
145	Taking Readings	9	22:04:00	17/6/2015
146	Taking Readings	10	22:04:10	17/6/2015
147	Taking Readings	10	22:04:20	17/6/2015
148	Taking Readings	11	22:04:30	17/6/2015
149	Taking Readings	13	22:04:40	17/6/2015
150	Taking Readings	14	22:04:50	17/6/2015
151	Cleaning Sensing Elements	308	22:05:00	17/6/2015
152	Cleaning Sensing Elements	215	22:05:10	17/6/2015
153	Cleaning Sensing Elements	176	22:05:20	17/6/2015
154	Cleaning Sensing Elements	163	22:05:30	17/6/2015
155	Cleaning Sensing Elements	155	22:05:40	17/6/2015
156	Cleaning Sensing Elements	149	22:05:50	17/6/2015
157	Taking Readings	45	22:06:00	17/6/2015
158	Taking Readings	13	22:06:10	17/6/2015
159	Taking Readings	9	22:06:20	17/6/2015
160	Taking Readings	9	22:06:30	17/6/2015
161	Taking Readings	9	22:06:40	17/6/2015
162	Taking Readings	10	22:06:50	17/6/2015
163	Taking Readings	10	22:07:00	17/6/2015
164	Taking Readings	12	22:07:10	17/6/2015
165	Taking Readings	12	22:07:20	17/6/2015
166	Cleaning Sensing Elements	301	22:07:30	17/6/2015
167	Cleaning Sensing Elements	210	22:07:40	17/6/2015
168	Cleaning Sensing Elements	173	22:07:50	17/6/2015
169	Cleaning Sensing Elements	159	22:08:00	17/6/2015
170	Cleaning Sensing Elements	152	22:08:10	17/6/2015
171	Cleaning Sensing Elements	148	22:08:20	17/6/2015
172	Taking Readings	44	22:08:30	17/6/2015
173	Taking Readings	14	22:08:40	17/6/2015

174	Taking Readings	8	22:08:50	17/6/2015
175	Taking Readings	8	22:09:00	17/6/2015
176	Taking Readings	8	22:09:10	17/6/2015
177	Taking Readings	10	22:09:20	17/6/2015
178	Taking Readings	11	22:09:30	17/6/2015
179	Taking Readings	12	22:09:40	17/6/2015
180	Taking Readings	11	22:09:50	17/6/2015
181	Cleaning Sensing Elements	297	22:10:00	17/6/2015
182	Cleaning Sensing Elements	209	22:10:10	17/6/2015
183	Cleaning Sensing Elements	172	22:10:20	17/6/2015
184	Cleaning Sensing Elements	157	22:10:30	17/6/2015
185	Cleaning Sensing Elements	150	22:10:40	17/6/2015
186	Cleaning Sensing Elements	145	22:10:50	17/6/2015
187	Taking Readings	45	22:11:00	17/6/2015
188	Taking Readings	13	22:11:10	17/6/2015
189	Taking Readings	8	22:11:20	17/6/2015
190	Taking Readings	8	22:11:30	17/6/2015
191	Taking Readings	9	22:11:40	17/6/2015
192	Taking Readings	10	22:11:50	17/6/2015
193	Taking Readings	10	22:12:00	17/6/2015
194	Taking Readings	11	22:12:10	17/6/2015
195	Taking Readings	12	22:12:20	17/6/2015
196	Cleaning Sensing Elements	294	22:12:30	17/6/2015
197	Cleaning Sensing Elements	210	22:12:40	17/6/2015
198	Cleaning Sensing Elements	171	22:12:50	17/6/2015
199	Cleaning Sensing Elements	156	22:13:00	17/6/2015
200	Cleaning Sensing Elements	147	22:13:10	17/6/2015
201	Cleaning Sensing Elements	144	22:13:20	17/6/2015
202	Taking Readings	45	22:13:30	17/6/2015
203	Taking Readings	14	22:13:40	17/6/2015
204	Taking Readings	8	22:13:50	17/6/2015
205	Taking Readings	8	22:14:00	17/6/2015
206	Taking Readings	8	22:14:10	17/6/2015
207	Taking Readings	8	22:14:20	17/6/2015
208	Taking Readings	9	22:14:30	17/6/2015
209	Taking Readings	11	22:14:40	17/6/2015
210	Taking Readings	12	22:14:50	17/6/2015
211	Cleaning Sensing Elements	284	22:15:00	17/6/2015
212	Cleaning Sensing Elements	206	22:15:10	17/6/2015
213	Cleaning Sensing Elements	168	22:15:20	17/6/2015
214	Cleaning Sensing Elements	154	22:15:30	17/6/2015
215	Cleaning Sensing Elements	147	22:15:40	17/6/2015
216	Cleaning Sensing Elements	141	22:15:50	17/6/2015
217	Taking Readings	47	22:16:00	17/6/2015
218	Taking Readings	13	22:16:10	17/6/2015

