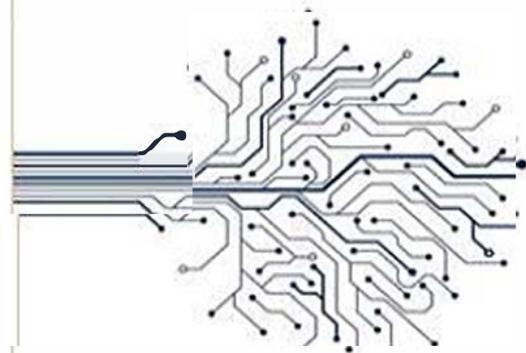


Car Real Time IoT Data Logger



Σχεδίαση & Ανάπτυξη IoT
Ενσωματωμένου
Συστήματος Real-Time
Παρακολούθησης &
Καταγραφής
Αισθητήρων & OBD II
Δεδομένων



Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

Πάτρα

Ημερομηνία

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1.
2.
3.

Υπεύθυνη Δήλωση Φοιτητή

Βεβαιώνω ότι είμαι συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά. Η έγκριση της διπλωματικής εργασίας από το Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος. Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία τ__ φοιτητ__ _____ που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης ο συγγραφέας/δημιουργός εκχωρεί στο Πανεπιστήμιο Πελοποννήσου, μη αποκλειστική άδεια χρήσης του δικαιώματος αναπαραγωγής, προσαρμογής, δημόσιου δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσής τους διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος και για όλο το χρόνο διάρκειας των δικαιωμάτων πνευματικής ιδιοκτησίας. Η ανοικτή πρόσβαση στο πλήρες κείμενο για μελέτη και ανάγνωση δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, αποθήκευση, πώληση, εμπορική χρήση, μετάδοση, διανομή, έκδοση, εκτέλεση, «μεταφόρτωση» (downloading), «ανάρτηση» (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού. Ο συγγραφέας/δημιουργός διατηρεί το σύνολο των ηθικών και περιουσιακών του δικαιωμάτων

Εισαγωγή

Στόχος της παρούσας διπλωματικής εργασίας, είναι η σχεδίαση, η ανάπτυξη και η υλοποίηση IoT Ενσωματωμένου Συστήματος το οποίο αφορά Real-Time παρακολούθηση και καταγραφή μεγάλου εύρους λειτουργιών και πληροφοριών του αυτοκίνητου μέσω των παρακάτω:

Αναλογικοί αισθητήρες, OBD-II δεδομένα και σήματα γεωγραφικής τοποθεσίας. Έπειτα από επεξεργασία και αποθήκευση σε SD-Card όλα τα δεδομένα τελικώς απεικονίζονται μέσω του **ThinkSpeak** στο Cloud

Στην εργασία χρησιμοποιούνται δυο από τις πιο σύγχρονες πλατφόρμες ανοιχτού κώδικα για την κατασκευή ηλεκτρονικών έργων, οι οποίες είναι **Arduino uno** και **mega**.

Δοκιμές έγιναν επίσης και σε **Raspberry Pi 4**.

Το συγκεκριμένο θέμα παρουσιάζει μεγάλο τεχνολογικό ενδιαφέρον, καθώς συνδυάζει τεχνολογίες επικοινωνίας και αλληλεπίδρασης με εγκέφαλους αυτοκίνητων μέσω Bluetooth και πρωτοκόλλων για ασύρματη επικοινωνία μικρών αποστάσεων. Η επεξεργασία και η λήψη των **OBD-II** δεδομένων γίνεται με τα λεγόμενα **PIDs (On-board diagnostics Parameter IDs)** έτσι ορίζονται οι κωδικοί που χρησιμοποιούνται για την αίτηση δεδομένων από ένα όχημα τα οποία παρέχονται μέσω της θύρας OBD.

Κατά την ανάπτυξη της συγκεκριμένης εργασίας οι μικροελεγκτες σε συνδυασμό με αναλογικούς και ψηφιακούς αισθητήρες καθώς και σήματα από τις θέσεις των δορυφόρων **GPS**, δημιουργούν ένα ενιαίο σύστημα παρακολούθησης σε πραγματικό χρόνο, του οποίου η υλοποίηση αναλύεται και απεικονίζεται ενδελεχώς στο Κεφάλαιο 4.

Η δε ανάπτυξη και ενσωμάτωση του κώδικα γίνεται μέσω του **Arduino IDE** και των απαιτούμενων βιβλιοθηκών οι οποίες βασίζονται σε C και C++.

Ευχαριστίες

Η συγκεκριμένη πτυχιακή εργασία διεξήχθη στα πλαίσια του προγράμματος σπουδών του πρώην τμήματος Μηχανικών Πληροφορικής Τ.Ε.Ι. Δυτικής Ελλάδας, νυν τμήμα Ηλεκτρολόγων Μηχανικών Και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου.

Θα ήθελα να ευχαριστήσω τον επιβλέπων καθηγητή μου κύριο Χρήστο Αντωνόπουλο, για την πολύτιμη βοήθεια και καθοδήγηση που μου προσέφερε καθ'όλη την διάρκεια έρευνας και υλοποίησης της πτυχιακής εργασίας, όπως και τον Κύριο Παρασκευά Κίτσο επίσης καθηγητή του τμήματος, ο οποίος στάθηκε δίπλα μου μη αφήνοντας με να χάσω το κίνητρο μου και την προσήλωση στο στόχο.

Πίνακας περιεχομένων

Κεφάλαιο 1^ο - Εισαγωγή	5
1.1 ECUs & Microcontrollers.....	5
1.2 Ιστορική Αναδρομή ECUs (EngineControlUnit)	11
1.3 Συναφείς Πλατφόρμες - Εφαρμογές.....	17
Κεφάλαιο 2^ο - Ενσωματωμένα & Μικροελεγκτές	19
2.1 Ενσωματωμένα Συστήματα	19
2.2 Εισαγωγή στους Μικροελεγκτές	20
2.3 Ιστορική Αναδρομή Μικροεπεξεργαστών - Μικροελεγκτών	21
2.4 Κατηγορίες Μικροελεγκτών	25
2.5 Βασικά Χαρακτηριστικά Μικροελεγκτή	26
2.6 Πρόσθετες Λειτουργίες Μικροελεγκτή	28
2.7 Προγραμματισμός Μικροελεγκτών	29
Κεφάλαιο 3^ο - Θεωρητικό Υπόβαθρο	31
3.1 Arduino	31
3.1.1 Arduino UNO Pinout.....	33
3.1.2 Arduino MEGA 2560 Pinout.....	36
3.2 Arduino Modules	39
3.2.1 Gps Module (L76X)	39
3.2.2 Bluetooth (HC-05)	41
3.2.3 Wifi (ESP-8266).....	44
3.2.4 SD Card Module	48
3.3 Raspberry Pi 4.....	50
3.3.1 AnalogToDigital Converter	52
3.4 ECUs με ELM 327 ,OBD II & Πρωτόκολλα	55
3.4.1 OBD II - PID Commands	59
3.5 After-Market Όργανα & Αισθητήρες	61
3.6 ThinkSpeak	63
Κεφάλαιο 4^ο - Υλοποίηση Συστήματος	66
4.1 Σχεδίαση του (Car Real Time IoT Data Logger) Based On Raspberry Pi	66
4.2 Σχεδίαση του (Car Real Time IoT Data Logger) Based On Arduino.....	68
4.2.1 Σχεδίαση και Υλοποίηση Βάσης και Σειριακής Επικοινωνίας	70
4.3 Σχεδίαση και Υλοποίηση Συνδέσεων Συστήματος & Αισθητήρων	73
4.4 Ανάπτυξη του Software	75
Κεφάλαιο 5^ο - Συμπεράσματα & Μελλοντικές Προεκτάσεις.....	77
5.1 Περίληψη	77
5.2 Προβλήματα - Παρατηρήσεις.....	78
5.3 Μελλοντικές Προεκτάσεις.....	79
5.4 Βιβλιογραφία	80

Κεφάλαιο 1^ο - Εισαγωγή

1.1 ECUs & Microcontrollers



Εικόνα 1.1: Κεντρική Ηλεκτρονική Μονάδα Ελέγχου (ECU)

Αναμφισβήτητα, ένα από τα πιο αντιπροσωπευτικά δείγματα της ραγδαίας τεχνολογικής εξέλιξης των τελευταίων δεκαετιών είναι και το αυτοκίνητο. Ωστόσο, σε σχέση με τις ξεπερασμένες πλέον κατασκευές περασμένων ετών η εφαρμογή της ηλεκτρονικής έχει αλλάξει ριζικά την αρχιτεκτονική των σύγχρονων μοντέλων. Σχεδόν όλες οι λειτουργίες του σύγχρονου αυτοκινήτου ελέγχονται πλέον από πλειάδα ηλεκτρονικών συστημάτων τα οποία έχουν εκτοπίσει αιμίμηστα εξαρτήματα όπως π.χ. το καρμπυρατέρ. Έτσι λοιπόν όλες οι λειτουργίες του αυτοκινήτου ελέγχονται από ηλεκτρονικές μονάδες επεξεργασίας δεδομένων. Καμία όμως από αυτές όμως δεν είναι τόσο ζωτική όσο αυτή που διαχειρίζεται την λειτουργία του κινητήρα.

ECU: Ο εγκέφαλος του κινητήρα

Οι ηλεκτρονικές μονάδες οι οποίες διαχειρίζονται αποκλειστικά τον κινητήρα ονομάζονται **ECU** ή **EMS** και τα αρκτικόλεξα τους βασίζονται στους πλήρη ορισμούς «**Engine Control Unit**» και «**Engine Management System**». Ο ανθρώπινος εγκέφαλος θα ήταν πρακτικά άχρηστος αν δεν υπήρχαν τα αισθητήρια όργανα του σώματος για να τον τροφοδοτούν συνεχώς με διάφορα ερεθίσματα από το περιβάλλον.

Κάπως έτσι λειτουργεί και η ηλεκτρονική μονάδα ελέγχου του κινητήρα η οποία κυριολεκτικά βομβαρδίζεται συνεχώς με δεδομένα τα οποία καταφθάνουν από διάφορους αισθητήρες στην μορφή ηλεκτρικής τάσης. Την κατάσταση του κινητήρα αντιλαμβάνονται περισσότεροι από πενήντα σένσορες οι οποίοι ενημερώνουν τον εγκέφαλο με διάφορες βασικές παραμέτρους όπως η περιστροφική ταχύτητα του μοτέρ, συγκεκριμένα του στροφαλοφόρου άξονα και το φορτίο. Ωστόσο, υπάρχουν και άλλα αισθητήρια όργανα τα οποία «διαβάζουν» την θέση της «πεταλούδας», την παροχή και την θερμοκρασία του αέρα στην πολλαπλή εισαγωγής (ώστε να υπολογισθεί η πυκνότητα του), την θερμοκρασία του σώματος του κινητήρα, την περιεκτικότητα του αέρα σε οξυγόνο και την τάση της μπαταρίας.

Η αρχιτεκτονική της ECU



Εικόνα 1.2: Διάταξη Στοιχείων της ECU

Το βασικό μέρος ενός εγκέφαλου (ECU) αποτελεί ένας κεντρικός μικροεπεξεργαστής (CPU) ο οποίος στις περισσότερες περιπτώσεις είναι τεχνολογίας 32-bit και «τρέχει» στα 20MHz. Τα ηλεκτρικά σήματα από τους αισθητήρες μετατρέπονται στην είσοδο τους σε ψηφιακά ώστε η CPU να έχει την δυνατότητα να τα επεξεργασθεί. Αφού τελικά ο εγκέφαλος επεξεργασθεί τα δεδομένα η τελική του εντολή μετατρέπεται κατά την έξοδο από ψηφιακή και πάλι σε αναλογική μορφή -δηλ. σε σήματα τάσης- ώστε να ενεργοποιηθούν για παράδειγμα οι ηλεκτρομαγνητικές βαλβίδες του συστήματος ψεκασμού. Οι αριθμητικοί υπολογισμοί εκτελούνται σε συνδυασμό με πολύπλοκα αλγοριθμικά προγράμματα τα οποία υπάρχουν στην μνήμη.

Η τελευταία αποτελείται από τρία βασικά στοιχεία, τις μνήμες RAM, ROM και PROM. Η RAM αποθηκεύει προσωρινά τα δεδομένα εισόδου από τους αισθητήρες και εξόδου πριν αρχίσει το πρόγραμμα ελέγχου. Στην ROM έχουν αποθηκευθεί μόνιμα πληροφορίες ενώ η συγκεκριμένη μνήμη επικοινωνεί απευθείας με το πρόγραμμα ελέγχου. Η μνήμη PROM περιέχει σε μορφή αλγοριθμικού προγράμματος όλα τα κατασκευαστικά στοιχεία λειτουργίας του κινητήρα και υπάρχει δυνατότητα επαναπρογραμματισμού. Επιπλέον υπάρχει και μία ακόμα μνήμη η KAM η οποία αποτελεί ουσιαστικά παραλλαγή της RAM και σε αυτή η CPU μπορεί να διαβάσει και να αποθηκεύσει νέα δεδομένα λειτουργίας.

Η ηλεκτρονική στην υπηρεσία του κινητήρα

Σε τι τελικά χρειάζεται ο ηλεκτρονικός εγκέφαλος; Η ανάγκη για μειωμένη κατανάλωση καυσίμου και η αυστηρή νομοθεσία περί των εκπομπών ρύπων επέβαλλε κατά κάποιο τρόπο την εφαρμογή της ηλεκτρονικής η οποία χαρακτηρίζεται από δύο βασικά στοιχεία: ακρίβεια και ταχύτητα. Η συλλογή όλων αυτών των πληροφοριών η ECU της χρειάζεται προκειμένου να ρυθμίσει με απόλυτη ακρίβεια την λειτουργία των συστημάτων ψεκασμού και ανάφλεξης. Όταν η ECU δίνει εντολή στο σύστημα ψεκασμού στην πράξη ορίζει την στιγμή που θα ενεργοποιηθεί η ηλεκτρομαγνητική βαλβίδα του εγχυτήρα (μπεκ) και την διάρκεια που θα μείνει ανοιχτή ρυθμίζοντας έτσι την ποσότητα του μίγματος στον θάλαμο καύσης. Όσον αφορά στο σύστημα ανάφλεξης η ECU ρυθμίζει την στιγμή της ανάφλεξης του σπινθηριστή (μπουζί) καθώς και την προπορεία ανάφλεξης (αβάνς).

Για να κατανοήσουμε καλύτερα όλα τα παραπάνω ας πάρουμε ένα παράδειγμα. Σε έναν τετρακύλινδρο τετράχρονο κινητήρα, σε κάθε πλήρη περιστροφή του στροφαλοφόρου του άξονα εκτελούνται δύο ψεκασμοί με ακρίβεια χιλιοστών του δευτερολέπτου. Για παράδειγμα, στο αυτοκίνητο μας την στιγμή που ο δείκτης του στροφόμετρου βρίσκεται στις 6.000 σ.α.λ. τότε στους κυλίνδρους του κινητήρα πραγματοποιούνται 12.000 ψεκασμοί το λεπτό, δηλαδή 200 το δευτερόλεπτο! Παράλληλα, η ECU λαμβάνει υπόψη της και της πληροφορίες που δίνει ο καταλύτης ρυθμίζοντας έτσι την αναλογία του στοιχειομετρική αναλογία (αέρας-καύσιμο) ανάλογα ώστε οι εκπομπές ρύπων να μην ξεπερνούν τις προδιαγραφές που ολοένα και γίνονται αυστηρότερες.

Στα ξεπερασμένα συστήματα με καρμπυρατέρ που δεν υπήρχε ηλεκτρονικός εγκέφαλος για να διεκπεραιωθεί ομαλά η καύση του μίγματος δίνονταν μεγάλη προπορεία ανάφλεξης ενώ το αναλογία του μίγματος ήταν πλουσιότερη από ότι θα έπρεπε επιβαρύνοντας τον τομέα της κατανάλωσης αλλά και την μόλυνση του περιβάλλοντος. Επιπλέον, η ECU επικοινωνεί και με τις υπόλοιπες μονάδες όπως για παράδειγμα αυτή του traction control που στην ουσία είναι παρόμοια με αυτή του ESP και χρησιμοποιούν τους αισθητήρες του ABS. Όταν το σύστημα αντιληφθεί σπινάρισμα των κινητήριων τροχών τότε η εν λόγω κατάσταση μεταφέρεται στην ECU και αυτή ανάλογα επεμβαίνει στην ισχύ του κινητήρα μειώνοντας την ροπή του. Ένας άλλος αισθητήρας ελέγχει την κατάσταση της μπαταρίας. Αν αυτή είναι μειωμένη ενημερώνει την ECU και τότε αυτή ανεβάζει τις στροφές λειτουργίας του κινητήρα προκειμένου το δυναμό να την φορτίσει.

Μικροελεγκτής

Ο εγκέφαλος αποτελείται από ένα ή περισσότερα τυπωμένα κυκλώματα (πλακέτες) όπου ένας κεντρικός μικροεπεξεργαστής, υποστηριζόμενος από μνήμες και άλλα περιφερειακά, διαβάζει και ελέγχει, μέσω κατάλληλης εξωτερικής αρτηρίας (bus) δεδομένων αλλά και απευθείας σύνδεσης, τα διάφορα υποσυστήματα του αυτοκινήτου. Στα σύγχρονα επιβατικά και φορτηγά οχήματα, το καθιερωμένο εξωτερικό bus είναι το **CAN**.

CAN Bus

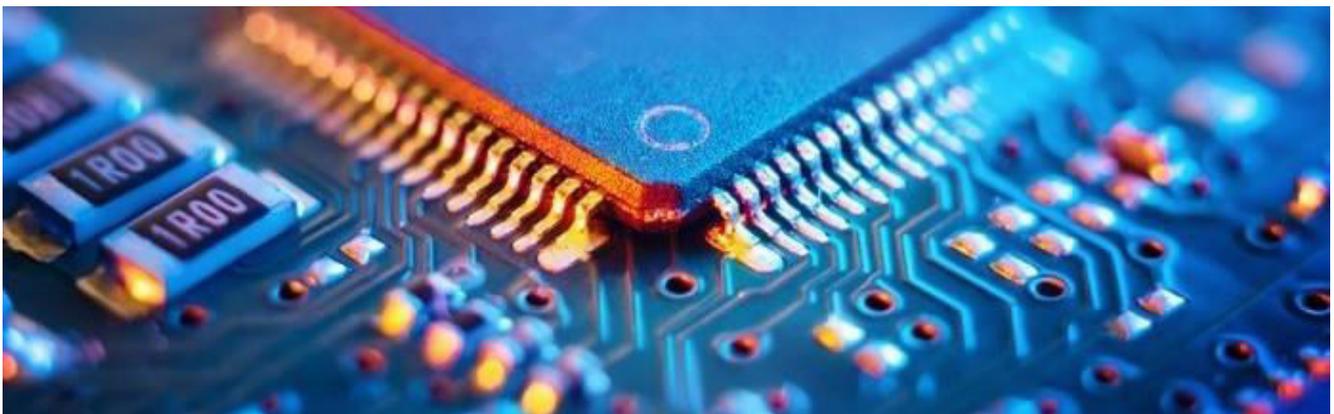
Το CAN bus είναι ένας σειριακός ψηφιακός τρόπος σύνδεσης του εγκεφάλου με τα υποσυστήματα του αυτοκινήτου, ο οποίος χρησιμοποιεί μόλις 2 καλώδια. Πάνω σε αυτά τα 2 καλώδια είναι συνδεδεμένα τα περιφερειακά με σύνδεση συμβατή με CAN bus, όπως έξυπνοι αισθητήρες, μονάδα ηλεκτρονικής ανάφλεξης, υποσύστημα ABS, ενσωματωμένος υπολογιστής ταξιδιού, ελεγκτές φώτων. Εκεί συνδέεται και η διαγνωστική υποδοχή (φίσα), μέσω της οποίας το διαγνωστικό μηχάνημα του συνεργείου μπορεί να πάρει πληροφορίες για την κατάσταση του οχήματος και να εντοπίσει τη βλάβη.

Έτσι, τα αναλογικά περιφερειακά μπορούν να διαθέτουν δικό τους μετατροπέα αναλογικού σε ψηφιακό και μικροελεγκτή, π.χ. έξυπνοι αισθητήρες και ηλεκτρομηχανικοί ενεργοποιητές (actuators), ώστε οι καλωδιώσεις προς τον εγκέφαλο να είναι καθαρά ψηφιακές. Αυτή η φιλοσοφία απλοποιεί αισθητά την καλωδίωση δίνοντας ταυτόχρονα δυνατότητα για σύνθετες λειτουργίες, ενώ διευκολύνει τη διάγνωση βλαβών.

Safe mode

Τι γίνεται όμως σε περίπτωση που κάποιος αισθητήρας εμφανίσει βλάβη; Στο software του BIOS υπάρχει ένα βοηθητικό πρόγραμμα ανάγκης που ανιχνεύει την δυσλειτουργία κάποιου βασικού σένσορα και θέτει τον κινητήρα σε λειτουργία «safe mode». Σε ασφαλείς συνθήκες λειτουργίας -τις περισσότερες φορές μέχρι τις 3.500σ.α.λ.- το αυτοκίνητο μπορεί να φτάσει μέχρι το πλησιέστερο συνεργείο. Αν η βλάβη δεν επιτρέπει την κατάσταση safe mode το αυτοκίνητο ακινητοποιείται και το κόστος αντικατάστασης του εγκεφάλου ξεκινά από περίπου 1.000 ευρώ ενώ σε πολυτελή μοντέλα συνήθως ξεπερνά τα 3.000 ευρώ. Οι περισσότερες αυτοκινητοβιομηχανίες προμηθεύονται εγκεφάλους από την Bosch, την Siemens, την Magneti Marelli, την Denso και την Delphi.

Ο τρόπος σκέψης της ECU



Εικόνα 1.3: Κεντρικός Μικροεπεξεργαστής ECU

Οι εντολές του εγκεφάλου διαμορφώνονται βάση περίπλοκου αλγοριθμικού προγράμματος εξαιτίας των πολλών παραμέτρων δεδομένων. Το software θα πρέπει ταυτόχρονα να ικανοποιήσει

διάφορες απαιτήσεις όπως οι εκπομπές ρύπων και η μειωμένη κατανάλωση ενώ ταυτόχρονα θα πρέπει να προστατεύει τον κινητήρα σε περίπτωση δυσλειτουργίας. Ο αλγόριθμος της βασίζεται σε μια αλληλουχία ορίζουσων πινάκων οι οποίοι καθορίζουν το μήκος κύματος, το οποίο στην πραγματικότητα είναι τα δεδομένα τα οποία η ECU επεξεργάζεται. Οι πίνακες αυτοί αντιστοιχούν σε διαφορετικές παραμέτρους οι οποίες αντιστοιχούν σε τιμές. Ας επιχειρήσουμε να αναλύσουμε στην απλούστερη μορφή του ένα υποθετικό πίνακα ο οποίος στο σύστημα ψεκασμού λαμβάνοντας υπόψη τρεις από τους δεκάδες παραμέτρους που μπορούν να επηρεάσουν την όλη κατάσταση. Ας υποθέσουμε ότι βάση του πίνακα 1 ο κινητήρας μας λειτουργεί στις 2.000σ.α.λ. με φορτίο σε επίπεδο 4. Αν διασταυρώσουμε αυτά τα νούμερα τότε διαπιστώνουμε ότι οι τιμές αυτές αντιστοιχούν σε 8 milliseconds το οποίο είναι το θεωρητικό μήκος κύματος.

Στροφές κινητήρα ανά λεπτό (σ.α.λ.)	Φορτίο				
	1	2	3	4	5
1000	1	2	3	4	5
2000	2	4	6	8	10
3000	3	6	9	12	15
4000	4	8	12	16	20

Πίνακας 1: Θεωρητικού κύματος

Στον δεύτερο πίνακα οι παράγοντες A και B αφορούν παραμέτρους συγκεκριμένων αισθητήρων. Ας υποθέσουμε ότι ο A αντιστοιχεί στον αισθητήρα θερμοκρασίας του κινητήρα και ο B στον αισθητήρα περιεκτικότητας του αέρα σε οξυγόνο. Αν η θερμοκρασία αντιστοιχεί με 100 και η ποιότητα του αέρα με 3 τότε ο A ισούται με 0,8 και ο B με 1,0.

A	Παράμετρος A		B	Παράμετρος B
0	1,2		0	1,0
25	1,1		1	1,0
50	1,0		3	1,0
75	0,9		3	1,0
100	0,8		4	0,75

Πίνακας 2: Αισθητήρων θερμοκρασίας και οξυγόνου

Σε τι μας χρειάζονται όμως όλα αυτά τα δεδομένα; Στην μνήμη του εγκεφάλου είναι καταχωρημένες διάφορες εξισώσεις όπως αυτή που συνδέει τα παραπάνω στοιχεία και μας δίνει το πραγματικό μήκος κύματος (το διάστημα δηλαδή που θα μείνει ανοιχτό το μπεκ) το οποίο ισούται

με το θεωρητικό μήκος κύματος επί τους παράγοντες A και B. Με λίγα λόγια ισχύει ότι: Πραγματικό μήκος κύματος = Θεωρητικό μήκος κύματος (σ.α.λ., φορτίο) x (A) x (B). Άρα: $8 \times 0,8 \times 1,0 = 6,4$ χιλιοστά του δευτερολέπτου. Προς αποφυγήν οποιασδήποτε παρεξηγήσεως το εν λόγω παράδειγμα διαμορφώθηκε κατάλληλα ώστε να γίνει κατανοητός ο τρόπος σκέψης της ECU. Στην πράξη τα πράγματα είναι πολύ δυσκολότερα όταν διασταυρώνονται σε κλάσματα του δευτερολέπτου περισσότερες από 100 τέτοιες παραμέτρους με χιλιάδες πληροφορίες να φτάνουν κάθε λεπτό.

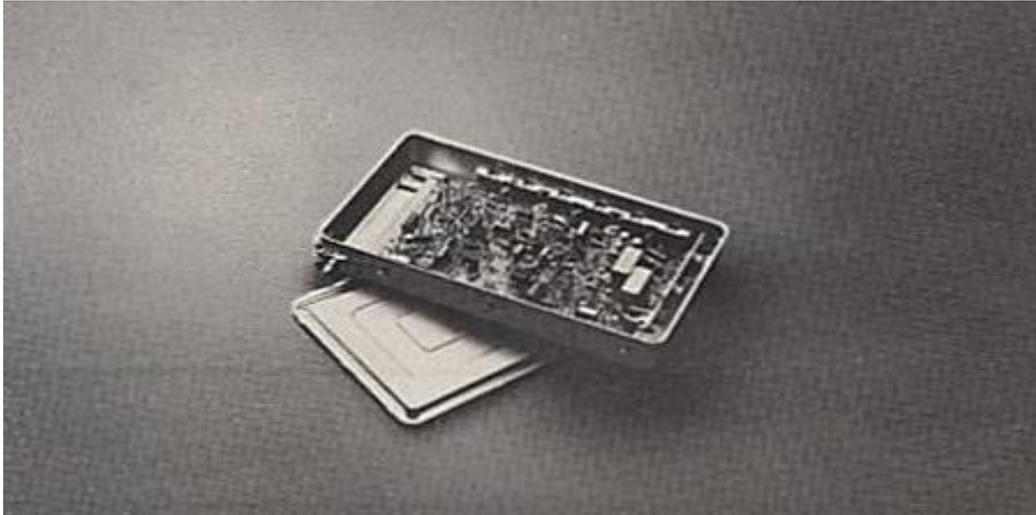
ECU Tunning(Αναβάθμιση Εγκεφάλου)



Εικόνα 1.4: Βελτιωτικό Chip πάνω στην ECU

Ουσιαστικά τα «τσιπάκια» βελτίωσης του κινητήρα δεν είναι παρά μνήμες PROM και στις περισσότερες των περιπτώσεων διατίθενται στην αγορά ως aftermarket προϊόντα. Η προσθήκη ή η αντικατάσταση με ένα τέτοιο τσιπ αλλάζει τον προγραμματισμό του εγκεφάλου και κατ' επέκταση την χαρτογράφηση του κινητήρα (συστήματα ψεκασμού και ανάφλεξης). Ωστόσο, η εν λόγω επεμβάσεις σε μη εξουσιοδοτημένα συνεργεία αλλοιώνουν την εγγύηση που δίνει ο κατασκευαστής για τα μηχανικά μέρη του αυτοκινήτου. Τι κάνει όμως αυτό το τσιπάκι; Στην μνήμη του υπάρχουν διαφορετικοί αλγοριθμικοί πίνακες από αυτούς του κατασκευαστή αλλάζοντας τον τρόπο σκέψης του εγκεφάλου ο οποίος πλέον περνά σε δεύτερη μοίρα τον έλεγχο των ρύπων δίνοντας προτεραιότητα στην αύξηση της απόδοσης του κινητήρα. Τι διαφορά έχουν τα τσιπάκια του κατασκευαστή με αυτά που κυκλοφορούν ως aftermarket; Τα πρώτα μπορεί να κοστίζουν κάτι παραπάνω είναι όμως περισσότερο αξιόπιστα και ευαισθητοποιημένα στον τομέα της κατανάλωσης.

1.2 Ιστορική Αναδρομή ECUs (EngineControlUnit)



Εικόνα 1.5: Ο πρώτος υπολογιστής αυτοκινήτων της VW (1968)

Η πρώτη χρήση υπολογιστή σε αυτοκίνητο ήταν απλώς για τον έλεγχο του κινητήρα. Οι αυτοκινητοβιομηχανίες άρχισαν να εισάγουν πρώτες εκδόσεις συστημάτων ελεγχόμενων από υπολογιστή για την εκτέλεση μιας συγκεκριμένης λειτουργίας. Το 1968, η Volkswagen παρουσίασε το πρώτο όχημα με ηλεκτρονικό σύστημα ηλεκτρονικού ψεκασμού καυσίμου (EFI) ελεγχόμενο από υπολογιστή - το D-Jetronic, μια τρανζίστορ ηλεκτρονική μονάδα που κατασκευάστηκε από τη Bosch. Προσφέρθηκε ως στάνταρ εξοπλισμός στα μοντέλα τους τύπου 3.

Τα ECU έχουν γίνει μια τυπική συσκευή για τα περισσότερα αυτοκίνητα από τα τέλη της δεκαετίας του 1970, όταν έγιναν απαραίτητα λόγω των ολοένα αυστηρότερων κυβερνητικών προτύπων εκπομπών.

Ο έλεγχος του κινητήρα είναι η πιο έντονη διαδικασία επεξεργασίας στο αυτοκίνητό σας και η μονάδα ελέγχου κινητήρα είναι ο πιο ισχυρός υπολογιστής στα περισσότερα αυτοκίνητα. Η ECU χρησιμοποιεί έλεγχο κλειστού βρόχου, ένα σύστημα ελέγχου που παρακολουθεί τις εξόδους ενός συστήματος για τον έλεγχο των εισροών σε ένα σύστημα διαχειριζόμενη τις εκπομπές και την οικονομία καυσίμου του κινητήρα (καθώς και πολλές άλλες παραμέτρους).

Συλλέγοντας δεδομένα από δεκάδες διαφορετικούς αισθητήρες, η ECU γνωρίζει τα πάντα, από τη θερμοκρασία του ψυκτικού υγρού, μέχρι την ποσότητα οξυγόνου στην εξάτμιση.

Με αυτά τα δεδομένα, η ECU εκτελεί εκατομμύρια υπολογισμούς κάθε δευτερόλεπτο, συμπεριλαμβανομένης της αναζήτησης τιμών σε πίνακες, υπολογισμού των αποτελεσμάτων μακρών εξισώσεων για την επιλογή του καλύτερου χρονισμού σπινθήρων και προσδιορισμού του χρόνου λειτουργίας του μπέκ ψεκασμού καυσίμου. Η ECU κάνει όλα αυτά για να εξασφαλίσει τις χαμηλότερες εκπομπές και τα καλύτερα χιλιόμετρα.

Η Motorola, η Intel και άλλοι συνεργάζονται με τους κατασκευαστές αυτοκινήτων τοποθετώντας τα πρώτα τσιπ μικροεπεξεργαστή στα αυτοκίνητα.



Εικόνα 1.6: Ford & Motorola PTEC Automotive Microcontroller chips (1994)

Το 1976, η General Motors (GM) ανακοίνωσε μια νέα εταιρική σχέση με τη Motorola Semiconductor να αναπτύξει μια προσαρμοσμένη μικροϋπολογιστών για χρήση σε οχήματα τους. Μέχρι το 1981, όλα τα οχήματα της GM θα εφοδιάζονταν με το νέο τους σύστημα ελέγχου των εκπομπών Computer Command System ("CCC") το οποίο διέθετε ECM (Electronic Control Module) που διέθετε μικροεπεξεργαστή 8-bit βασισμένο σε Motorola 6802 που κατασκευάστηκε από την Delco Electronics. Στα τέλη της δεκαετίας του 1980, η GM και η Chrysler ECM άρχισαν και οι δύο να χρησιμοποιούν το τσιπ μικροελεγκτή Motorola 68HC11 16-bit στα οχήματά τους.

Το 1983, η Intel και η Ford ξεκίνησε μια κοινοπραξία που χτίζει μονάδες EEC (Electronic Engine Control) που είναι πλέον γνωστές ως ECU (Electronic control Units). Αυτά τα ECU χρησιμοποίησαν προσαρμοσμένο Intel 8061 (παράγωγο του μικροελεγκτή Intel 8096) και αργότερα χρησιμοποίησαν τσιπ μικροελεγκτή 8065, 83251 & 8051 για τις λειτουργίες επεξεργασίας του. Οι 8061 μικρο-ελεγκτές και τα παράγωγά του χρησιμοποιούνται σχεδόν σε όλα τα Ford αυτοκίνητα που χτίστηκε από το 1983 μέσα από το 1994.

Το 1985, η Ford Μικροηλεκτρονικής ανακοίνωσε ότι θα αρχίσει να σχεδιάσει και να επιβλέψει την κατασκευή του βασίζεται γαλλίου αρσενικούχο ολοκληρωμένων κυκλωμάτων σε τους Colorado Springs CO. Φυτών για χρήση στα δικά τους αυτοκίνητα και αεροδιαστημικά προϊόντα. Η Ford Microelectronics δημιουργήθηκε το 1982 για να αναπτύξει προσαρμοσμένα ολοκληρωμένα κυκλώματα για αυτοκινητιστικές χρήσεις και για τη Ford Aerospace.

Το 1994, σε μια τελευταία προσπάθεια να σώσει τον επιβραδυνόμενο μικροελεγκτή EEC IV, η Ford χρησιμοποίησε δύο τσιπ 16-bit Intel 8065 στο εσωτερικό του ελεγκτή. Λίγο αργότερα, η Ford θα ήταν έτοιμη να τερματίσει τη μακροχρόνια χρήση μικροελεγκτών Intel και τη μετάβαση στη χρήση των νέων ελεγκτών PTEC 32-bit της Motorola.

Το 2005, η Intel ανακοίνωσε ότι διακόπτει την παραγωγή όλων των εκδόσεων αυτοκινήτων των τσιπ μικροελεγκτών τους.

Προηγουμένως το 1991, η Διεύθυνση Ηλεκτρονικών της Ford είχε ανακοινώσει ότι θα συνεργαζόταν με την Motorola Semiconductor Products Sector για να τους προμηθεύσει μελλοντικά ηλεκτρονικά τσιπ μικροελεγκτών κινητήρα και μετάδοσης που θα χρησιμοποιηθούν στον επερχόμενο ελεγκτή ηλεκτρονικών συστημάτων Powertrain (PTEC). Ο ελεγκτής PTEC αρχικά επρόκειτο να χρησιμοποιήσει το MC88300 ενσωματωμένο τσιπ RISC 32-bit που είχε η Motorola στην ανάπτυξη, αλλά αργότερα θα ακυρωθεί. Η Ford επέλεξε να χρησιμοποιήσει το νέο προσαρμοσμένο τσιπ επεξεργαστή PowerPC βασισμένο σε RISC 32-bit της Motorola. Οι μικροελεγκτές PTEC χρησιμοποιήθηκαν σε οχήματα Ford από το 1994, αντικαθιστώντας τα τσιπ μικροελεγκτών EEC IV που σχεδίασε η Intel που είχε χρησιμοποιήσει η Ford από το 1983.

Ιστορικό συστημάτων ελεγχόμενου από υπολογιστή αυτοκινήτου

1968 - Η Volkswagen παρουσιάζει το πρώτο καταναλωτικό όχημα που διατίθεται με υπολογιστή - ένα τρανζίστορ, ηλεκτρονικά ελεγχόμενο σύστημα ψεκασμού καυσίμου.

1969 - Η Ford παρουσιάζει το πρώτο αντιολισθητικό σύστημα που ελέγχεται από υπολογιστή.

1971 - Η General Motors παρουσιάζει το πρώτο υπολογιστή που ελέγχεται εκπομπής τους.

1973 - Όλα τα μοντέλα της Chrysler έρχονται τώρα με ηλεκτρονικό έλεγχο κινητήρα (EEC).

1975- Η Ford παρουσιάζει το πρώτο της σύστημα EEC-1 χρησιμοποιώντας μικροεπεξεργαστή Toshiba TLCS-12 12-bit.

1976 - Η General Motors και η Motorola συνεργάζονται για να δημιουργήσουν έναν προσαρμοσμένο μικροϋπολογιστή για χρήση στα οχήματά τους.

1977 - Η Oldsmobile εισάγει έναν ψηφιακό υπολογιστή για τον έλεγχο της χρονικής ανάφλεξης στο μοντέλο Toronado.

1978 - Η Cadillac εισάγει έναν ηλεκτρονικό υπολογιστή «ταξιδιού», εκεί το μοντέλο της Σεβίλλης, που τροφοδοτείται από έναν προσαρμοσμένο μικροεπεξεργαστή Motorola 6802 .

1978- Η Mercedes-Benz και η BOSCH εισάγουν το πρώτο στον κόσμο σύστημα αντιμπλοκαρίσματος (ABS).

1979 - Η ECU της Ford Electronics χρησιμοποιεί μια προσαρμοσμένη έκδοση του 8049 8-bit μικροελεγκτή Intel, Intel 80A49H.

1980 - Η Delphi Automotive Systems Corp. ξεκινά την παραγωγή μονάδων εκπομπής και μονάδων ελέγχου .

1981 - Όλα τα οχήματα της General Motors θα έρχονταν τώρα με ECM βασισμένο στη Motorola 6802 με έλεγχο εκπομπών ρύπων.