219	Taking Readings	8	22:16:20	17/6/2015
220	Taking Readings	6	22:16:30	17/6/2015
221	Taking Readings	8	22:16:40	17/6/2015
222	Taking Readings	8	22:16:50	17/6/2015
223	Taking Readings	9	22:17:00	17/6/2015
224	Taking Readings	10	22:17:10	17/6/2015
225	Taking Readings	11	22:17:20	17/6/2015
226	Cleaning Sensing Elements	276	22:17:30	17/6/2015
227	Cleaning Sensing Elements	206	22:17:40	17/6/2015
228	Cleaning Sensing Elements	167	22:17:50	17/6/2015
229	Cleaning Sensing Elements	152	22:18:00	17/6/2015
230	Cleaning Sensing Elements	146	22:18:10	17/6/2015
231	Cleaning Sensing Elements	141	22:18:20	17/6/2015
232	Taking Readings	47	22:18:30	17/6/2015
233	Taking Readings	13	22:18:40	17/6/2015
234	Taking Readings	8	22:18:50	17/6/2015
235	Taking Readings	6	22:19:00	17/6/2015
236	Taking Readings	7	22:19:10	17/6/2015
237	Taking Readings	8	22:19:20	17/6/2015
238	Taking Readings	9	22:19:30	17/6/2015
239	Taking Readings	9	22:19:40	17/6/2015
240	Taking Readings	9	22:19:50	17/6/2015
241	Cleaning Sensing Elements	264	22:20:00	17/6/2015
242	Cleaning Sensing Elements	206	22:20:10	17/6/2015
243	Cleaning Sensing Elements	167	22:20:20	17/6/2015
244	Cleaning Sensing Elements	151	22:20:30	17/6/2015
245	Cleaning Sensing Elements	144	22:20:40	17/6/2015
246	Cleaning Sensing Elements	140	22:20:50	17/6/2015
247	Taking Readings	48	22:21:00	17/6/2015
248	Taking Readings	13	22:21:10	17/6/2015
249	Taking Readings	8	22:21:20	17/6/2015
250	Taking Readings	7	22:21:30	17/6/2015
251	Taking Readings	7	22:21:40	17/6/2015
252	Taking Readings	8	22:21:50	17/6/2015
253	Taking Readings	9	22:22:00	17/6/2015
254	Taking Readings	10	22:22:10	17/6/2015
255	Taking Readings	9	22:22:20	17/6/2015
256	Cleaning Sensing Elements	259	22:22:30	17/6/2015
257	Cleaning Sensing Elements	201	22:22:40	17/6/2015
258	Cleaning Sensing Elements	162	22:22:50	17/6/2015
259	Cleaning Sensing Elements	148	22:23:00	17/6/2015
260	Cleaning Sensing Elements	139	22:23:10	17/6/2015
261	Cleaning Sensing Elements	137	22:23:20	17/6/2015
262	Taking Readings	47	22:23:30	17/6/2015
263	Taking Readings	12	22:23:40	17/6/2015

264	Taking Readings	8	22:23:50	17/6/2015
265	Taking Readings	6	22:24:00	17/6/2015
266	Taking Readings	5	22:24:10	17/6/2015
267	Taking Readings	7	22:24:20	17/6/2015
268	Taking Readings	7	22:24:30	17/6/2015
269	Taking Readings	8	22:24:40	17/6/2015
270	Taking Readings	9	22:24:50	17/6/2015
271	Cleaning Sensing Elements	242	22:25:00	17/6/2015
272	Cleaning Sensing Elements	201	22:25:10	17/6/2015
273	Cleaning Sensing Elements	161	22:25:20	17/6/2015
274	Cleaning Sensing Elements	147	22:25:30	17/6/2015
275	Cleaning Sensing Elements	140	22:25:40	17/6/2015
276	Cleaning Sensing Elements	135	22:25:50	17/6/2015
277	Taking Readings	48	22:26:00	17/6/2015
278	Taking Readings	11	22:26:10	17/6/2015
279	Taking Readings	6	22:26:20	17/6/2015
280	Taking Readings	6	22:26:30	17/6/2015
281	Taking Readings	6	22:26:40	17/6/2015
282	Taking Readings	6	22:26:50	17/6/2015
283	Taking Readings	6	22:27:00	17/6/2015
284	Taking Readings	7	22:27:10	17/6/2015
285	Taking Readings	8	22:27:20	17/6/2015
286	Cleaning Sensing Elements	233	22:27:30	17/6/2015
287	Cleaning Sensing Elements	203	22:27:40	17/6/2015
288	Cleaning Sensing Elements	164	22:27:50	17/6/2015
289	Cleaning Sensing Elements	149	22:28:00	17/6/2015
290	Cleaning Sensing Elements	141	22:28:10	17/6/2015
291	Cleaning Sensing Elements	136	22:28:20	17/6/2015
292	Taking Readings	51	22:28:30	17/6/2015
293	Taking Readings	11	22:28:40	17/6/2015
294	Taking Readings	7	22:28:50	17/6/2015
295	Taking Readings	5	22:29:00	17/6/2015
296	Taking Readings	6	22:29:10	17/6/2015
297	Taking Readings	7	22:29:20	17/6/2015
298	Taking Readings	8	22:29:30	17/6/2015
299	Taking Readings	8	22:29:40	17/6/2015
300	Taking Readings	9	22:29:50	17/6/2015
301	Cleaning Sensing Elements	227	22:30:00	17/6/2015
302	Cleaning Sensing Elements	208	22:30:10	17/6/2015
303	Cleaning Sensing Elements	165	22:30:20	17/6/2015
304	Cleaning Sensing Elements	151	22:30:30	17/6/2015
305	Cleaning Sensing Elements	143	22:30:40	17/6/2015
306	Cleaning Sensing Elements	137	22:30:50	17/6/2015
307	Taking Readings	55	22:31:00	17/6/2015
308	Taking Readings	13	22:31:10	17/6/2015

309	Taking Readings	6	22:31:20	17/6/2015
310	Taking Readings	6	22:31:30	17/6/2015
311	Taking Readings	7	22:31:40	17/6/2015
312	Taking Readings	7	22:31:50	17/6/2015
313	Taking Readings	8	22:32:00	17/6/2015
314	Taking Readings	10	22:32:10	17/6/2015
315	Taking Readings	8	22:32:20	17/6/2015
316	Cleaning Sensing Elements	226	22:32:30	17/6/2015
317	Cleaning Sensing Elements	208	22:32:40	17/6/2015
318	Cleaning Sensing Elements	165	22:32:50	17/6/2015
319	Cleaning Sensing Elements	148	22:33:00	17/6/2015
320	Cleaning Sensing Elements	141	22:33:10	17/6/2015
321	Cleaning Sensing Elements	137	22:33:20	17/6/2015
322	Taking Readings	56	22:33:30	17/6/2015
323	Taking Readings	12	22:33:40	17/6/2015
324	Taking Readings	7	22:33:50	17/6/2015
325	Taking Readings	6	22:34:00	17/6/2015
326	Taking Readings	6	22:34:10	17/6/2015
327	Taking Readings	7	22:34:20	17/6/2015
328	Taking Readings	7	22:34:30	17/6/2015
329	Taking Readings	8	22:34:40	17/6/2015
330	Taking Readings	8	22:34:50	17/6/2015
331	Cleaning Sensing Elements	207	22:35:00	17/6/2015
332	Cleaning Sensing Elements	209	22:35:10	17/6/2015
333	Cleaning Sensing Elements	166	22:35:20	17/6/2015
334	Cleaning Sensing Elements	150	22:35:30	17/6/2015
335	Cleaning Sensing Elements	143	22:35:40	17/6/2015
336	Cleaning Sensing Elements	139	22:35:50	17/6/2015
337	Taking Readings	60	22:36:00	17/6/2015
338	Taking Readings	13	22:36:10	17/6/2015
339	Taking Readings	6	22:36:20	17/6/2015
340	Taking Readings	6	22:36:30	17/6/2015
341	Taking Readings	6	22:36:40	17/6/2015
342	Taking Readings	7	22:36:50	17/6/2015
343	Taking Readings	8	22:37:00	17/6/2015
344	Taking Readings	8	22:37:10	17/6/2015
345	Taking Readings	7	22:37:20	17/6/2015
346	Cleaning Sensing Elements	193	22:37:30	17/6/2015
347	Cleaning Sensing Elements	212	22:37:40	17/6/2015
348	Cleaning Sensing Elements	166	22:37:50	17/6/2015
349	Cleaning Sensing Elements	150	22:38:00	17/6/2015
350	Cleaning Sensing Elements	141	22:38:10	17/6/2015
351	Cleaning Sensing Elements	137	22:38:20	17/6/2015
352	Taking Readings	61	22:38:30	17/6/2015
353	Taking Readings	13	22:38:40	17/6/2015