1983 - Αρχίζουν να χρησιμοποιούνται τα τσιπ μικροελεγκτών αυτοκινήτου 8061 της Intel, που έχουν σχεδιαστεί ειδικά για τα αυτοκίνητα της Ford

1986 - Η νεοσυσταθείσα Hypertech, παρουσιάζει το πρώτο τους "Power Chip", επιτρέποντας στους ιδιοκτήτες αυτοκινήτων να βελτιώσουν την απόδοση του κινητήρα αναβαθμίζοντας το τσιπ PROM εκεί σε ECU.

1986 - Το «Navlab 1» του Πανεπιστημίου Carnegie Mellon γίνεται το πρώτο μηχανογραφημένο αυτόνομο, αυτόνομο αυτοκίνητο.

1986 - Η Chrysler εισάγει ενότητες πολλαπλών συρμάτων επικοινωνίας με τσιπ που παρέχονται από τον Harris Semiconductor.

1987 - Τα πρώτα τσιπ μικροελεγκτή αυτοκινήτου που παράγονται σύμφωνα με τα πρότυπα λεωφορείων CAN οχημάτων από την Intel και τη Philips Semiconductor .

1991 - Η Ford και η Motorola συνεργάζονται για τον σχεδιασμό και την παραγωγή μικροελεγκτών PTEC κινητήρων και μετάδοσης.

2000 - Η Ford Microelectronics Inc. (FMI) εξαγοράζεται από την Intel Corp.

2009 - Η Google ξεκινά το νέο της πρόγραμμα αυτοκινούμενων αυτοκινήτων, ένα τροποποιημένο Toyota Prius, χρησιμοποιώντας επεξεργαστές Intel.

2012 - Η Google λαμβάνει την πρώτη άδεια που εκδόθηκε στις Ηνωμένες Πολιτείες για αυτόνομο αυτοκίνητο να λειτουργεί σε δημόσιους δρόμους στην πολιτεία της Νεβάδα.

2014 - Παρουσιάστηκε το πρώτο εμπορικά διαθέσιμο αυτόνομο όχημα - Το λεωφορείο Navia.

2015-Το "Freight-liner Inspiration" της Daimler γίνεται το πρώτο αυτόνομο, ημιαυτόνομο, εμπορικό ημφορτηγό.

2016 - Η Tesla ξεκινά να χρησιμοποιεί υπολογιστικούς πίνακες της Nvidia Drive PX 2 με τσιπ Tegra X2 SoC που τροφοδοτούν το σύστημα αυτόματου πιλότου. Το A8 της Audi γίνεται το πρώτο sedan παραγωγής που προσφέρεται με τεχνολογία αυτόνομης οδήγησης επιπέδου 3.

2017-Η Waymo και η Intel ανακοινώνουν συνεργασία για την ανάπτυξη αυτοκινούμενων αυτοκινήτων με επεξεργαστές Intel Xeon και προγραμματιζόμενα τσιπ (FPGA) στο νέο στόλο των αυτόματων μίνι βαν υβριδικών Chrysler Pacifica Hybrid .

2017 - Η Toyota ανακοινώνει ότι θα ξεκινήσει να χρησιμοποιεί τσιπ Nvidia GPU για να τροφοδοτήσει το AI στα αυτοκινούμενα αυτοκίνητά τους.

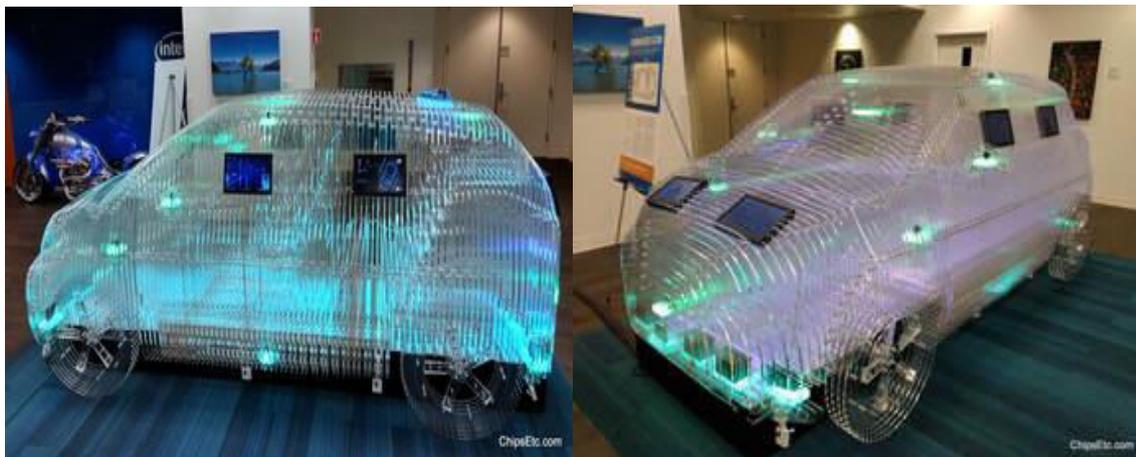
2017 - Η Tesla "Semi" παρουσίασε το πρώτο τους μοντέλο από όλα τα ηλεκτρικά, αυτόνομα αυτοκινούμενα φορτηγά.

2018 - Η Mobileye της Intel παρουσιάζει τον επεξεργαστή EyeQ5, ένα τσιπ SoC σχεδιασμένο για πλήρως αυτόνομα οχήματα.

2019 - Η Tesla ξεκινά την αποστολή αυτοκινήτων Model S και X με τον νέο τους Full Self - Driving Computer (FSD), ο οποίος τροφοδοτείται από προσαρμοσμένους μικροεπεξεργαστές σχεδιασμένους από την Tesla, κατασκευασμένους από τη Samsung.

2019 - Η Volvo ανακοινώνει ότι θα αρχίσει να χρησιμοποιεί υπολογιστικούς πίνακες της Nvidia Drive AGX Pegasus στα αυτοκινούμενα αυτοκίνητά τους.

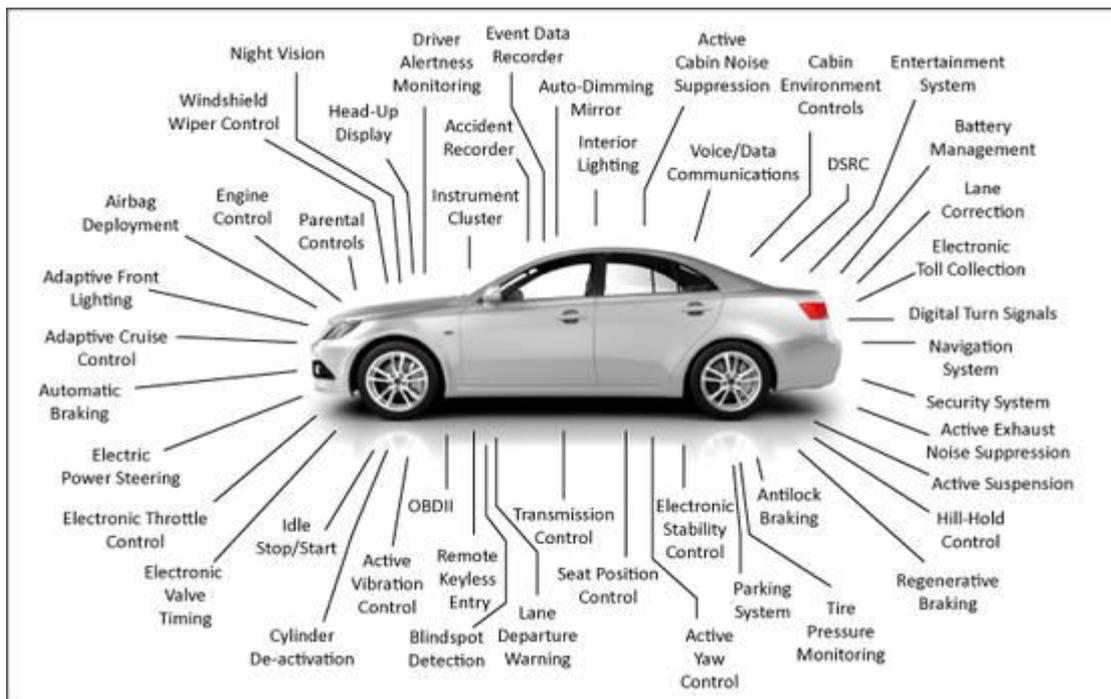
Το αυτόνομο όχημα Mobileye της Intel δημιουργήθηκε για το Consumer Electronics Show 2018



Εικόνα 1.6 & 1.7: Intel Mobileye διαφανής αυτόνομη οθόνη τεχνολογίας οχημάτων (2018)



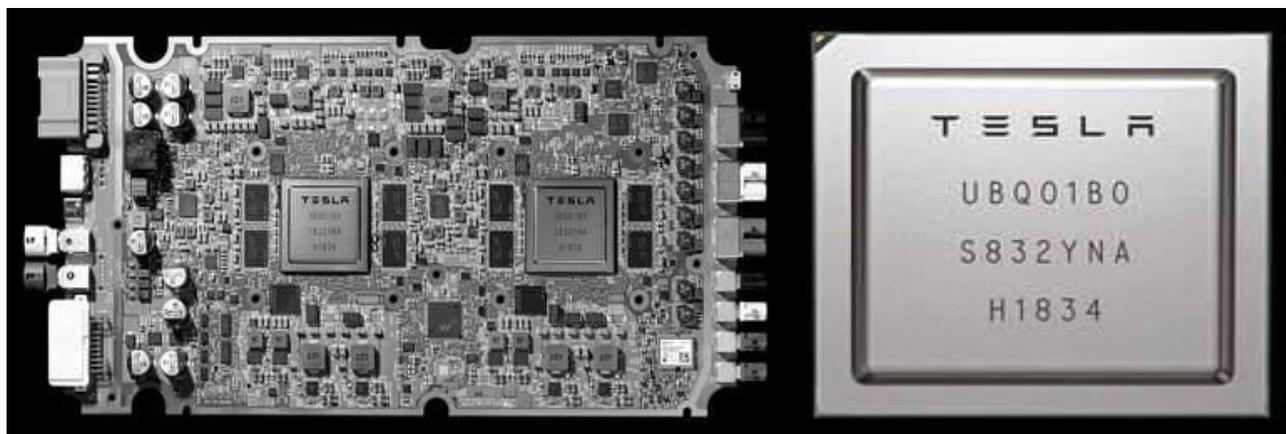
Εικόνα 1.8: Πρωτότυπο αυτοκινούμενο αυτοκίνητο Waymo "Firefly" της Google, Μουσείο Ιστορίας Υπολογιστών, Mountain View CA. (2018)



Εικόνα 1.9: Αισθητήρες-Συστήματα Ελέγχου

Σήμερα, ένα αυτοκίνητο μπορεί να έχει πάνω από 50 συστήματα υπολογιστών που παρακολουθούν και ελέγχουν τα πάντα, από το χειρισμό της οδήγησης, μέχρι τα συστήματα ψυχαγωγίας και επικοινωνίας επί του σκάφους.

Πολυάριθμες αυτοκινητοβιομηχανίες, συμπεριλαμβανομένων των Ford, GM, Tesla, VW & BMW, εργάζονται σε αυτόνομα αυτόνομα αυτοκίνητα, τα οποία θα προσθέσουν μόνο στον αυξανόμενο αριθμό υπολογιστών που βρίσκονται μέσα σε ένα αυτοκίνητο. Εταιρείες τεχνολογίας που εμπλέκονται στην ανάπτυξη των συστημάτων πληροφορικής για αυτά τα αυτόνομα αυτοκίνητα περιλαμβάνουν, Tesla, η AMD , η Ford, Lyft, Google /Αλφάβητο (Waymo), Nvidia και Intel . Τρέχουσα αυτοκινήτων προμηθευτές ημιαγωγών περιλαμβάνουν Freescale / NXP , Renesas, Infineon, STMicroelectronics , Bosch, Texas Instruments , ON Semiconductor, Samsung , MIPS , Qualcomm, Toshiba και Micron Technology .



Εικόνα 1.10 & 1.11 "Υπολογιστής πλήρους αυτόματης οδήγησης" (FSD) της TESLA με επεξεργαστές που κατασκευάζονται από τη Samsung

1.3 Συναφείς Πλατφόρμες - Εφαρμογές

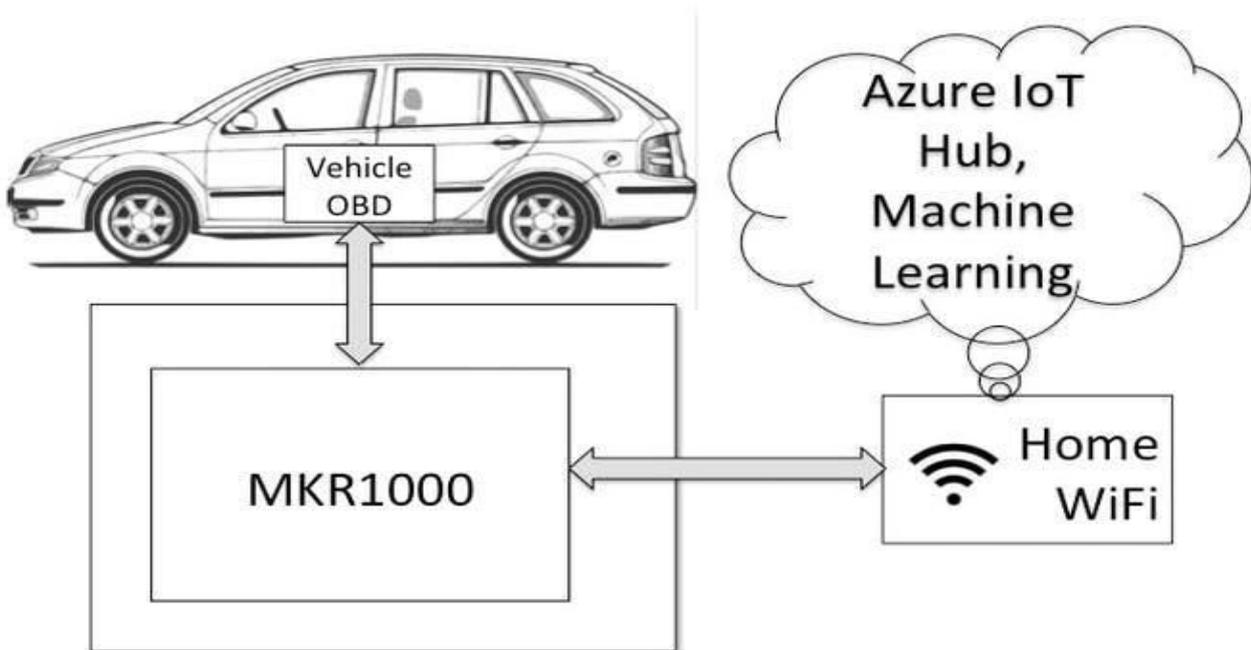
Όπως σε όλα τα στάδια της τεχνολογίας, τα μηχανοκίνητα οχήματα καταλαμβάνουν μια σημαντική θέση στη ζωή μας, αλλά και στο **IoT** έξυπνων αυτοκινήτων. Η ανάπτυξη της τεχνολογίας έχει αρχίσει να κάνει τόσο τον τομέα των αυτοκινήτων όσο και την περιέργεια των χρηστών αυτοκινήτων όλο και πιο ελκυστικό για τα δικτυωμένα οχήματα. Ως αποτέλεσμα του IoT, η έννοια του **έξυπνου συνδεδεμένου αυτοκινήτου** θα είναι ένας από τους μεγαλύτερους στόχους για τη διαμόρφωση των πολλών συσκευών που θα εμφανιστούν στο μέλλον με τρόπο που να μπορεί να λειτουργήσει σε σχέση με τέτοια αυτοκίνητα και να διευκολύνει τη ζωή του κοινού χρήστη.

CarSmart © CC BY-NC-SA

Είναι μια συναφή εργασία βασισμένη σε παρόμοια πλατφόρμα ανοιχτού κώδικα που αφορά την έξυπνη καταγραφή και ανάλυση διαγνωστικών οχημάτων χρησιμοποιώντας OBD-II δεδομένα με βάση το σύννεφο Azure IoT Hub, Stream Analytics και Machine Learning.

Σε μια τέτοια ανάλυση βασίζεται η ανίχνευση ζητημάτων συντήρησης, σε αυτό το πρωτότυπο σύστημα παρακολουθείται η θερμοκρασία ψυκτικού του κινητήρα και αναζητούνται ανωμαλίες. Ο αλγόριθμος μηχανικής μάθησης μαθαίνει τη φυσιολογική συμπεριφορά θερμοκρασίας και στη συνέχεια ταξινομεί τις εκτός φυσιολογικών συνθηκών.

Για πιο πλούσια και ακριβή μοντελοποίηση, συμπεριλαμβάνονται πληροφορίες μηνών, επιτρέποντας στο σύστημα να μάθει εποχιακές αλλαγές στη συμπεριφορά.



Εικόνα 1.12: CarSmart Διάγραμμα

Το επίκεντρο αυτού του έργου είναι η μετακίνηση της νοημοσύνης στο Cloud, οπότε η συσκευή στο όχημα είναι ένας απλός καταγραφέας δεδομένων και δεν κάνει δική της ανάλυση δεδομένων.

IoT4Car © CC BY

Άλλη μια συναφή εργασία με το MKR WiFi 1000 να παίζει κομβικό ρολό καθώς μιλάει με ένα αυτοκίνητο μέσω διεπαφής OBD-II και ανεβάζει τα δεδομένα στο IoT cloud για παρακολούθηση και μετά επεξεργασία σε πραγματικό χρόνο.

Όταν οδηγούμε το όχημά μας, ρίχνοντας μια ματιά στο ταμπλό, βρισκόμαστε μπροστά από κάποιες μετρήσεις, μπορούμε να τις συλλέξουμε και να κάνετε κάποια ανάλυση. Αυτά τα δεδομένα μπορεί να περιέχουν κρυμμένους θησαυρούς. Για ιδιώτες, μπορεί να αντικατοπτρίζει τις οδηγικές συνήθειες, μπορεί να πει την ταχύτητά, τον μέσο όρο καταναλώσης, πόσα ήταν τα φανάρια ακόμα και τον χρόνο αναμονής σε κάθε διασταύρωση. Για τις εταιρείες, αυτά τα δεδομένα είναι κρίσιμα για την παρακολούθηση σε πραγματικό χρόνο στη διαχείριση του στόλου. Η κατάσταση του οχήματος, η κατανομή του φορτίου εργασίας, η απόδοση της βενζίνης, ακόμη και η θέση του οχήματος, μπορούν να επαναφερθούν σε ένα κεντρικό σύστημα ελέγχου μέσω cloud.



Εικόνα 1.13: IoT4Car Διάγραμμα

Οι εφαρμογές και οι υλοποιήσεις με διαφορετική οπτική κάθε φορά και σίγουρα διαφορετικό στόχο είναι παρά πολλές, και αυτό διότι το αυτοκίνητο είναι μέρος της ζωής μας πλέον. Η ιδέα του IoT δίνει συνεχώς τροφή για καινοτόμες προσεγγίσεις αυτού του θέματος και αναμφίβολα πολλές από αυτές θα αφομοιωθούν στην αυτοκινητοβιομηχανία .