354	Taking Readings	7	22:38:50	17/6/2015
355	Taking Readings	5	22:39:00	17/6/2015
356	Taking Readings	6	22:39:10	17/6/2015
357	Taking Readings	6	22:39:20	17/6/2015
358	Taking Readings	6	22:39:30	17/6/2015
359	Taking Readings	8	22:39:40	17/6/2015
360	Taking Readings	8	22:39:50	17/6/2015
361	Cleaning Sensing Elements	173	22:40:00	17/6/2015
362	Cleaning Sensing Elements	215	22:40:10	17/6/2015
363	Cleaning Sensing Elements	168	22:40:20	17/6/2015
364	Cleaning Sensing Elements	152	22:40:30	17/6/2015
365	Cleaning Sensing Elements	143	22:40:40	17/6/2015
366	Cleaning Sensing Elements	138	22:40:50	17/6/2015
367	Taking Readings	68	22:41:00	17/6/2015
368	Taking Readings	15	22:41:10	17/6/2015
369	Taking Readings	7	22:41:20	17/6/2015
370	Taking Readings	5	22:41:30	17/6/2015
371	Taking Readings	5	22:41:40	17/6/2015
372	Taking Readings	7	22:41:50	17/6/2015
373	Taking Readings	6	22:42:00	17/6/2015
374	Taking Readings	9	22:42:10	17/6/2015
375	Taking Readings	8	22:42:20	17/6/2015
376	Cleaning Sensing Elements	158	22:42:30	17/6/2015
377	Cleaning Sensing Elements	219	22:42:40	17/6/2015
378	Cleaning Sensing Elements	169	22:42:50	17/6/2015
379	Cleaning Sensing Elements	150	22:43:00	17/6/2015
380	Cleaning Sensing Elements	142	22:43:10	17/6/2015
381	Cleaning Sensing Elements	137	22:43:20	17/6/2015
382	Taking Readings	70	22:43:30	17/6/2015
383	Taking Readings	14	22:43:40	17/6/2015
384	Taking Readings	6	22:43:50	17/6/2015
385	Taking Readings	6	22:44:00	17/6/2015
386	Taking Readings	6	22:44:10	17/6/2015
387	Taking Readings	6	22:44:20	17/6/2015
388	Taking Readings	6	22:44:30	17/6/2015
389	Taking Readings	8	22:44:40	17/6/2015
390	Taking Readings	7	22:44:50	17/6/2015
391	Cleaning Sensing Elements	140	22:45:00	17/6/2015
392	Cleaning Sensing Elements	219	22:45:10	17/6/2015
393	Cleaning Sensing Elements	167	22:45:20	17/6/2015
394	Cleaning Sensing Elements	150	22:45:30	17/6/2015
395	Cleaning Sensing Elements	141	22:45:40	17/6/2015
396	Cleaning Sensing Elements	137	22:45:50	17/6/2015
397	Taking Readings	74	22:46:00	17/6/2015
398	Taking Readings	15	22:46:10	17/6/2015

399	Taking Readings	6	22:46:20	17/6/2015
400	Taking Readings	6	22:46:30	17/6/2015
401	Taking Readings	5	22:46:40	17/6/2015
402	Taking Readings	6	22:46:50	17/6/2015
403	Taking Readings	7	22:47:00	17/6/2015
404	Taking Readings	6	22:47:10	17/6/2015
405	Taking Readings	9	22:47:20	17/6/2015
406	Cleaning Sensing Elements	126	22:47:30	17/6/2015
407	Cleaning Sensing Elements	221	22:47:40	17/6/2015
408	Cleaning Sensing Elements	167	22:47:50	17/6/2015
409	Cleaning Sensing Elements	148	22:48:00	17/6/2015
410	Cleaning Sensing Elements	141	22:48:10	17/6/2015
411	Cleaning Sensing Elements	135	22:48:20	17/6/2015
412	Taking Readings	76	22:48:30	17/6/2015
413	Taking Readings	14	22:48:40	17/6/2015
414	Taking Readings	6	22:48:50	17/6/2015
415	Taking Readings	4	22:49:00	17/6/2015
416	Taking Readings	4	22:49:10	17/6/2015
417	Taking Readings	6	22:49:20	17/6/2015
418	Taking Readings	7	22:49:30	17/6/2015
419	Taking Readings	7	22:49:40	17/6/2015
420	Taking Readings	7	22:49:50	17/6/2015
421	Cleaning Sensing Elements	107	22:50:00	17/6/2015
422	Cleaning Sensing Elements	218	22:50:10	17/6/2015
423	Cleaning Sensing Elements	166	22:50:20	17/6/2015
424	Cleaning Sensing Elements	148	22:50:30	17/6/2015
425	Cleaning Sensing Elements	140	22:50:40	17/6/2015
426	Cleaning Sensing Elements	137	22:50:50	17/6/2015
427	Taking Readings	80	22:51:00	17/6/2015
428	Taking Readings	16	22:51:10	17/6/2015
429	Taking Readings	6	22:51:20	17/6/2015
430	Taking Readings	5	22:51:30	17/6/2015
431	Taking Readings	5	22:51:40	17/6/2015
432	Taking Readings	5	22:51:50	17/6/2015
433	Taking Readings	7	22:52:00	17/6/2015
434	Taking Readings	7	22:52:10	17/6/2015
435	Taking Readings	6	22:52:20	17/6/2015
436	Cleaning Sensing Elements	96	22:52:30	17/6/2015
437	Cleaning Sensing Elements	225	22:52:40	17/6/2015
438	Cleaning Sensing Elements	170	22:52:50	17/6/2015
439	Cleaning Sensing Elements	150	22:53:00	17/6/2015
440	Cleaning Sensing Elements	140	22:53:10	17/6/2015
441	Cleaning Sensing Elements	134	22:53:20	17/6/2015
442	Taking Readings	87	22:53:30	17/6/2015
443	Taking Readings	16	22:53:40	17/6/2015

444	Taking Readings	6	22:53:50	17/6/2015
445	Taking Readings	5	22:54:00	17/6/2015
446	Taking Readings	4	22:54:10	17/6/2015
447	Taking Readings	5	22:54:20	17/6/2015
448	Taking Readings	5	22:54:30	17/6/2015
449	Taking Readings	6	22:54:40	17/6/2015
450	Taking Readings	7	22:54:50	17/6/2015
451	Cleaning Sensing Elements	84	22:55:00	17/6/2015
452	Cleaning Sensing Elements	228	22:55:10	17/6/2015
453	Cleaning Sensing Elements	171	22:55:20	17/6/2015
454	Cleaning Sensing Elements	150	22:55:30	17/6/2015
455	Cleaning Sensing Elements	141	22:55:40	17/6/2015
456	Cleaning Sensing Elements	137	22:55:50	17/6/2015
457	Taking Readings	90	22:56:00	17/6/2015
458	Taking Readings	16	22:56:10	17/6/2015
459	Taking Readings	7	22:56:20	17/6/2015
460	Taking Readings	5	22:56:30	17/6/2015
461	Taking Readings	5	22:56:40	17/6/2015
462	Taking Readings	6	22:56:50	17/6/2015
463	Taking Readings	6	22:57:00	17/6/2015
464	Taking Readings	6	22:57:10	17/6/2015
465	Taking Readings	6	22:57:20	17/6/2015
466	Cleaning Sensing Elements	70	22:57:30	17/6/2015
467	Cleaning Sensing Elements	231	22:57:40	17/6/2015
468	Cleaning Sensing Elements	170	22:57:50	17/6/2015
469	Cleaning Sensing Elements	148	22:58:00	17/6/2015
470	Cleaning Sensing Elements	139	22:58:10	17/6/2015
471	Cleaning Sensing Elements	134	22:58:20	17/6/2015
472	Taking Readings	93	22:58:30	17/6/2015
473	Taking Readings	15	22:58:40	17/6/2015
474	Taking Readings	7	22:58:50	17/6/2015
475	Taking Readings	5	22:59:00	17/6/2015
476	Taking Readings	5	22:59:10	17/6/2015
477	Taking Readings	4	22:59:20	17/6/2015
478	Taking Readings	5	22:59:30	17/6/2015
479	Taking Readings	5	22:59:40	17/6/2015
480	Taking Readings	6	22:59:50	17/6/2015
481	Cleaning Sensing Elements	59	23:00:00	17/6/2015
482	Cleaning Sensing Elements	224	23:00:10	17/6/2015
483	Cleaning Sensing Elements	169	23:00:20	17/6/2015
484	Cleaning Sensing Elements	148	23:00:30	17/6/2015
485	Cleaning Sensing Elements	139	23:00:40	17/6/2015
486	Cleaning Sensing Elements	133	23:00:50	17/6/2015
487	Taking Readings	97	23:01:00	17/6/2015
488	Taking Readings	17	23:01:10	17/6/2015

489	Taking Readings	7	23:01:20	17/6/2015
490	Taking Readings	4	23:01:30	17/6/2015
491	Taking Readings	4	23:01:40	17/6/2015
492	Taking Readings	5	23:01:50	17/6/2015
493	Taking Readings	5	23:02:00	17/6/2015
494	Taking Readings	5	23:02:10	17/6/2015
495	Taking Readings	7	23:02:20	17/6/2015
496	Cleaning Sensing Elements	49	23:02:30	17/6/2015
497	Cleaning Sensing Elements	230	23:02:40	17/6/2015
498	Cleaning Sensing Elements	169	23:02:50	17/6/2015
499	Cleaning Sensing Elements	148	23:03:00	17/6/2015
500	Cleaning Sensing Elements	139	23:03:10	17/6/2015
501	Cleaning Sensing Elements	135	23:03:20	17/6/2015
502	Taking Readings	103	23:03:30	17/6/2015
503	Taking Readings	18	23:03:40	17/6/2015
504	Taking Readings	7	23:03:50	17/6/2015
505	Taking Readings	5	23:04:00	17/6/2015
506	Taking Readings	5	23:04:10	17/6/2015
507	Taking Readings	5	23:04:20	17/6/2015
508	Taking Readings	7	23:04:30	17/6/2015
509	Taking Readings	7	23:04:40	17/6/2015
510	Taking Readings	7	23:04:50	17/6/2015