Κεφάλαιο 2^ο - Ενσωματωμένα & Μικροελεγκτές

2.1 Ενσωματωμένα Συστήματα



Εικόνα 2.1: Διάταξη Ενσωματωμένου Συστήματος

Ένα ενσωματωμένο σύστημα είναι σύστημα υπολογιστή με ειδική λειτουργία μέσα σε μεγαλύτερο μηχανικό ή ηλεκτρικό σύστημα, συχνά με περιορισμούς υπολογισμών σε πραγματικό χρόνο. Είναι ενσωματωμένο ως μέρος μιας πλήρους συσκευής που συχνά περιλαμβάνει υλικό και μηχανικά μέρη. Τα ενσωματωμένα συστήματα ελέγχουν πολλές συσκευές σε κοινή χρήση σήμερα. Το 98% όλων των μικροεπεξεργαστών κατασκευάζονται ως συστατικά ενσωματωμένων συστημάτων.

Παραδείγματα

Παραδείγματα ιδιοτήτων των τυπικών ενσωματωμένων υπολογιστών σε σύγκριση με τους ομολόγους τους γενικής χρήσης είναι η χαμηλή κατανάλωση ενέργειας, το μικρό μέγεθος, τα εύρωστα λειτουργικά εύρη και το χαμηλό κόστος ανά μονάδα. Αυτό έρχεται στην τιμή των περιορισμένων πόρων επεξεργασίας, γεγονός που τους καθιστά πολύ πιο δύσκολο να προγραμματιστούν και να αλληλεπιδρούν. Ωστόσο, με την οικοδόμηση μηχανισμών πληροφοριών πάνω από το υλικό, εκμεταλλευόμενοι τους πιθανούς υπάρχοντες αισθητήρες και την ύπαρξη ενός δικτύου ενσωματωμένων μονάδων, μπορεί κανείς να διαχειριστεί άριστα τους διαθέσιμους πόρους σε επίπεδο μονάδας και δικτύου, καθώς και να παράσχει αυξημένες λειτουργίες πολύ πιο πέρα από τις διαθέσιμες. Για παράδειγμα, μπορούν να σχεδιαστούν έξυπνες τεχνικές για τη διαχείριση της κατανάλωσης ενέργειας των ενσωματωμένων συστημάτων.

Τα σύγχρονα ενσωματωμένα συστήματα συχνά βασίζονται σε μικροελεγκτές (δηλ. κεντρικές μονάδες επεξεργασίας με ενσωματωμένη μνήμη ή περιφερειακές διεπαφές), αλλά οι συνηθισμένοι μικροεπεξεργαστές (χρησιμοποιώντας εξωτερικά τσιπ για μνήμη και περιφερειακά κυκλώματα διασύνδεσης) είναι επίσης συνηθισμένοι, ειδικά σε πιο σύνθετα συστήματα. Και στις δύο περιπτώσεις, οι χρησιμοποιούμενοι επεξεργαστές μπορεί να είναι τύποι που κυμαίνονται από γενικούς σκοπούς έως αυτούς που εξειδικεύονται σε συγκεκριμένες κατηγορίες υπολογισμών ή ακόμη και προσαρμοσμένοι για την εφαρμογή

που βρίσκεται στο χέρι. Μια κοινή πρότυπη κατηγορία αποκλειστικών επεξεργαστών είναι ο επεξεργαστής ψηφιακών σημάτων.

Δεδομένου ότι το ενσωματωμένο σύστημα είναι αφιερωμένο σε συγκεκριμένες εργασίες, οι μηχανικοί σχεδιασμού μπορούν να το βελτιστοποιήσουν για να μειώσουν το μέγεθος και το κόστος του προϊόντος και να αυξήσουν την αξιοπιστία και την απόδοση. Ορισμένα ενσωματωμένα συστήματα είναι μαζικής παραγωγής, επωφελούμενα από οικονομίες κλίμακας.

Είδη

Τα ενσωματωμένα συστήματα κυμαίνονται από φορητές συσκευές όπως ψηφιακά ρολόγια, ηλεκτρικές συσκευές, κυκλώματα τηλεπικοινωνιών, συστήματα συλλογής δεδομένων, εφαρμογές δικτύων, συστήματα αυτοματισμών έως μεγάλες σταθερές εγκαταστάσεις και πολύπλοκα συστήματα, όπως τα υβριδικά οχήματα, η μαγνητική τομογραφία και η αεροηλεκτρονική. Η πολυπλοκότητα ποικίλλει από χαμηλή, με ένα ενιαίο μικροελεγκτή τσιπ, έως πολύ υψηλή με πολλαπλές μονάδες, περιφερειακά και δίκτυα τοποθετημένα μέσα σε ένα μεγάλο πλαίσιο ή περίβλημα. Καθώς επίσης και το τυπικό μέγεθος λέξης μπορεί να κυμαίνεται σε 8-bit, 16-bit και 32-bit.

Χαρακτηριστικά

Τα ενσωματωμένα συστήματα έχουν σχεδιαστεί για να κάνουν κάποια συγκεκριμένη εργασία, αντί να είναι ένας υπολογιστής γενικής χρήσης για πολλαπλές εργασίες. Ορισμένα έχουν επίσης περιορισμούς απόδοσης σε πραγματικό χρόνο που πρέπει να πληρούνται, για λόγους όπως η ασφάλεια και η χρηστικότητα. Άλλοι μπορεί να έχουν χαμηλές ή καθόλου απαιτήσεις απόδοσης, επιτρέποντας την απλοποίηση του υλικού του συστήματος για τη μείωση του κόστους. Τα ενσωματωμένα συστήματα δεν είναι πάντα αυτόνομες συσκευές.

Πολλά ενσωματωμένα συστήματα αποτελούνται από μικρά κομμάτια μέσα σε μια μεγαλύτερη συσκευή που εξυπηρετεί έναν γενικότερο σκοπό. Για παράδειγμα, η κιθάρα Gibson Robot διαθέτει ενσωματωμένο σύστημα συντονισμού των χορδών, αλλά ο γενικός σκοπός της κιθάρας ρομπότ είναι φυσικά να παίζει μουσική. Ομοίως, ένα ενσωματωμένο σύστημα σε ένα αυτοκίνητο παρέχει μια συγκεκριμένη λειτουργία ως υποσύστημα του ίδιου του αυτοκινήτου.

2.2 Εισαγωγή στους Μικροελεγκτές

Έναν ορισμό που θα μπορούσαμε να δώσουμε για τους μικροελεγκτές είναι ο εξής:

Μικροελεγκτής είναι ένα προγραμματιζόμενο ολοκληρωμένο κύκλωμα το οποίο διαθέτει επεξεργαστή, μνήμη, διάφορα περιφερειακά κυκλώματα καθώς επίσης και θύρες εισόδου/εξόδου για επικοινωνία με εξωτερικές συσκευές. Θα μπορούσε να παρομοιαστεί με έναν μικροϋπολογιστή. Όπως ακριβώς ένας μικροϋπολογιστής έχει επεξεργαστή, μνήμη, περιφερειακές συσκευές και εκτελεί προγράμματα έτσι κι ένας μικροελεγκτής διαθέτει τα παραπάνω χαρακτηριστικά και μάλιστα ολοκληρωμένα σε ένα μόνο chip. Το πρόγραμμα που εκτελεί ο μικροελεγκτής αποθηκεύεται μόνιμα στη μνήμη προγράμματος

Εφαρμογές μικροελεγκτών

- Οι μικροελεγκτές βρίσκουν εφαρμογή στα παρακάτω πεδία:
- Σε συστήματα αυτοματισμών
- Σε κυκλώματα τηλεπικοινωνιών
- Στις ηλεκτρονικές συσκευές
- Στις ηλεκτρικές συσκευές
- Σε συστήματα τηλεματικής
- Σε συστήματα συλλογής δεδομένων (Data Acquisition)
- Σε εφαρμογές ηλεκτρονικών ισχύος
- Σε συστήματα διασύνδεσης
- Σε εφαρμογές δικτύων

Γενικότερα οι μικροελεγκτές χρησιμοποιούνται οπουδήποτε απαιτείται έλεγχος συστημάτων.

Όταν λέμε ενσωματωμένα συστήματα (Embedded Systems) εννοούμε συστήματα τα οποία είναι βασισμένα σε μικροεπεξεργαστή (ή επίσης FPGA ή DSP)

Μερικοί από τους γνωστότερους κατασκευαστές μικροελεγκτών είναι οι

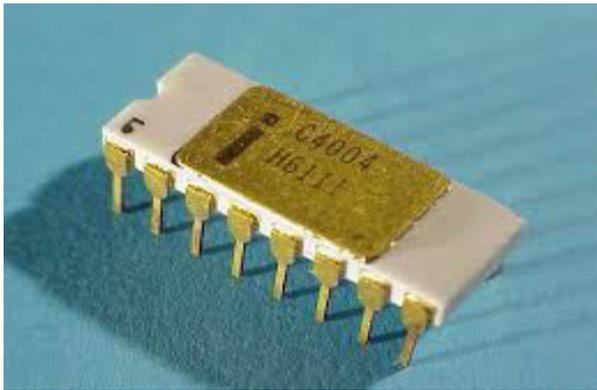
- ARM (δεν κατασκευάζει αλλά παραχωρεί δικαιώματα χρήσης του πυρήνα)
- Atmel
- Epson
- Freescale Semiconductor (πρώην Motorola)
- Hitachi
- Maxim (μετά την εξαγορά της Dallas)
- Microchip
- NEC
- Toshiba
- Texas Instruments

2.3 Ιστορική Αναδρομή Μικροεπεξεργαστών - Μικροελεγκτών

Στην επιθυμία των κατασκευαστικών για δημιουργία συστημάτων, τα οποία θα έχουν περισσότερες δυνατότητες σε μικρότερο μέγεθος, καταλήξαμε στην καινοτομία της ενσωμάτωσης των λειτουργιών ενός υπολογιστή σε ένα (ή λίγα) ολοκληρωμένο κύκλωμα. Η ανάπτυξη της τεχνολογίας, η οποία χρησιμοποιήθηκε για τη δημιουργία των κυκλωμάτων οδήγησε στην ανάπτυξη του επεξεργαστή. Στους μικροεπεξεργαστές οφείλεται η δημιουργία μερικών από τις μεγαλύτερες καινοτομίες στα συστήματα υπολογιστών.

Παραδείγματα αυτών των καινοτομιών αποτελούν:

- Ενσωματωμένοι μικροελεγκτές (Embedded Microcontrollers)
- Προσωπικοί υπολογιστές.
- Σύγχρονοι σταθμοί εργασίας.
- Κινητές συσκευές.
- Servers
- Ευρείας κλίμακας δίκτυα υπολογιστών



Εικόνα 2.2: Μικροεπεξεργαστής Intel 4004



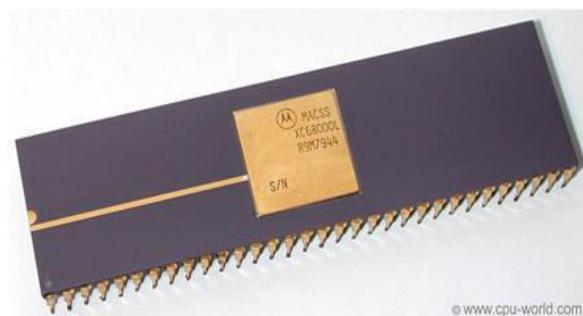
Εικόνα 2.3: Μικροεπεξεργαστής TMS1000

Χαρακτηριστικά ο πρώτος μικροεπεξεργαστής ήταν ο 4-bit Intel 4004 ο οποίος κυκλοφόρησε τον 1971. Ήταν ένας 4-bit επεξεργαστής αποτελούμενος από 2.300 τρανζίστορ, ο οποίος εκτελούσε 60.000 πράξεις το δευτερόλεπτο και σκοπός της δημιουργίας του ήταν η κατασκευή αριθμομηχανών. Ακολούθησε ο Intel 8008 (1972), ο οποίος είναι ο πρώτος 8-bit επεξεργαστής. Και οι δύο όμως επεξεργαστές χρειαζόταν πολλά εξωτερικά chip ώστε να δημιουργήσουν ένα ολοκληρωμένο σύστημα, κάτι που ανέβασε το κόστος και το καθιστούσε αδύνατο.

Οι Gary Boone και Michael Cochran δημιούργησαν τον πρώτο μικροελεγκτή το 1971. Το αποτέλεσμα της δουλειάς τους ήταν ο TMS 1000 ο οποίος ήταν διαθέσιμος από το 1974. Συνδύαζε Read-only μνήμη, Read/Write μνήμη, επεξεργαστή και clock σε ένα chip και στόχευε στα ενσωματωμένα συστήματα.



Εικόνα 2.4: Μικροεπεξεργαστής Z80



Εικόνα 2.5: Μικροεπεξεργαστής 6800

Το 1974 εμφανίζεται ο Intel 8080, μία εξέλιξη του Intel 8008. Πρόκειται για έναν επεξεργαστή 8bit με συχνότητα λειτουργίας 2MHz. Την ίδια χρονική περίοδο η Motorola παρουσίασε τον 6800, ο οποίος βρήκε κατά κύριο λόγο χρήση σε βιομηχανικές συσκευές ελέγχου. Αποτελούνταν από 4000 τρανζίστορ με σήμα χρονισμού στα 1 ή 2 MHz. Ως σύμβολο ανταγωνισμού για τον επεξεργαστή Intel 8080, η MOS Technology κατασκεύασε τον MOS 6502 ο οποίος χρησιμοποιήθηκε σε ευρέως διαδεδομένα micros όπως το Apple II, Commodore και Acorn.

Το 1976 η Zilog δημιουργεί τον Z80. Πρόκειται για έναν 8bit μικροεπεξεργαστή βασισμένο στον 8080 του οποίου η γλώσσα μηχανής αποτελεί υπερσύνολο αυτής της Intel 8080 με σήμα χρονισμού στα 3.5 MHz και μπορούσε να δει 64Kbytes μνήμης. Βασικό του χαρακτηριστικό ήταν το χαμηλό κόστος και ο συνυπολογισμός των στοιχείων του κυκλώματος (το οποίο θα έπρεπε να γίνει σε ξεχωριστό chip).



Εικόνα 2.5: Μικροεπεξεργαστής Intel 386

Οι πρώτοι 16bit μικροεπεξεργαστές εμφανίστηκαν το 1978. Ο Intel 8086/8088 απαρτιζόταν από 29.000 τρανζίστορ και είχε συχνότητα λειτουργίας τα 10MHz. Η Motorola σε αντιστοιχία παρουσιάζει τον 68000 με συχνότητα λειτουργίας τα 8MHz. Το χαμηλό κόστος, η υψηλή ταχύτητα και ο μεγάλος αποθηκευτικός χώρος τον έκανε τον δημοφιλέστερο επεξεργαστή με αποτέλεσμα να αποτελέσει πρώτη επιλογή για την Apple στα μοντέλα Lisa και Macintosh.

Στα μέσα της δεκαετίας του 90 εμφανίζονται οι πρώτοι 32bit μικροεπεξεργαστές από την Intel και την Motorola, ο Intel 80386 και ο 68020 αντίστοιχα. Έγινε ιδιαίτερα δημοφιλής στην αγορά microcomputer Unix, καθώς επίσης και πολλές επιχειρήσεις άρχισαν να παράγουν συστήματα Desktop.



Εικόνα 2.6: Μικροεπεξεργαστής Intel Pentium 1



Εικόνα 2.7: Μικροεπεξεργαστής Intel Pentium 4

Το 1992 θα γίνει μία επανάσταση στον σχεδιασμό με την παρουσίαση του Intel Pentium. Με το συγκεκριμένο μοντέλο εισήχθη η λογική της Superscalar επεξεργασίας έναντι της έως τότε x86 σειράς. Αποτελούνταν από 3.100.000 τρανζίστορ και η συχνότητα λειτουργίας του τα 166MHz. Ακολούθησαν οι Pentium 2, 3 και το 2000 παρουσιάστηκε ο Intel Pentium 4, που αποτέλεσε σταθμό με την σειρά του καθώς εισήγαγε την έννοια της Netburst αρχιτεκτονικής. Ο Intel Pentium 4 ήταν κατασκευασμένος πλέον με 42.000.000 τρανζίστορ και ο χρονισμός ανέρχονταν πλέον από τα 1.4 στα 3.8GHz.

Κατά τη διάρκεια του έτους 2005 η Intel παρουσιάζει τον Pentium D, τον πρώτο Dual-Core επεξεργαστή, με πρώτο μοντέλο παρουσίασης να είναι ο Pentium Extreme.



Εικόνα 2.8: Μικροεπεξεργαστής Intel Pentium D

Παράλληλα, σχεδιάζει τον Xeon 5300, ο οποίος είναι ο πρώτος τετραπλού πυρήνα επεξεργαστής (αποτελούνταν από δύο Dual-Core), με στόχο την χρήση τους σε Workstations και Servers.

Στη συνεχώς, τρομακτικά γρήγορη εξέλιξη της τεχνολογίας των επεξεργαστών όπως παρατηρούμε, το 2008 έρχεται η Qualcomm[5]. Πρόκειται για μία εταιρεία ασύρματης τεχνολογίας η οποία ξεκινά να παράγει υψηλής απόδοσης επεξεργαστές για Smart-Phones βασισμένους σε ARM αρχιτεκτονική. Είναι κατασκευασμένοι από 200.000.000 chips και χρονισμένοι στα 1GHz.

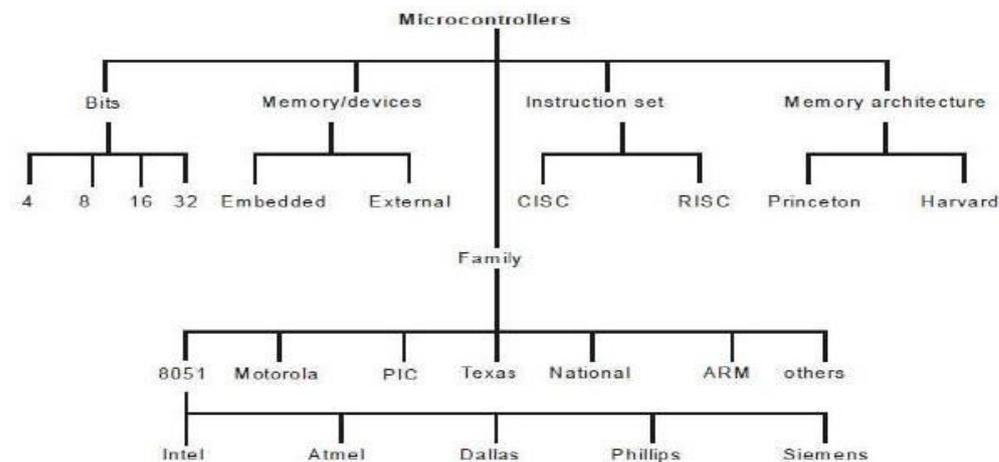
Μέχρι σήμερα η Intel έχει κατασκευάσει επεξεργαστές μέχρι και 8 πυρήνες (Core i3, i5, i7) βασισμένοι στην αρχιτεκτονική Sandy Bridge, οι οποίοι αποτελούνται από 1.45 δισεκατομμύρια τρανζίστορ, ενώ η ARM ανακοίνωσε την ARMv8 αρχιτεκτονική με την οποία μπορούμε να έχουμε μέχρι και 128 πυρήνες μέσα σε έναν επεξεργαστή.