Όπως αναφέρθηκε και στο 2^ο κεφάλαιο το κύκλωμα του heater του MQ-7 λειτουργεί σε τάση 5V για 60sec και σε τάση 1.4V για 90sec και οι τάσεις εναλλάσσονται κυκλικά για όση χρονική διάρκεια βρίσκεται σε λειτουργία ο αισθητήρας. Όταν το κύκλωμα λειτουργεί σε τάση 5V, γίνεται καθαρισμός των χημικών στοιχείων του αισθητήρα, η στήλη Mode έχει την τιμή Cleaning Sensing Elements και η τιμή στην στήλη CO (ppm) δε λαμβάνεται υπ' όψιν. Όταν το κύκλωμα λειτουργεί σε τάση 1.4V, τότε μόνο ο αισθητήρας μετράει την τιμή του μονοξειδίου του άνθρακα από τον περιβάλλοντα χώρο και μόνο αυτές οι τιμές μας ενδιαφέρουν από τη στήλη CO (ppm). Γι αυτό το λόγο σε αυτή τη λειτουργία στη στήλη Mode καταγράφεται η τιμή Taking Readings. Το κύκλωμα κάνει καταγραφή με συχνότητα 10sec. Τα στοιχεία που καταγράφει είναι το ID της καταγραφής, η λειτουργία του κυκλώματος εκείνη τη στιγμή (Mode) δηλαδή αν ο αισθητήρας παίρνει μετρήσεις (Taking Readings) ή καθαρίζει τα χημικά του στοιχεία (Cleaning Sensing Elements), η ποσότητα του μονοξειδίου του άνθρακα στον

περιβάλλοντα χώρο (CO(ppm)), η ώρα της καταγραφής (Time), και η ημερομηνία (Date).

Οι μετρήσεις λήφθηκαν σε δωμάτιο διαμερίσματος. Παρατηρείται στις μετρήσεις με ID από 36 μέχρι και 44 ότι υπάρχει αύξηση της τιμής του μονοξειδίου του άνθρακα και αυτό εξηγείται στο ότι εκείνο το χρονικό διάστημα υπήρχε μεγάλη ποσότητα καπνού στο χώρο. Επίσης παρατηρήθηκε ότι όσο ο χώρος ήταν κλειστός και δεν αεριζόταν η ποσότητα CO είχε μεγαλύτερη τιμή σε σχέση με τις τιμές ποσότητας CO που καταγράφηκαν όσο το δωμάτιο αεριζόταν από το εξωτερικό περιβάλλον. Ακόμη ένα φαινόμενο που παρατηρήθηκε είναι ότι κατά την αλλαγή της λειτουργίας από τον καθαρισμό των χημικών στοιχείων του αισθητήρα στη λήψη μετρήσεων, η επαναφορά της πραγματικής τιμής του μονοξειδίου του άνθρακα γίνεται σταδιακά και όχι στιγμιαία.

ΠΑΡΑΡΤΗΜΑ Β΄

Στο παράρτημα αυτό παρατίθεται ο κώδικας ανάπτυξης της παρούσας εφαρμογής.

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#include <RTClib.h>

#include <SD.h>

#include <SPI.h>

#define COMMON_ANODE

#define VREG 8

#define MQ7_ANALOG_PIN 0

#define CHIP_SELECT 10

#define redPin 6 //the red pin of the LCD screen

#define greenPin 5 //the green pin of the LCD screen

#define bluePin 3 //the blue pin of the LCD screen

#define MQ7_HEATER_5_V_TIME 60000

#define MQ7_HEATER_1_4_V_TIME 90000

#define CO_VALUE_READING_PERIOD 100
```

```
LiquidCrystal_I2C lcd(0x38,16,2); //set the LCD address to  
0x38 for a display about 16 chars and 2 lines
```

```
RTC_DS1307 rtc;
```

```
unsigned long startMillis;
```

```
unsigned long switchTimeMillis;
```

```
unsigned long id = 1; //the id number of our reading
```

```
boolean heaterInHighPhase;
```

```
String line1 = "InitializingCard...";
```

```
String line2 = "SD card ready!";
```

```
int len1 = line1.length(); //calculates the nubmer of  
characters in string "line"
```

```
int len2 = line2.length();
```

```
int s = 500;
```

```
void setup(){
```

```
pinMode(VREG, OUTPUT); //set the VREG pin as output
```

```

    startMillis = millis(); //saves the the number of
millisecons since the Arduino began running the sketch

    turnHeaterHigh();

    pinMode(redPin, OUTPUT); //set the redPin as output

    pinMode(greenPin, OUTPUT); //set the greenPin as output

    pinMode(bluePin, OUTPUT); //set the bluePin as output

    pinMode(CHIP_SELECT, OUTPUT); //set the CHIP_SELECT pin as
output

    Wire.begin(); //initialize the Wire library and join the I2C
bus

    rtc.begin(); //initialize the real time clock

    lcd.init(); //initialize the LCD screen

    setColor(255, 0, 255); //call the setColor function to set
the color magenta of the RGB LED backlight

    lcd.clear(); //clears the lcd screen and positions the
cursor in the upper-left corner

    int t = 30;

    for(int i=0; i<len1; i++){

        if(i == 12){

            lcd.setCursor(0, 1); //set the cursor of the lcd at the
first column and second row

        }

```

```
if(i == 15){  
    t = 150;  
}  
  
lcd.print(line1[i]); //print a message to the lcd  
  
delay(t);  
}
```

```
//checks if the real time clock works correctly
```

```
if(!rtc.isrunning()){  
    setColor(255, 0, 0); //red  
  
    lcd.clear();  
  
    lcd.print("RTC isnt running");  
  
    delay(1000); //pauses the program for 1000ms  
  
    return;  
}
```

```
//check if the SD card is initialized
```

```
if(!SD.begin(CHIP_SELECT)){  
    setColor(255, 0, 0); //red  
  
    lcd.clear();  
  
    lcd.print("SD card failure!");  
  
    delay(1000);  
  
    return;  
}
```

```

}

setColor(0, 255, 0); //green

lcd.clear();

for(int i=0; i<len2; i++){

    lcd.print(line2[i]);

    delay(30);

}

delay(300);

File datafile = SD.open("co_log.csv", FILE_WRITE);
//creates a file and opens it for writinnng

if(datafile){

    String header = "ID, Mode, CO (ppm), Time, Date";

    datafile.println(header); //prints the header to the file
named datafile

    datafile.close();

}

else{

    setColor(255, 0, 0); //red

    lcd.clear();

    lcd.print("Couldn't open");

    lcd.setCursor(0,1);

    lcd.print("log file...");

```



```

    delay(1000);

    return;
}

}

void loop(){

    if(heaterInHighPhase){

        // 5v phase of cycle. see if need to switch low yet

        if(millis() > switchTimeMillis) {

            turnHeaterLow();

        }

    }

    else{

        // 1.4v phase of cycle. see if need to switch high yet

        if(millis() > switchTimeMillis) {

            turnHeaterHigh();

        }

    }

}

    readCOLevel();

    delay(CO_VALUE_READING_PERIOD);

```

```

}

void turnHeaterHigh(){

    // 5v phase

    digitalWrite(VREG, LOW);

    heaterInHighPhase = true;

    switchTimeMillis = millis() + MQ7_HEATER_5_V_TIME;

}

void turnHeaterLow(){

    // 1.4v phase

    digitalWrite(VREG, HIGH);

    heaterInHighPhase = false;

    switchTimeMillis = millis() + MQ7_HEATER_1_4_V_TIME;

}

void readCOLevel(){

    unsigned int co_value = analogRead(MQ7_ANALOG_PIN);

    DateTime now = rtc.now();

    String timeString;

    String dateString;

    String dataString;

```

```
String mode;

if(heaterInHighPhase){

    setColor(255, 255, 255); //white

    lcd.clear();

    lcd.print("Cleaning Sensing");

    lcd.setCursor(0, 1);

        lcd.print("Elements");

    lcd.setCursor(9, 1);

    lcd.print(co_value);

    mode = "Cleaning Sensing Elements";

}

else{

    if(co_value < 60){

        setColor(0, 255, 0); //green

    }

    else if(co_value < 100){

        setColor(255, 165, 0); //orange

    }

    else{
```

```

        setColor(255, 0, 0); //red
    }

    lcd.clear();

    lcd.print("Taking Readings");

    lcd.setCursor(0, 1);

    lcd.print(co_value);

    lcd.setCursor(5, 1);

    lcd.print("ppm");

    mode = "Taking Readings";
}

if(now.second() % 10 == 0 && now.second() != s){

    s = now.second();

    timeString = String(now.hour()) + ":" +
String(now.minute()) + ":" + String(now.second());

    dateString = String(now.day()) + "/" + String(now.month())
+ "/" + String(now.year());

    dataString = String(id) + "," + mode + "," +
String(co_value) + "," + timeString + "," + dateString;