Εικόνα 2.9: Μικροεπεξεργαστές Intel i3,i5,i7



2.4 Κατηγορίες Μικροελεγκτών

Types of Microcontrollers



Εικόνα 2.10: Τύποι Μικροελεγκτών

1. Μικροελεγκτές (καμιά φορά **4-bit** αλλά συνήθως **8-bit**) πολύ χαμηλού κόστους, γενικής χρήσης, με πολύ μικρό αριθμό ακροδεκτών (ακόμη και λιγότερους από 8). Σχεδιάζονται με έμφαση στη χαμηλή κατανάλωση ισχύος και την αυτάρκεια, ώστε να χρειάζονται ελάχιστα ή και καθόλου εξωτερικά εξαρτήματα για να μην μπορεί να αντιγραφεί εύκολα το εσωτερικό λογισμικό τους. Απουσιάζει η δυνατότητα επέκτασης της μνήμης τους. Μερικά μοντέλα είναι ευρέως γνωστά στους ερασιτέχνες ηλεκτρονικούς, όπως για παράδειγμα οι περισσότεροι μικροελεγκτές των σειρών PIC(Microchip), AVR (Atmel) και 8051 (Intel, Atmel, Dallas κ.α.).
2. 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 ίδιας της μνήμης τους για μεγαλύτερα project.
3. 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

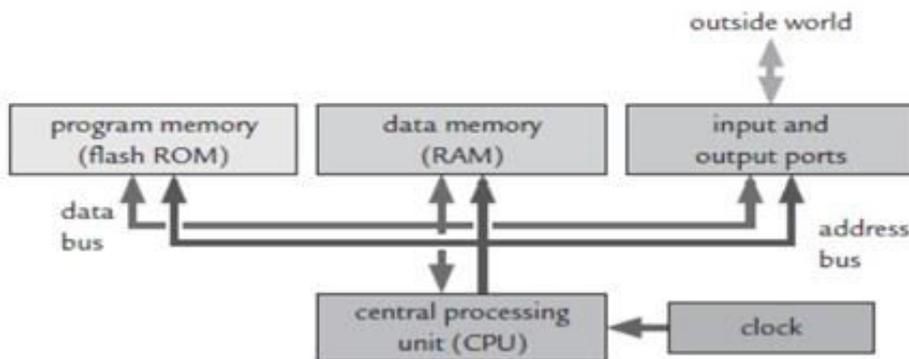
άλλο κατασκευαστή. Για παράδειγμα μεταξύ των μικροελεγκτών τύπου ARM ή MIPS, το σύνολο των βασικών εντολών που αναγνωρίζει η ALU είναι ακριβώς το ίδιο, μειώνοντας έτσι τις μεγάλες αλλαγές στο λογισμικό, όταν στο μέλλον ο πελάτης υιοθετήσει ένα μικροελεγκτή άλλου κατασκευαστή (αρκεί φυσικά να υποστηρίζει κι αυτός το σύνολο εντολών ARM ή MIPS, αντίστοιχα).

4. 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 επιλογή ενός μικροελεγκτή για μία διεργασία γίνεται με τον καθορισμό των προδιαγραφών και στην συνέχεια επιλέγεται ο φθηνότερος που να μπορεί να τις ικανοποιήσει. Στην συγκεκριμένη εργασία χρησιμοποιείται ο μικροελεγκτής ATmega328 που διαθέτει το Arduino UNO και ο ATmega2560 που διαθέτει το Arduino MEGA 2560.

2.5 Βασικά Χαρακτηριστικά Μικροελεγκτή

Anatomy of a Typical Small Microcontroller



Εικόνα 2.11: Διάγραμμα Τυπικού Μικροελεγκτή

1. Η Κεντρική Μονάδα Επεξεργασίας – CPU το κεντρικό εξάρτημα το οποίο επεξεργάζεται δεδομένα και σε ένα μικροελεγκτή αποτελείται από ένα μόνο ολοκληρωμένο κύκλωμα. Τα στοιχεία της είναι:
 - a. Μονάδα Αριθμητικής Λογικής (Arithmetic Logic Unit(ALU)) η οποία είναι υπεύθυνη για τους υπολογισμούς.
 - b. Καταχωρητές Γενικού – Ειδικού σκοπού.

- c. Οι πρώτοι είναι υπεύθυνοι για την αποθήκευση των δεδομένων ενώ οι δεύτεροι για συγκεκριμένες λειτουργίες.
 - d. Instruction Decoder για τον έλεγχο της Κεντρικής Μονάδας Επεξεργασίας.
2. Μνήμη Προγράμματος (ROM) η μνήμη που αφού σταματήσει η τροφοδοσία του κυκλώματος αποθηκεύει τα δεδομένα της.
 3. Μνήμη Δεδομένων (RAM) μνήμη Δεδομένων ή Μνήμη Τυχαίας Προσπέλασης είναι η μνήμη που χρησιμοποιείται για την προσωρινή αποθήκευση δεδομένων καθώς και για την πρόσβαση στα αποθηκευμένα δεδομένα. Με την διακοπή της τροφοδοσίας τα δεδομένα της χάνονται.
 4. Κύκλωμα συνδετικής λογικής (glue Logic) για τη σύνδεση των εξωτερικών μνημών και άλλων περιφερειακών παράλληλης σύνδεσης στην αρτηρία δεδομένων (Buses) του επεξεργαστή.
 5. Θύρες Εισόδου – Εξόδου (I/O Ports) θύρες με τις οποίες ο μικροελεγκτής αλληλοεπιδρά με το περιβάλλον.
 6. Ρολόι (CLOCK) υπεύθυνο για τον χρονισμό του μικροελεγκτή.

Τα παραπάνω στοιχεία είναι κατά κόρον ίδια σε κάθε μικροελεγκτή. Φυσικά ο τρόπος υλοποίησης του συστήματος μπορεί, και συνήθως διαφέρει. Εκεί που οι μικροελεγκτές παρουσιάζουν τις μεγαλύτερες διαφορές τους είναι στα περιφερειακά.

Παλαιότερα, τα περιφερειακά χρειαζόταν δικό τους εξοπλισμό με την γοργή τεχνολογική εξέλιξη όμως κατορθώθηκε η ενσωμάτωση τους στο κύκλωμα του επεξεργαστή.

Αναλυτικά τα πιο συνηθισμένα περιφερειακά κομμάτια ενός μικροελεγκτή είναι τα εξής:

1. **Χρονιστές (Timers):** Οι μικροελεγκτές χρησιμοποιούν συνήθως έναν ή περισσότερους χρονιστές λόγω του εύρους των διεργασιών που πραγματοποιούν. Μερικές συνηθισμένες λειτουργίες αυτών είναι η δημιουργία καθυστερήσεων, η μέτρηση διάρκειας γεγονότων, καθώς και η απαρίθμηση γεγονότων. Ακόμα έχουν την δυνατότητα να θέσουν το σύστημα μας σε ενεργεία ή καταστολή (ON/OFF) ανάλογα με τις ανάγκες της διαδικασίας που θα πραγματοποιήσει ο μικροελεγκτής, κάτι το οποίο συντελεί καταλυτικά στην εξοικονόμηση ενέργειας. Η τελευταία δυνατότητα είναι εξαιρετικά σημαντική αφού σε πολλά κυκλώματα οι μικροελεγκτές τροφοδοτούνται με μπαταρίες.
2. **Χρονιστής Watchdog (Watchdog Timer):** Ο συγκεκριμένος χρονιστής, εξασφαλίζει την ομαλότητα του επεξεργαστή μας, σε περίπτωση που το τρέχων πρόγραμμα βρεθεί σε ατέρμον βρόγχο. Στην περίπτωση αυτή ο χρονιστής θέτει σε επανεκκίνηση τον επεξεργαστή.
3. **Διεπαφές επικοινωνίας:** Χρησιμοποιούνται για την ανταλλαγή πληροφοριών με άλλα συστήματα και η γκάμα τους είναι πάρα πολύ μεγάλη. Ενδεικτικά μερικές από αυτές είναι:
 - a. SPI (serial peripheral interface).
 - b. Inter-integrated circuit (IC2 ή IIC).
 - c. Ασύγχρονες διεπαφές όπως RS-232, USB (universal serial bus), Ethernet, CAN (controller area network).

4. **Analog-to-digital Converter:** Η χρήση τους είναι ευρεία, καθώς αυτό που διαβάζουμε από το εξωτερικό περιβάλλον είναι συνεχώς μεταβαλλόμενες και σε δεν είναι ψηφιακές.
5. **Digital-to-analog Converter:** Δεν χρησιμοποιούνται συχνά καθώς μπορούμε να κάνουμε προσομοίωση των περισσότερων αναλογικών εξόδων.

2.6 Πρόσθετες Λειτουργίες Μικροελεγκτή

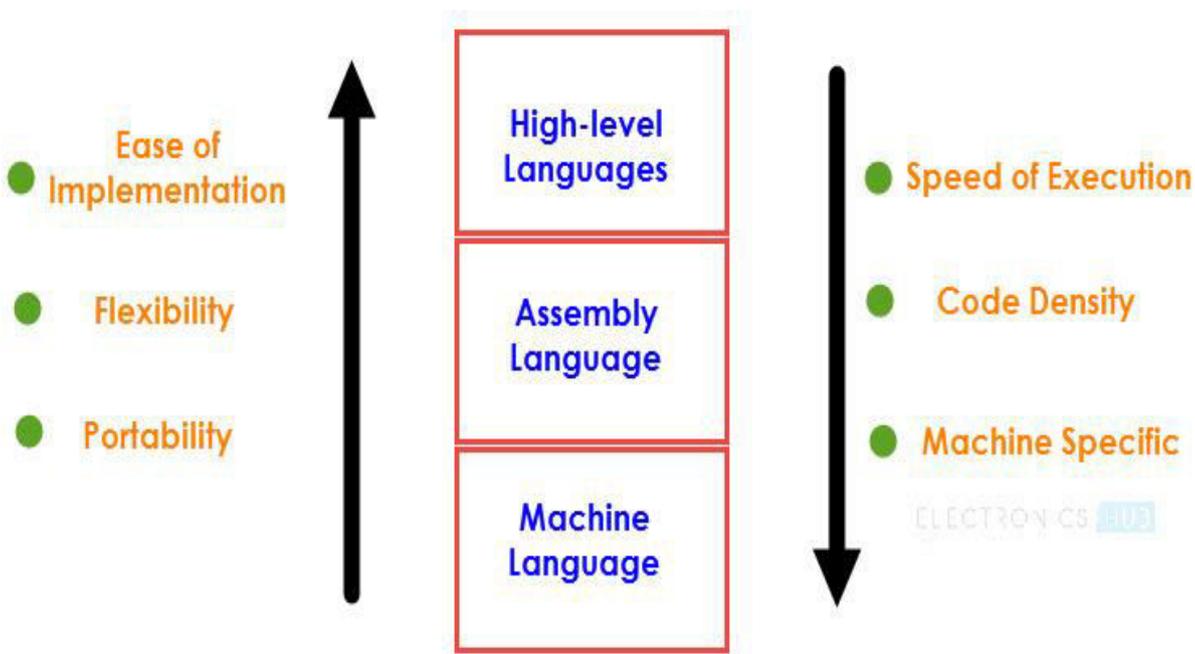
Οι μικροελεγκτές, ανάλογα με την εφαρμογή για την οποία έχουμε σκοπό να τους χρησιμοποιήσουμε μπορούν να περιέχουν περισσότερα εξαρτήματα [21]. Μερικά από αυτά είναι:

- Σύγχρονες σειριακές θύρες επικοινωνίας, όπως είναι το Ethernet (το συνηθέστερο πρότυπο δίκτυο ενσύρματης τοπικής δικτύωσης υπολογιστών).
- Πολλαπλές εισόδους για μετατροπή αναλογικού σήματος σε ψηφιακό (Analog to Digital Converter). Αυτό φυσικά μπορεί να παραλείπεται και να υλοποιηθεί με έναν εξωτερικό ADC όπως ο MCP3008.
- Συστήματα τα οποία υποστηρίζουν την άμεση υποστήριξη Firmware πιο σύνθετων πρωτοκόλλων επικοινωνίας όπως είναι το ADSL και το ISDN.
- Ασύγχρονες σειριακές θύρες επικοινωνίας (UART)
- Μονάδα άμεσης εκτέλεσης πράξεων κινητής υποδιαστολής (FPU – Floating Point Processing Unit) η οποία είναι γρηγορότερη από την ALU του επεξεργαστή. Μία τέτοια μονάδα δίνει στον μικροελεγκτή την δυνατότητα της ψηφιακής επεξεργασίας σήματος (DSP).
- Υποσύστημα προγραμματισμού (τύπου ISP) και διάγνωσης (συνήθως είναι το καθιερωμένο πρότυπο JTAG). Έτσι, είναι δυνατός ο προγραμματισμός της μνήμης προγράμματος χωρίς να προαπαιτείται κάποιο πρόγραμμα υποδοχής. Γι' αυτόν τον λόγο, είναι ιδιαίτερα χρήσιμο στον αρχικό προγραμματισμό, π.χ. κατά τη συναρμολόγηση, ή σε περίπτωση σφάλματος (bug) στο λογισμικό υποδοχής το οποίο να καθιστά αδύνατη την κανονική αναβάθμιση.
- Υποσύστημα προγραμματισμού πάνω στο κύκλωμα (τύπου ISP, βλ. παραπάνω). Χάρη σε αυτό το κύκλωμα, είναι δυνατός ο επαναπρογραμματισμός (αναβάθμιση λογισμικού) της εφαρμογής, συνδέοντας στη συσκευή μια εξωτερική συσκευή προγραμματισμού (συνήθως σε UART RS-232) ή ακόμη και από το διαδίκτυο. Αυτή η δυνατότητα απαιτεί την προϋπαρξη λογισμικού υποδοχής (bootstrap) μέσα στη μνήμη προγράμματος και επομένως δεν μπορεί να γίνει σε τελείως άδεια μνήμη προγράμματος.

2.7 Προγραμματισμός Μικροελεγκτών

Ο προγραμματισμός τέτοιων συσκευών, όπως οι μικροελεγκτές, γίνεται με συγκεκριμένη σύνταξη εντολών και χρησιμοποιούν δύο καταστάσεις, το 0 και το 1 .

Στην αρχή ο μόνος τρόπος προγραμματισμού των μικροελεγκτών ήταν η assembly, μία γλώσσα χαμηλού επιπέδου που ερχόταν πολύ κοντά στην γλώσσα μηχανής. Η γλώσσα μηχανής χρησιμοποιεί εντολές οι οποίες εκτελούν περιορισμένες διεργασίες και για να εκτελεστεί μία πιο σύνθετη λειτουργία, χρειάζεται ένας μεγάλος αριθμός εντολών (κάτι το οποίο απαιτεί πολύ χρόνο και υποβόσκει μεγάλη πιθανότητα λάθους). Με την εξέλιξη του προγραμματισμού όμως έχουμε πλέον τη δυνατότητα προγραμματισμού ενός μικροελεγκτή σε γλώσσες υψηλού επιπέδου και μετά με την βοήθεια κάποιου μεταγλωττιστή (compiler), μεταφράζεται σε γλώσσα χαμηλού επιπέδου (γλώσσα μηχανής). Αξίζει να παραθέσουμε μερικά πλεονεκτήματα και μειονεκτήματα των δύο ειδών γλωσσών καθώς ο προγραμματισμός αποτελεί ένα πάρα πολύ σημαντικό κομμάτι στη χρήση των μικροελεγκτών.



Εικόνα 2.12: Πλεονεκτήματα Γλωσσών Προγραμματισμού

Πλεονεκτήματα Γλωσσών Χαμηλού Επιπέδου

- Ο Προγραμματιστής έχει τον απόλυτο έλεγχο του μικροελεγκτή.
- Επίτευξη πολύ μεγάλης (σχεδόν απόλυτης) ακρίβειας χρονισμών.
- Δωρεάν διάθεση του assembler από την κατασκευάστρια εταιρεία.

Μειονεκτήματα Γλωσσών Χαμηλού Επιπέδου

- Δυσκολία εκμάθησης της γλώσσας του κάθε μικροελεγκτή.

- Η δημιουργία των προγραμμάτων γίνεται σε συμβολική γλώσσα επομένως είναι πολύ δύσκολη η κατανόηση της λογικής σε περίπτωση που το πρόγραμμα χρειαστεί τροποποίηση ή διόρθωση.
- Είναι πολύ δύσκολη η συνεργασία πολλών προγραμματιστών μαζί στο ίδιο πρόγραμμα.

Πλεονεκτήματα Γλωσσών Υψηλού Επιπέδου

- Ευκολία στην ανάπτυξη σύνθετων προγραμμάτων.
- Ευκολία στην συνεργασία πολλών προγραμματιστών.

Μειονεκτήματα Γλωσσών Υψηλού Επιπέδου

- Μεγάλο κόστος αγοράς του compiler.
- Σε εφαρμογές με κρίσιμους χρονισμούς είναι δυσκολότερη η συγγραφή κώδικα που ανταποκρίνεται στους χρονισμούς αυτούς.
- Σε παλιότερους compilers ο κώδικας μηχανής που παραγόταν δεν ήταν βελτιστοποιημένος με αποτέλεσμα να απαιτείται μικροελεγκτής με πολύ περισσότερη μνήμη.

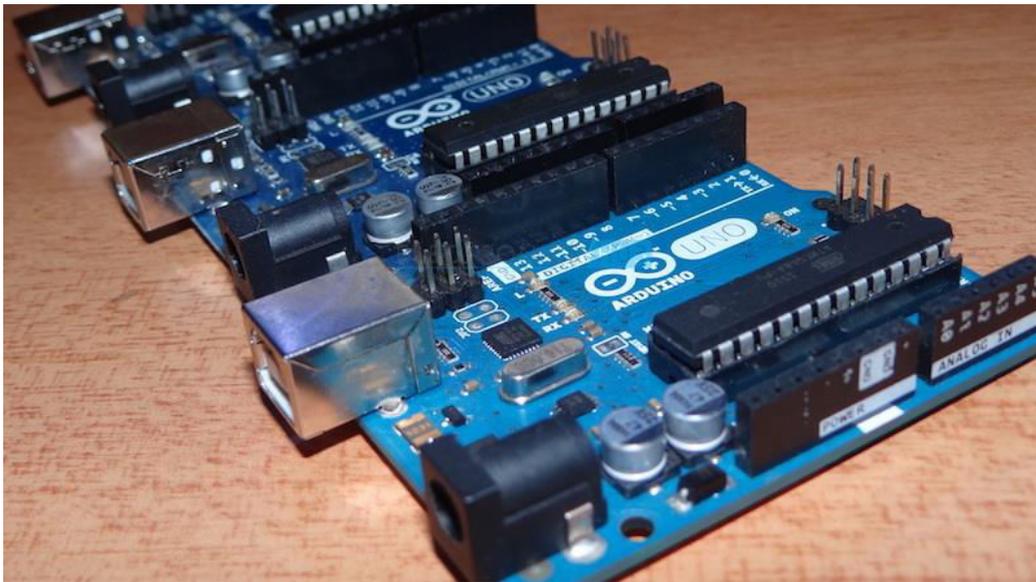
Αναλυτικότερα υπάρχουν διάφορες γλώσσες με τις οποίες μπορούμε να προγραμματίσουμε ένα μικροελεγκτή, οι οποίες είναι:

- **Γλώσσα Μηχανής** : Πρόκειται για την γλώσσα στην οποία προγραμματίζονταν οι πρώτοι υπολογιστές, της οποίας κάθε εντολή αποτελείται από μία δυαδική τιμή(opcode). Είναι εξαιρετικά δυσνόητη από τον χρήστη.
- **Assembly**: Η γλώσσα assembly βρίσκεται ένα επίπεδο παραπάνω από την γλώσσα μηχανής. Το πρόγραμμα που χρησιμοποιείται ονομάζεται assembler και οι εντολές γράφονται σε λέξεις που ονομάζονται mnemonics. Το assembler δηλαδή μετατρέπει τα mnemonics σε γλώσσα μηχανής.
- **C**: Η C αποτελεί σήμερα την πλέον προτιμητέα επιλογή για τον προγραμματισμό ενός μικροελεγκτή. Ανήκει στις γλώσσες υψηλού επιπέδου επομένως περιέχει όλα τα πλεονεκτήματα που είχαμε προαναφέρει. Αυτό που κάνει ο μεταγλωττιστής είναι να μετατρέπει την C σε Assembly και στην συνέχεια σε γλώσσα μηχανής. Με την εξέλιξη της τεχνολογίας φυσικά πλέον έχει επιτευχθεί η μετατροπή της C απευθείας σε γλώσσα μηχανής.
- **C++**: Μια object-oriented γλώσσα, κάποια χαρακτηριστικά της οποίας μπορούν να χρησιμοποιηθούν και για μικροελεγκτές, αλλά κάποια άλλα δημιουργούν προβλήματα στην παραγωγή του κώδικα.
- Η γλώσσα που χρησιμοποιεί το Arduino είναι επί της ουσίας η C και μεγάλη γκάμα βιβλιοθηκών υλοποιημένες και σε C++.

Κεφάλαιο 3^ο - Θεωρητικό Υπόβαθρο

3.1 Arduino

Το Arduino είναι μια πλατφόρμα πρωτοτύπων ανοιχτού κώδικα στα ηλεκτρονικά που βασίζεται σε εύρηστο υλικό και λογισμικό. Με λεπτό τρόπο, το Arduino είναι ένα board πρωτοτύπων βασισμένος σε μικροελεγκτή, το οποίο μπορεί να χρησιμοποιηθεί στην ανάπτυξη ψηφιακών συσκευών που μπορούν να διαβάζουν εισόδους όπως δάχτυλο σε κουμπί, άγγιγμα οθόνης, φως στον αισθητήρα, περιστροφή κινητήρα, αναπαραγωγή τραγουδιών μέσω ηχείου κ.λ.π.



Εικόνα 3.1: Arduino

Ο στόχος του Arduino είναι να εισαγάγει τον κόσμο των ηλεκτρονικών σε άτομα που έχουν μικρή έως καθόλου εμπειρία σε ηλεκτρονικά είδη, όπως χομπίστες, σχεδιαστές, καλλιτέχνες κ.λ.π. Το Arduino βασίζεται σε ηλεκτρονικά έργα ανοιχτού κώδικα, δηλαδή όλες οι προδιαγραφές σχεδιασμού, τα σχήματα, το λογισμικό διατίθενται ανοιχτά σε όλους τους χρήστες.

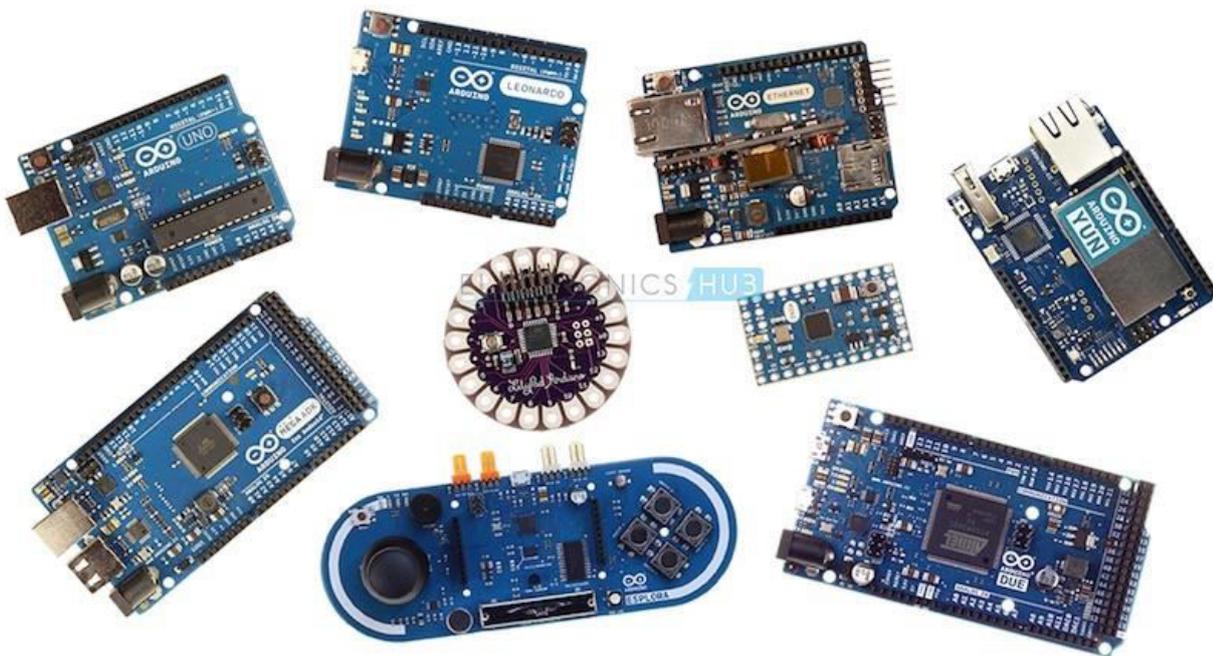
Υλικό και λογισμικό

Οι πλακέτες Arduino βασίζονται γενικά σε μικροελεγκτές της Atmel Corporation όπως μικροελεγκτές 8, 16 ή 32 bit βασισμένοι σε αρχιτεκτονική AVR. Το σημαντικό χαρακτηριστικό των πλακετών Arduino είναι οι τυπικοί σύνδεσμοι. Χρησιμοποιώντας αυτούς τους συνδετήρες, μπορούμε να συνδέσουμε την πλακέτα Arduino σε άλλες συσκευές όπως LED ή πρόσθετες μονάδες που ονομάζονται Shields. Οι πλακέτες Arduino αποτελούνται επίσης από ρυθμιστή τάσης και ταλαντωτή κρυστάλλου. Αποτελούνται επίσης από προσαρμογέα USB σε σειριακό, μέσω του οποίου ο πίνακας Arduino μπορεί να προγραμματιστεί χρησιμοποιώντας σύνδεση USB.

Τύποι Arduino Boards

Υπάρχουν πολλοί τύποι Arduino που διατίθενται στην αγορά, αλλά όλα τα boards έχουν ένα κοινό χαρακτηριστικό: μπορούν να προγραμματιστούν χρησιμοποιώντας το Arduino IDE. Οι λόγοι για τους διαφορετικούς τύπους πλακετών είναι οι διαφορετικές απαιτήσεις τροφοδοσίας, οι επιλογές συνδεσιμότητας, οι εφαρμογές τους κ.λ.π.

Τα Arduino διατίθενται σε διαφορετικά μεγέθη, παράγοντες μορφής, διαφορετικό αριθ. των ακίδων εισόδου/εξόδου κλπ. Μερικές από τις κοινώς γνωστές και συχνά χρησιμοποιούμενες πλακέτες Arduino είναι οι Arduino UNO, Arduino Mega, Arduino Nano, Arduino Micro και Arduino Lilypad.



Εικόνα 3.2: Arduino Models

Υπάρχουν πρόσθετες ενότητες που ονομάζονται Arduino Shields, οι οποίες μπορούν να χρησιμοποιηθούν για την επέκταση των λειτουργιών των πλακετών Arduino. Μερικές από τις ασπίδες που χρησιμοποιούνται συνήθως είναι η ασπίδα Arduino Proto, η ασπίδα Arduino WiFi και η ασπίδα Arduino Yun.

Arduino UNO & MEGA 2560

Σε αυτή την εφαρμογή, θα χρησιμοποιήσουμε τα Arduino UNO & MEGA 2560. Το Arduino UNO είναι μια βασική και φθηνή πλακέτα Arduino και είναι η πιο δημοφιλής από όλες του Arduino με μερίδιο αγοράς άνω του 50%. Το Arduino UNO θεωρείται η καλύτερη πλακέτα πρωτοτύπων για αρχάριους στην ηλεκτρονική και την κωδικοποίηση.

Το Arduino MEGA 2560 από την άλλη είναι η απάντηση στην ζήτηση που πάντα υπήρχε για περισσότερες δυνατότητες από αυτές που μπορεί να προσφέρει το Arduino UNO.

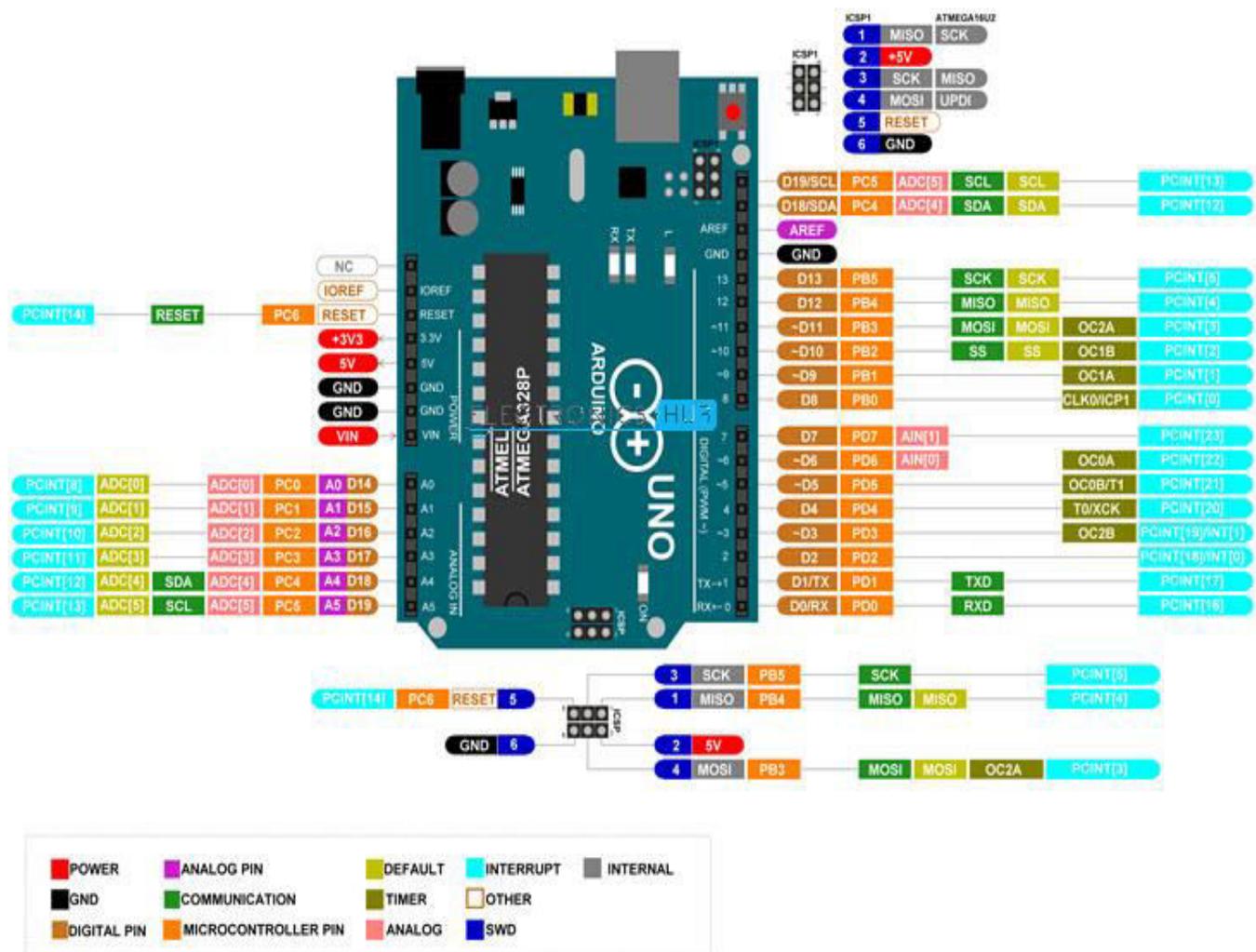
Το UNO βασίζεται στον μικροελεγκτή **ATmega328P**. Υπάρχουν δύο παραλλαγές του Arduino UNO: μία που αποτελείται από σύνδεση μικροελεγκτή μέσω οπών και άλλη με τύπο επιτοίχιας τοποθέτησης.

Το μοντέλο μέσω οπών θα είναι επωφελές καθώς μπορούμε να βγάλουμε το τσιπ σε περίπτωση οποιουδήποτε προβλήματος και να ανταλλάξουμε με ένα νέο. Όπως αναφέρθηκε προηγουμένως, ο μικροελεγκτής που χρησιμοποιείται στο UNO είναι ATmega328P, ο οποίος είναι ένας μικροελεγκτής 8-bit βασισμένος στην αρχιτεκτονική AVR.

Το Arduino MEGA βασίζεται στον μικροελεγκτή **ATmega2560**, ένα 8-bit AVR Architecture MCU βασισμένο στην ATMEL.

Διατίθεται σε τετράκλινο πακέτο 100 ακίδων. Έχει σχεδιαστεί και αναπτυχθεί για να παρέχει περισσότερο αριθμό γραμμών I/O (ψηφιακών και αναλογικών), περισσότερη μνήμη flash και περισσότερη RAM σε σύγκριση με το UNO.

3.1.1 Arduino UNO Pinout



Εικόνα 3.3: Arduino Uno Pinout

MCU	ATmega328P
Αρχιτεκτονική	AVR
Τάση λειτουργίας	5V
Τάση εισόδου	6V - 20V (όριο)
	7V - 12V (συνιστάται)
Ταχύτητα ρολογιού	16 MHz
Flash Memory	32 KB (2 KB από αυτό χρησιμοποιείται από το bootloader)
SRAM	2 KB
EEPROM	1 KB
Digital ή ψηφιακές ακίδες I/O	24 (εκ των οποίων 6 μπορούν να παράγουν PWM)

Τροφοδοσία

Υπάρχουν δύο τρόποι με τους οποίους μπορούμε να τροφοδοτήσουμε την πλακέτα. Ο πρώτος και εύκολος τρόπος είναι η χρήση του συνδετήρα USB Type-B. Ο επόμενος τρόπος είναι να παράσχουμε μια μη ρυθμιζόμενη παροχή στην περιοχή 6V έως 20V στο VIN pin του UNO. Μπορούμε επίσης να παρέχουμε την ίδια παροχή μέσω της υποδοχής DC 2,1mm, οπότε μπορούμε να έχουμε πρόσβαση στην παρεχόμενη τάση μέσω του VIN pin.

Μνήμες

Αυστηρά μιλώντας, αυτό είναι συγκεκριμένο για το MCU, δηλαδή, ATmega328P, που χρησιμοποιείται στο Arduino UNO Board. Υπάρχουν τρεις διαφορετικές μνήμες διαθέσιμες στο ATmega328P. Αυτοί είναι:

- 32 KB μνήμης flash
- 2 KB SRAM
- 1 KB EEPROM
- 0,5 KB της μνήμης Flash χρησιμοποιείται από τον κωδικό εκκίνησης.

Ακίδες εισόδου και εξόδου (I/O Pins)

Από τις 32 ακίδες που είναι διαθέσιμες στην πλακέτα UNO, 22 ακίδες σχετίζονται με είσοδο και έξοδο. Οι 14 ακίδες (D0 έως D13) είναι πραγματικές ψηφιακές ακίδες I/O, οι οποίες μπορούν να διαμορφωθούν σύμφωνα με την εφαρμογή σας χρησιμοποιώντας τις λειτουργίες **pinMode ()**, **digitalWrite ()** και **digitalRead ()**.

Όλοι αυτοί οι ψηφιακοί ακροδέκτες I/O είναι ικανοί να προμηθευτούν ή να βυθίσουν ρεύμα 20mA (επιτρέπεται το μέγιστο 40mA). Ένα πρόσθετο χαρακτηριστικό των ψηφιακών ακίδων I/O είναι η διαθεσιμότητα εσωτερικής αντίστασης έλξης (η οποία δεν είναι συνδεδεμένη από προεπιλογή). Η τιμή της εσωτερικής αντίστασης έλξης θα κυμαίνεται από 20KΩ έως 50KΩ.

Αναλογικές Ακίδες

Υπάρχουν επίσης 6 αναλογικές ακίδες εισόδου (A0 έως A5). Όλοι οι ακροδέκτες αναλογικής εισόδου παρέχουν μια λειτουργία ADC ανάλυσης 10-bit, η οποία μπορεί να διαβαστεί χρησιμοποιώντας τη λειτουργία **analogRead** (). Ένα σημαντικό σημείο σχετικά με τις ακίδες αναλογικής εισόδου είναι ότι μπορούν να διαμορφωθούν ως ψηφιακές ακίδες I/O, εάν απαιτείται. Οι ψηφιακές ακίδες I/O **3, 5, 6, 9, 10** και **11** είναι ικανές να παράγουν **σήματα PWM 8-bit**. Μπορούμε να χρησιμοποιήσουμε τη λειτουργία **analogWrite** () για αυτό.

Διεπαφές επικοινωνίας

Το Arduino UNO υποστηρίζει τρεις διαφορετικούς τύπους διεπαφών επικοινωνίας. Αυτοί είναι:

- Serial
- I2C
- SPI

Ίσως η πιο κοινή διεπαφή επικοινωνίας στο σύμπαν του Arduino είναι η Serial Communication. Στην πραγματικότητα, οι πλακέτες Arduino (UNO ή Nano ή Mega) προγραμματίζονται χρησιμοποιώντας τη σειριακή επικοινωνία. Οι ψηφιακές ακίδες I/O 0 και 1 χρησιμοποιούνται ως σειριακές ακίδες RX και TX για λήψη και μετάδοση σειριακών δεδομένων. Αυτοί οι ακροδέκτες είναι συνδεδεμένοι με τις σειριακές ακίδες του ενσωματωμένου IC σε USB Serial Converter. Οι ακίδες αναλογικής εισόδου **A4** και **A5** έχουν εναλλακτικές λειτουργίες. Μπορούν να διαμορφωθούν ως **SDA (A4)** και **SCL (A5)** για υποστήριξη επικοινωνίας I2C ή I²C ή Two Wire Interface (TWI).

Το τελικό περιβάλλον επικοινωνίας είναι το SPI. Οι ψηφιακές ακίδες I/O (**10, 11, 12, 13**) μπορούν να διαμορφωθούν ως ακίδες SPI (**SS, MOSI, MISO, SCK**) αντίστοιχα.

Επιπλέον χαρακτηριστικά

Υπάρχει ενσωματωμένο LED συνδεδεμένο με την ψηφιακή ακίδα I/O 13. Χρησιμοποιούμε αυτό το LED για να εκτελέσουμε λειτουργίες **Blinky**. Η τάση αναφοράς για το εσωτερικό ADC είναι από προεπιλογή ρυθμισμένη στα 5V. Αλλά χρησιμοποιώντας τον πείρο AREF, μπορείτε να ορίσετε μη αυτόματα το ανώτερο όριο του ADC.

Χρησιμοποιώντας την ακίδα I/O AREF, μπορούμε να ορίσουμε την τάση αναφοράς για λειτουργίες μικροελεγκτών.