File datafile = SD.open("co_log.csv", FILE_WRITE);

if(datafile){

    datafile.println(dataString);

    datafile.close();
}
}

```

```

    }

    else{

        setColor(255, 0, 0); //red

        lcd.clear();

        lcd.print("Failed to open");

        lcd.setCursor(0,1);

        lcd.print("the file");

        delay(1000);

    }

    id++; //increment the id number

}

}

void setColor(int red, int green, int blue) //sets the color
of the RGB LED backlight

{

    //inverts the values because is common anode RGB LED

    #ifdef COMMON_ANODE

        red = 255 - red;

        green = 255 - green;

        blue = 255 - blue;

```

```
#endif

    analogWrite(redPin, red); //writes the value of the
variable "red" to the pin "redPin"

    analogWrite(greenPin, green);

    analogWrite(bluePin, blue);

}
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

Σελίδες από το Διαδίκτυο

- [1] <http://el.wikipedia.org/wiki/%CE%9C%CE%B9%CE%BA%CF%81%CE%BF%CE%B5%CE%BB%CE%B5%CE%B3%CE%BA%CF%84%CE%AE%CF%82>
- [2] <http://en.wikipedia.org/wiki/Microcontroller>
- [3] <http://www.newagepublishers.com/samplechapter/001599.pdf>
- [4] <http://www.engineersgarage.com/tutorials/difference-between-microprocessor-and-microcontroller>
- [5] <http://www.engineersgarage.com/microcontroller>
- [6] <http://www.mikroe.com/chapters/view/65/chapter-2-8051-microcontroller-architecture/>
- [7] <http://cgi.di.uoa.gr/~std04013/>
- [8] <http://el.wikipedia.org/wiki/Arduino#.CE.95.CE.BA.CE.B4.CF.8C.CF.83.CE.B5.CE.B9.CF.82>
- [9] <http://deltahacker.gr/arduino-intro/>
- [10] <http://arduino.cc/en/Main/ArduinoBoardUno>
- [11] <http://openlab.teipir.gr/2012/05/arduino-uno/>
- [12] <https://el.wikipedia.org/wiki/%CE%9C%CE%BF%CE%BD%CE%BF%CE%BE%CE%B5%CE%AF%CE%B4%CE%B9%CE%BF%CF%84%CE%BF%CF%85%CE%AC%CE%BD%CE%B8%CF%81%CE%B1%CE%BA%CE%B1>
- [13] <http://lap.physics.auth.gr/atmdiasp/simeiwseis/chapter2.pdf>
- [14] <https://en.wikipedia.org/wiki/User:Lerisker>
- [15] Kitchen, M. (2005). A history of modern Germany, 1800–2000. Wiley-Blackwell ([ISBN 1-4051-0041-9](https://doi.org/10.1017/9781405100419)).

- [16] http://en.wikipedia.org/wiki/Data_logger
- [17] <https://learn.adafruit.com/adafruit-data-logger-shield>
- [18] <https://learn.adafruit.com/adafruit-data-logger-shield/overview>
- [19] <https://learn.adafruit.com/adafruit-data-logger-shield/shield-overview>
- [20] <http://www.instructables.com/id/Logger-Shield-Datalogging-for-Arduino/step1/Overview/>
- [21] <https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf>
- [22] <https://www.fairchildsemi.com/datasheets/KA/KA278RA05C.pdf>
- [23] <http://www.vishay.com/docs/83836/lh1546ad.pdf>
- [24] <http://www.adafruit.com/products/399>
- [25] <http://learning.grobotronics.com/2013/07/controlling-an-lcd-display-hd44780-via-i2c-bus/>
- [26] <http://www.adafruit.com/datasheets/HD44780.pdf>
- [27] http://www.nxp.com/documents/data_sheet/PCF8574.pdf

Ελληνική Βιβλιογραφία

- Βασιλικιώτης, Γ.Σ. (1989). Χημεία Περιβάλλοντος. Εκδόσεις University Studio Press, Θεσσαλονίκη.
- Γεντεκάκης, Ι.Β. (2003). Ατμοσφαιρική Ρύπανση, Επιπτώσεις, Έλεγχος & Εναλλακτικές Τεχνολογίες. Εκδόσεις Τζιόλα, Θεσσαλονίκη.
- Καρβούνης, Σ., Γεωργακέλλος, Δ. (2003). Διαχείριση του περιβάλλοντος. Εκδόσεις Σταμούλη, Αθήνα.
- Κωτσοβίνος, Ν. (1992). Ρύπανση και προστασία περιβάλλοντος. Εκδόσεις Φούντας, Αθήνα.