Για να επαναφέρουμε τον μικροελεγκτή, μπορούμε να χρησιμοποιήσουμε το ενσωματωμένο κουμπί RESET.

Παρόλο που μπορούμε να προγραμματίσουμε το Arduino UNO χρησιμοποιώντας το καλώδιο USB, υπάρχει μια πρόβλεψη για τον προγραμματισμό του MCU χρησιμοποιώντας τη διεπαφή σειριακού προγραμματισμού σε κύκλωμα (ICSP). Ο φορτωτής εκκίνησης UART , ο οποίος είναι προφορτωμένος στον μικροελεγκτή ATmega328P, επιτρέπει τον προγραμματισμό μέσω σειριακής διεπαφής. Αλλά το ICSP δεν χρειάζεται κανέναν bootloader..

Οι ψηφιακές ακίδες I/O 2 και 3 μπορούν να διαμορφωθούν ως εξωτερικοί διακόπτες PIN INT0 και INT1 αντίστοιχα. Χρησιμοποιούμε τη λειτουργία **attachInterrupt** () για να διαμορφώσουμε τη διακοπή για ανερχόμενη παρυφή, κατερχόμενη παρυφή ή αλλαγή στάθμης στην ακίδα.

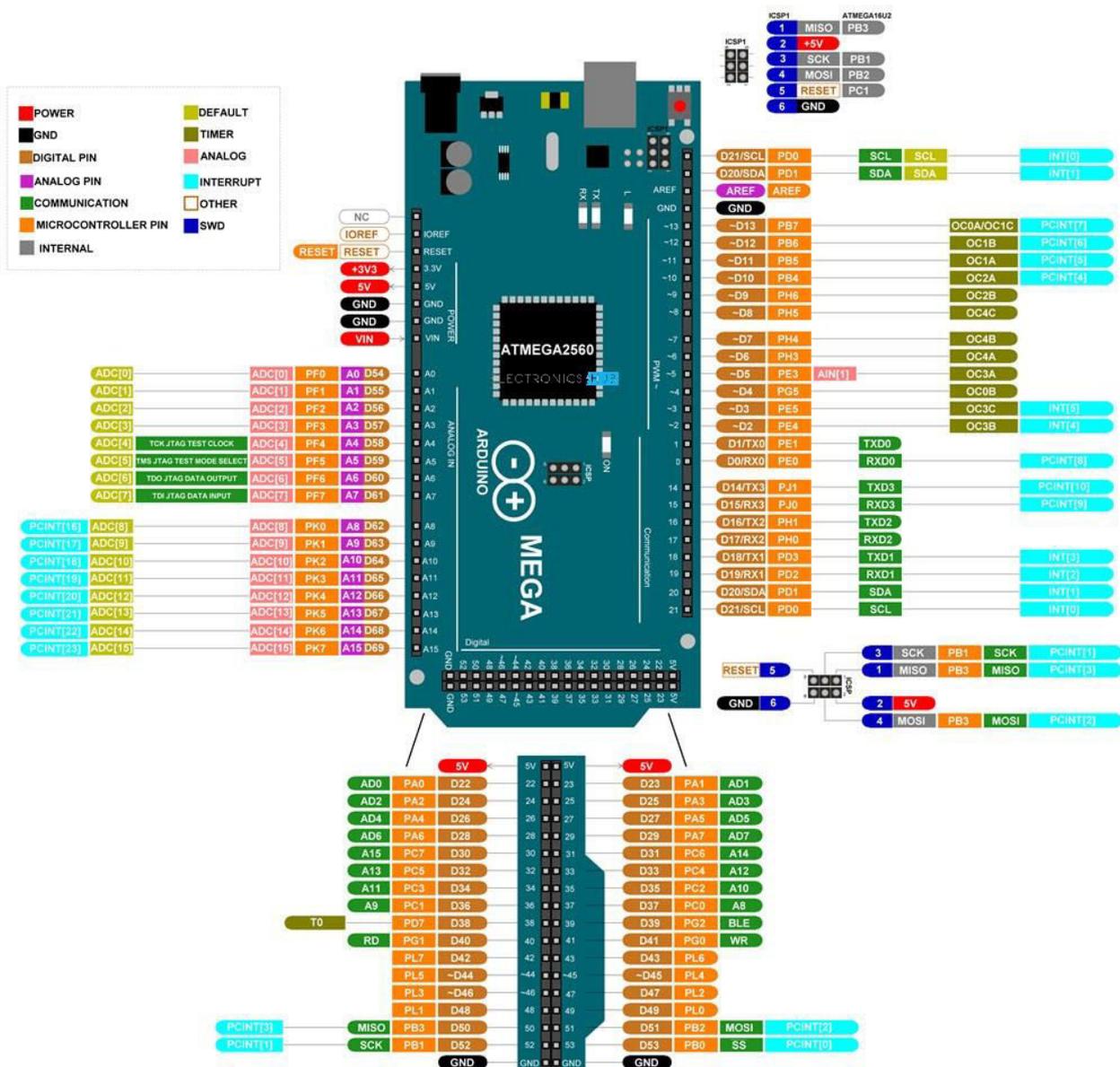
Περιγραφή ακίδων

Για περιγραφή pin του Arduino UNO, η αρίθμηση αρχίζει με το RX Pin (D0). Έτσι, το RX είναι το Pin 1, το TX είναι το Pin 2, το D2 είναι το Pin 3 και ούτω καθεξής. Από την άλλη πλευρά, το NC είναι Pin 19, το I/OREF είναι Pin 20 κλπ. Συνολικά, υπάρχουν 32 ακίδες στο Arduino UNO Board.

3.1.2 Arduino MEGA 2560 Pinout

Καθώς το Arduino Mega βασίζεται στον μικροελεγκτή ATmega2560, οι τεχνικές προδιαγραφές του Arduino Mega σχετίζονται κυρίως με το ATmega2560 MCU. Ωστόσο, μετά το διάγραμμα υπάρχει μια σύντομη επισκόπηση σχετικά με ορισμένες σημαντικές τεχνικές προδιαγραφές του Arduino Mega 2560.

MCU	ATmega2560
Αρχιτεκτονική	AVR
Τάση λειτουργίας	5V
Τάση εισόδου	6V - 20V (όριο)
	7V - 12V (συνιστάται)
Ταχύτητα ρολογιού	16 MHz
Flash Memory	256 KB (8 KB από αυτό χρησιμοποιείται από το bootloader)
SRAM	8 KB
EEPROM	4 KB
Digital ή ψηφιακές ακίδες I/O	54 (εκ των οποίων 15 μπορούν να παράγουν PWM)
Αναλογικές ακίδες εισαγωγής	16



Εικόνα 3.4: Arduino Mega Pinout

Τροφοδοσία

Οι τρόποι με τους οποίους μπορούμε να τροφοδοτήσουμε την πλακέτα Mega 2560 είναι ακριβώς οι ίδιοι με τη UNO και τους έχουμε προαναφέρει στο (3.2.1_ τροφοδοσία).

Μνήμες

Αυστηρά μιλώντας, τα παρακάτω είναι συγκεκριμένα για τον MCU, ATmega2560, που χρησιμοποιείται στο Arduino Mega. Υπάρχουν τρεις διαφορετικές μνήμες διαθέσιμες στο ATmega2560, των οποίων τα χαρακτηριστικά είναι και μια από τις βασικές διαφορές με το UNO.

- 256 KB μνήμης flash
- 8 KB SRAM
- 4 KB EEPROM
- 8 KB της μνήμης Flash χρησιμοποιούνται από τον κωδικό εκκίνησης

Ακίδες εισόδου και εξόδου (I/O Pins)

Από τις 86 ακίδες που διατίθενται στον πίνακα Mega, οι 72 ακίδες σχετίζονται με είσοδο και έξοδο. Σε αυτό 54 ακίδες (D0 έως D53) είναι πραγματικές ψηφιακές ακίδες I/O, οι οποίες μπορούν να διαμορφωθούν σύμφωνα με την εφαρμογή σας χρησιμοποιώντας τις λειτουργίες **pinMode** (), **digitalWrite** () και **digitalRead** ().

Όλοι αυτοί οι ψηφιακοί ακροδέκτες I/O είναι ικανοί να προμηθευτούν ή να βυθίσουν ρεύμα 20mA (επιτρέπεται το μέγιστο 40mA). Ένα πρόσθετο χαρακτηριστικό των ψηφιακών ακίδων IO είναι η διαθεσιμότητα εσωτερικής αντίστασης έλξης (η οποία δεν είναι συνδεδεμένη από προεπιλογή). Η τιμή της εσωτερικής αντίστασης έλξης θα κυμαίνεται από 20KΩ έως 50KΩ.

Υπάρχουν επίσης 16 αναλογικές ακίδες εισόδου (A0 έως A15). Όλοι οι ακροδέκτες αναλογικής εισόδου παρέχουν μια λειτουργία ADC ανάλυσης 10-bit, η οποία μπορεί να διαβαστεί χρησιμοποιώντας τη λειτουργία **analogRead** ().

Ένα σημαντικό σημείο σχετικά με τις ακίδες αναλογικής εισόδου είναι ότι μπορούν να διαμορφωθούν ως ψηφιακές ακίδες I/O, εάν απαιτείται.

Οι ψηφιακές ακίδες I/O 2-13 και 44-46 είναι ικανές να παράγουν σήματα PWM 8 bit. Μπορούμε να χρησιμοποιήσουμε τη λειτουργία **analogWrite** () για αυτό.

Διεπαφές επικοινωνίας

Οι διεπαφές επικοινωνίας στην πλακέτα Mega 2560 είναι ακριβώς οι ίδιες με τη UNO και τις έχουμε προαναφέρει στο **Κεφάλαιο (3.2.1_ Διεπαφές Επικοινωνίας)**. Με την μοναδική διάφορα να βρίσκεται στον αριθμό των διεπαφών σειριακής επικοινωνίας υλικού.

Το Arduino Mega υποστηρίζει τέσσερις διεπαφές σειριακής επικοινωνίας υλικού. Οι ψηφιακές ακίδες I/O 0 και 1 χρησιμοποιούνται ως σειριακές ακίδες RX0 και TX0 για λήψη και μετάδοση σειριακών δεδομένων. Αυτοί οι ακροδέκτες είναι συνδεδεμένοι με τις σειριακές ακίδες του ενσωματωμένου IC σε USB Serial Converter.

Ομοίως. Οι ψηφιακές ακίδες I/O 19 και 18 ως RX1 και TX1, 17 και 16 ως RX2 και TX2 και 15 και 14 ως RX3 και TX3 αντίστοιχα.

Οι ψηφιακές ακίδες I/O 20 και 21 μπορούν να διαμορφωθούν ως SDA (20) και SCL (21) για να υποστηρίξουν την επικοινωνία I2C ή I²C ή Two Wire Interface (TWI).

Το τελικό περιβάλλον επικοινωνίας είναι το SPI. Οι ψηφιακές ακίδες I/O (50, 51, 52, 53) μπορούν να διαμορφωθούν ως ακίδες SPI (MISO, MOSI, SCK, SS) αντίστοιχα.

Επιπλέον χαρακτηριστικά

Και εδώ βρίσκουμε ομοιότητα ανάμεσα στις 2 πλακέτες πέραν του ότι, οι ψηφιακοί ακροδέκτες I/O που μπορούν να διαμορφωθούν ως ακίδες εξωτερικής διακοπής είναι περισσότεροι. Αυτοί είναι τα I/O Pins (**2, 3, 18, 19, 20 και 21**) οι οποίοι διαμορφώνονται σε ακίδες εξωτερικής διακοπής (**INT0, INT1, INT5, INT4, INT3 και INT2**) αντίστοιχα.

3.2 Arduino Modules

Με απλά λόγια, ένα "**Module**" στο περιβάλλον Arduino είναι μια πλακέτα έτοιμη προς χρήση με έναν αισθητήρα ή μια συσκευή I/O πάνω της. Για να χρησιμοποιήσουμε έναν αισθητήρα ή μια συσκευή (όπως μια οθόνη LCD), χρειάζεστε λίγα επιπλέον εξαρτήματα.

Αυτό που κάνει μια μονάδα Arduino είναι ότι ενσωματώνει όλα τα απαραίτητα εξαρτήματα και συνδέσεις που απαιτούνται για να λειτουργήσει ένας αισθητήρας (ή μια συσκευή) και με αυτόν τον τρόπο μπορούμε να παρουσιάσουμε ένα έτοιμο προς χρήση προϊόν.

3.2.1 Gps Module (L76X)

Πιο συγκεκριμένα ένα από τα Arduino modules που θα χρησιμοποιήσουμε είναι το L76X GPS-Module μέσω του οποίου καθορίζουμε την ακριβή γεωγραφική θέση του οχήματος και την ταχύτητα του.

Χαρακτηριστικά

- Υποστηρίζει συστήματα Multi-GNSS: GPS, BDS και QZSS
- EASY™, τεχνολογία αυτόματης πρόβλεψης, βοηθά στη γρήγορη τοποθέτηση
- AlwaysLocate intellig, έξυπνος ελεγκτής περιοδικής λειτουργίας για εξοικονόμηση ενέργειας
- Υποστηρίζει DGPS, SBAS (WAAS/EGNOS/MSAS/GAGAN)
- Baudrate επικοινωνίας UART: 4800 ~ 115200bps (9600bps από προεπιλογή)
- Επαναφορτιζόμενη επαναφορτιζόμενη μπαταρία Li MS621FE, για τη διατήρηση των πληροφοριών εφήμερου και τις καυτές εκκινήσεις
- 2x LED για ένδειξη της κατάστασης λειτουργίας της μονάδας
- Έρχεται με πόρους ανάπτυξης και εγχειρίδιο (παραδείγματα για Raspberry Pi/Arduino/STM32)

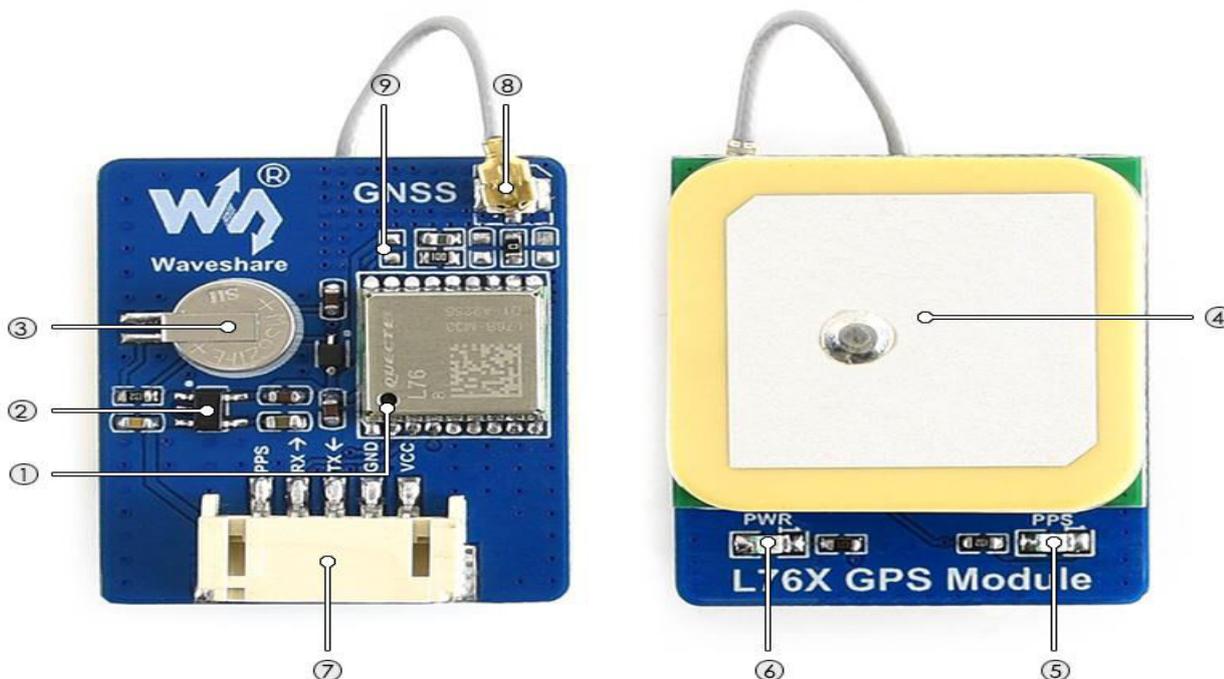
Προδιαγραφές GNSS

- Ζώνη: GPS L1 (1575.42Mhz), BD2 B1 (1561.098MHz)
- Κανάλια: 33 παρακολούθηση ch, 99 απόκτηση ch, 210 PRN ch
- Κωδικός C/A
- SBA: WAAS, EGNOS, MSAS, GAGAN
- Οριζόντια ακρίβεια θέσης:
- Αυτόνομο: <2,5mCEP
- Time-To-First-Fix @-130dBm (EASY™ ενεργοποιημένο):
- Έναρξη κρύου: <15 δευτ
- Θερμές εκκινήσεις: <5 δευτ
- Θερμές εκκινήσεις: <1 δευτ
- Ευαισθησία:
- Απόκτηση: -148dBm

- Παρακολούθηση: -163dBm
- Εξαγορά: -160dBm
- Δυναμική απόδοση:
- Υψόμετρο (μέγιστο): 18000μ
- Ταχύτητα (μέγιστο): 515m/s
- Επιτάχυνση (μέγιστο): 4G

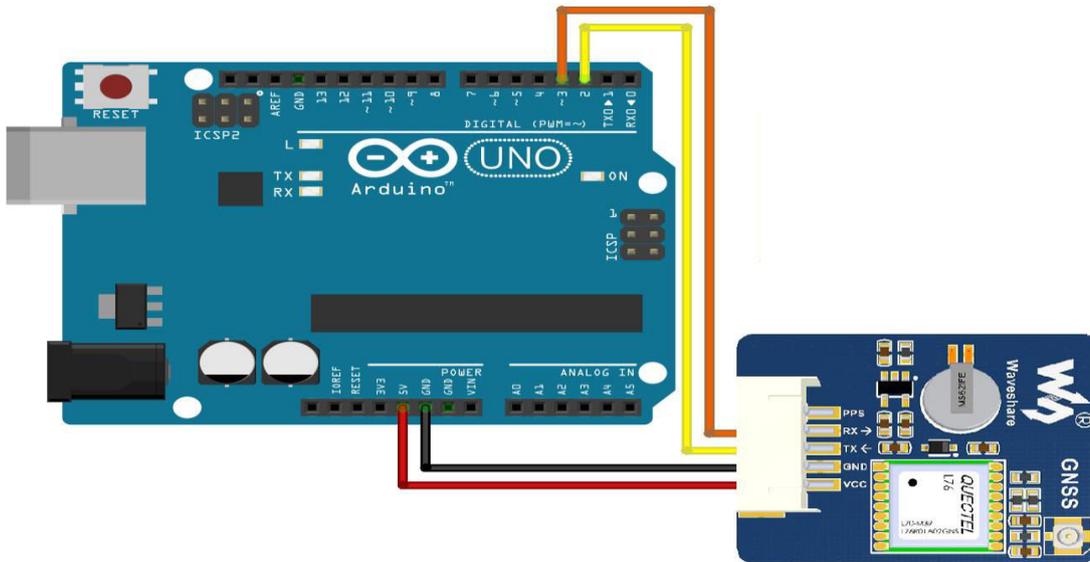
Γενικά Χαρακτηριστικά

- Διεπαφή επικοινωνίας: UART
- Baudrate: 4800 ~ 115200bps (9600bps από προεπιλογή)
- Ρυθμός ενημέρωσης: 1Hz (προεπιλογή), 10Hz (μέγιστο)
- Πρωτόκολλα: NMEA 0183, PMTK
- Τάση τροφοδοσίας: 5V / 3.3V
- Ρεύμα λειτουργίας: 11mA
- Θερμοκρασία λειτουργίας: -40 °C ~ 85
- Διαστάσεις: 32,5mm x 25,5mm



Εικόνα 3.5: GPS Module

1. **RT9193-33:** διαχειριστής ισχύος
2. **Επαναφορτιζόμενη μπαταρία MS621FE Li:** για διατήρηση εφημερίδων και θερμών εκκινήσεων
3. **Κεραμική ενεργή κεραία**
4. **Ένδειξη κατάστασης GPS**
5. **Ένδειξη λειτουργίας**
6. **Υποδοχή PH2.0 5PIN**
7. **Υποδοχή κεραίας GNSS**
8. **Δημιουργία εφεδρικού βραχυκυκλωτήρα :** δεν έχει κολληθεί από προεπιλογή, κολλάμε το βραχυκυκλωτήρα για έξοδο από τη λειτουργία δημιουργίας αντιγράφων ασφαλείας.



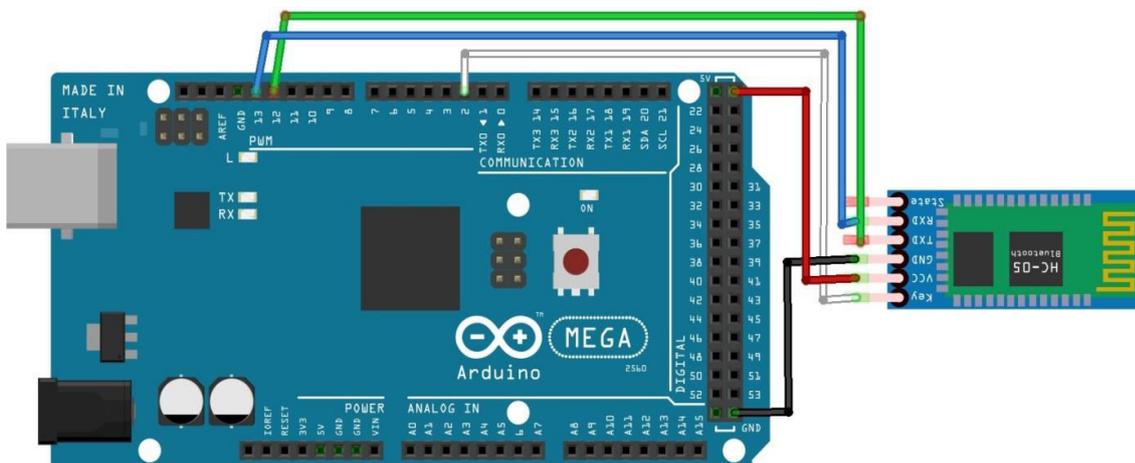
Εικόνα 3.6: Σύνδεση GPS Module

Pinout

ΑΡΙΘ.ΠΙΝ	ΣΥΜΒΟΛΟ	ΠΕΡΙΓΡΑΦΗ
1	VCC	Τροφοδοσία (είσοδος 3.3V/5V)
2	GND	Έδαφος
3	TX	Σύνδεση στο MCU.RX (κατεύθυνση σήματος: TX >> MCU.RX)
4	RX	Σύνδεση στο MCU.TX (κατεύθυνση σήματος: MCU.TX >> RX)
5	PPS	Ένδειξη κατάστασης GPS

3.2.2 Bluetooth (HC-05)

Το HC-05 είναι μια δημοφιλής μονάδα bluetooth που μπορεί να προσθέσει ασύρματη λειτουργικότητα αμψίδρομης (full-duplex) επικοινωνίας στα έργα μας για αυτό και παίζει κομβικό ρολό στην συλλογή δεδομένων της παρακάτω εργασίας.



Εικόνα 3.7: Σύνδεση Bluetooth Module

Αριθμός Ακίδας	Όνομα Ακίδων	Περιγραφή
1	Enable / Key	Αυτός ο πείρος χρησιμοποιείται για εναλλαγή μεταξύ της λειτουργίας δεδομένων (χαμηλή ρύθμιση) και της λειτουργίας εντολών AT (υψηλή ρύθμιση). Από προεπιλογή είναι σε λειτουργία δεδομένων
2	Vcc	Τροφοδοτεί την ενότητα. Συνδέστε σε +5V Τάση τροφοδοσίας
3	Ground	Πείρος γείωσης της μονάδας, συνδεθείτε στη γείωση του συστήματος.
4	TX - Πομπός	Μεταδίδει σειριακά δεδομένα. Όλα όσα λαμβάνονται μέσω Bluetooth θα δοθούν από αυτήν την καρτίτσα ως σειριακά δεδομένα.
5	RX - Δέκτης	Λήψη σειριακών δεδομένων. Όλα τα σειριακά δεδομένα που δίνονται σε αυτό το pin θα μεταδίδονται μέσω Bluetooth
6	State	Ο ακροδέκτης κατάστασης είναι συνδεδεμένος με την ενδεικτική λυχνία LED, μπορεί να χρησιμοποιηθεί ως ανατροφοδότηση για να ελέγξετε εάν το Bluetooth λειτουργεί σωστά.
7	LED	Υποδεικνύει την κατάσταση της ενότητας
		Αναβοσβήνει μία φορά σε 2 δευτερόλεπτα: Η μονάδα έχει εισέλθει στη λειτουργία εντολών
		Επαναλαμβανόμενο αναβοσβήσιμο: Αναμονή σύνδεσης στη λειτουργία δεδομένων
		Αναβοσβήνει δύο φορές σε 1 δευτερόλεπτο: Η σύνδεση είναι επιτυχής στη λειτουργία δεδομένων
8	Button	Χρησιμοποιείται για τον έλεγχο της καρτίτσας Key/Enable για εναλλαγή μεταξύ δεδομένων και λειτουργίας εντολών

Προεπιλεγμένες ρυθμίσεις

- Προεπιλεγμένο όνομα Bluetooth: "HC-05"
- Προεπιλεγμένος κωδικός πρόσβασης: 1234 ή 0000
- Προεπιλεγμένη επικοινωνία: Slave
- Προεπιλεγμένη λειτουργία: Λειτουργία δεδομένων
- Λειτουργία δεδομένων Βαθμός Baud: 9600, 8, N, 1
- Λειτουργία εντολής Baud Rate: 38400, 8, N, 1
- Προεπιλεγμένο υλικολογισμικό: LINVOR

Τεχνικές προδιαγραφές

- Σειριακή μονάδα Bluetooth για Arduino και άλλους μικροελεγκτές
- Τάση λειτουργίας: 4V έως 6V (Συνήθως +5V)
- Ρεύμα λειτουργίας: 30mA
- Εμβέλεια: <100μ
- Λειτουργεί με σειριακή επικοινωνία (USART) και συμβατή με TTL
- Ακολουθεί το τυποποιημένο πρωτόκολλο IEEE 802.15.1
- Χρησιμοποιεί φάσμα διασποράς συχνότητας (FHSS)
- Μπορεί να λειτουργήσει σε λειτουργία **Master, Slave** ή **Master/Slave**
- Μπορεί να διασυνδεθεί εύκολα με φορητό υπολογιστή ή κινητά τηλέφωνα με Bluetooth
- Υποστηριζόμενο **baudrate**: 9600,19200,38400,57600,115200,230400,460800.

Εφαρμογές

Το **HC-05** είναι μια δημοφιλής ενότητα που μπορεί να προσθέσει αμφίδρομη (πλήρης αμφίδρομη) ασύρματη λειτουργικότητα. Μπορούμε να χρησιμοποιήσετε αυτό το “module” για επικοινωνία μεταξύ δύο μικροελεγκτών όπως το Arduino ή να επικοινωνούμε με οποιαδήποτε συσκευή με λειτουργικότητα Bluetooth, όπως τηλέφωνο ή φορητό υπολογιστή. Υπάρχουν πολλές Android εφαρμογές που είναι ήδη διαθέσιμες, γεγονός που καθιστά αυτή τη διαδικασία πολύ πιο εύκολη. Η μονάδα επικοινωνεί με τη βοήθεια του USART με 9600 baudrate, επομένως είναι εύκολη η διασύνδεση με οποιονδήποτε μικροελεγκτή που υποστηρίζει USART. Μπορούμε επίσης να διαμορφώσουμε τις προεπιλεγμένες τιμές της μονάδας χρησιμοποιώντας τη λειτουργία “**command mode**”

Ωστόσο, αυτό το “**module**” δεν μπορεί να μεταφέρει πολυμέσα όπως φωτογραφίες ή τραγούδια.

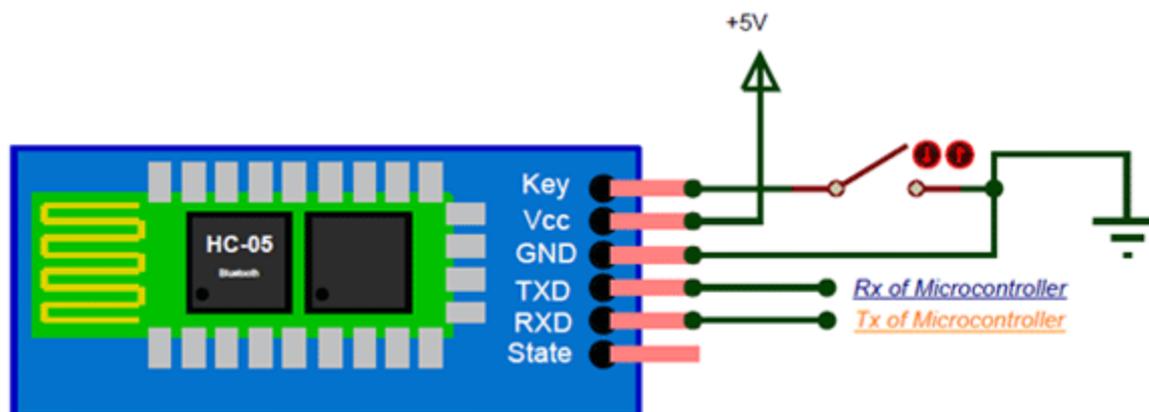
Οι βασικές υλοποιήσεις που βρίσκει εφαρμογή είναι οι εξής:

- Ασύρματη επικοινωνία μεταξύ δύο μικροελεγκτών
- Επικοινωνήστε με φορητό υπολογιστή, επιτραπέζιους υπολογιστές και κινητά τηλέφωνα
- Εφαρμογή καταγραφής δεδομένων
- Καταναλωτικές εφαρμογές
- Ασύρματα ρομπότ
- Αυτοματισμός σπιτιού

Λειτουργίες

Το HC-05 έχει δύο τρόπους λειτουργίας, ο ένας είναι η λειτουργία δεδομένων στην οποία μπορεί να στέλνει και να λαμβάνει δεδομένα από άλλες συσκευές Bluetooth και η άλλη είναι η λειτουργία **AT Command** όπου μπορούν να αλλάξουν οι προεπιλεγμένες ρυθμίσεις της συσκευής. Μπορούμε να λειτουργήσουμε τη συσκευή σε οποιαδήποτε από αυτές τις δύο λειτουργίες χρησιμοποιώντας την ακίδα **KEY** όπως εξηγείται στην περιγραφή της.

Μπορούμε εύκολα να συνδυάσουμε τη μονάδα HC-05 με μικροελεγκτές επειδή λειτουργεί χρησιμοποιώντας το πρωτόκολλο σειριακής θύρας (SPP). Τροφοδοτούμε τη μονάδα με +5V και συνδέουμε την ακίδα Rx της μονάδας στο Tx της MCU και την ακίδα Tx της μονάδας στο Rx της MCU όπως φαίνεται στο παρακάτω σχήμα.



Εικόνα 3.8: Διάγραμμα Bluetooth Module

Κατά τη διάρκεια της ενεργοποίησης, ο ακροδέκτης key μπορεί να γειωθεί για είσοδο στη λειτουργία εντολών, εάν παραμείνει ελεύθερος, από προεπιλογή θα εισέλθει στη λειτουργία δεδομένων. Μόλις ενεργοποιηθεί η μονάδα, θα πρέπει να μπορούμε να ανιχνεύσουμε τη συσκευή Bluetooth ως "HC-05" και στη συνέχεια να συνδεθούμε με αυτήν χρησιμοποιώντας τον προεπιλεγμένο κωδικό πρόσβασης **1234**. Ο κωδικός πρόσβασης ονόματος και άλλες προεπιλεγμένες παράμετροι μπορούν να αλλάξουν μέσω της λειτουργίας **AT Command**.

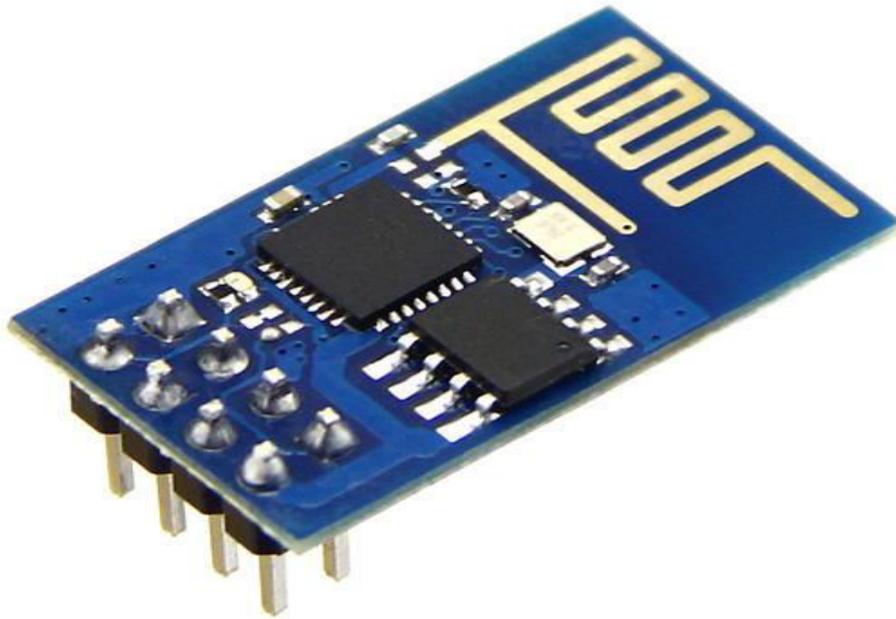
3.2.3 Wifi (ESP-8266)

Εισαγωγή

Το ESP8266 είναι λειτουργική μονάδα Wi-Fi συστήματος σε τσιπ (SoC) που αναπτύχθηκε από το σύστημα Espressif. Χρησιμοποιείται κυρίως για την ανάπτυξη ενσωματωμένων εφαρμογών IoT (Internet of Things). Το τσιπ ήρθε για πρώτη φορά στην προσοχή των δυτικών κατασκευαστών τον Αύγουστο του 2014 με τη μονάδα **ESP-01**, κατασκευασμένη από τον κατασκευαστή Ai-Thinker. Αυτή η μικρή μονάδα επιτρέπει στους μικροελεγκτές να συνδεθούν σε δίκτυο Wi-Fi και να κάνουν απλές συνδέσεις TCP/IP χρησιμοποιώντας εντολές τύπου **Hayes**.

Ωστόσο, στην αρχή, δεν υπήρχε σχεδόν καμία αγγλική τεκμηρίωση στο τσιπ και τις εντολές που δέχτηκε. Η πολύ χαμηλή τιμή και το γεγονός ότι υπήρχαν πολύ λίγα εξωτερικά εξαρτήματα στη μονάδα, γεγονός που υποδηλώνει ότι θα μπορούσε τελικά να είναι πολύ φθηνό σε όγκο, προσέλκυσε πολλούς χάκερ να εξερευνήσουν τη μονάδα, το τσιπ και το λογισμικό σε αυτήν, καθώς και για τη μετάφραση της κινεζικής τεκμηρίωσης.

Υλικολογισμικό που θα αναπτύξουμε είναι με βάση το Arduino. Με αυτόν τον πυρήνα, ο επεξεργαστής ESP8266 και τα στοιχεία του Wi-Fi μπορούν να προγραμματιστούν όπως κάθε άλλη συσκευή Arduino



Εικόνα 3.9: ESP8266-01 WiFi Module

Το ESP8266 έρχεται με τις δυνατότητες

- Wi-Fi 2,4 GHz (802,11 b/g/n, υποστηρίζει WPA/WPA2),
- είσοδος/έξοδος γενικής χρήσης (16 GPIO),
- Πρωτόκολλο σειριακής επικοινωνίας Inter-Integrated Circuit (I²C),
- μετατροπή αναλογικού σε ψηφιακό (ADC 10-bit)
- Serial Peripheral Interface (SPI) σειριακό πρωτόκολλο επικοινωνίας,
- Διεπαφές I²S (Inter-IC Sound) με DMA (Direct Memory Access) (κοινή χρήση ακίδων με GPIO),
- UART (σε ειδικές καρφίτσες, συν ένα UART μόνο για μετάδοση μπορεί να ενεργοποιηθεί στο GPIO2), και
- διαμόρφωση πλάτους παλμού (PWM).

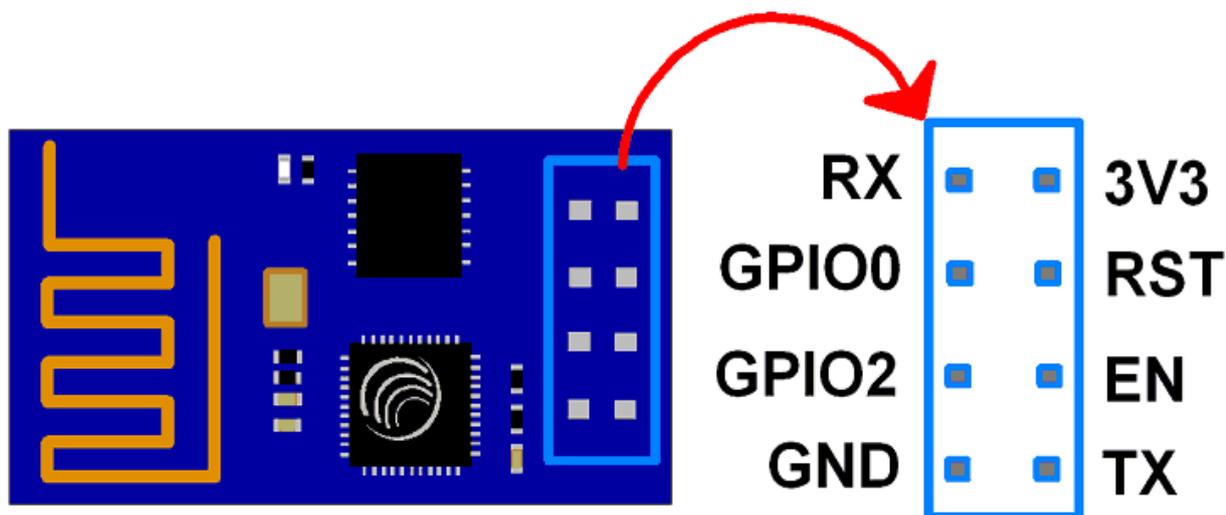
Χρησιμοποιεί έναν επεξεργαστή RISC 32-bit βασισμένο στο Tensilica Xtensa L106 που λειτουργεί στα 80 MHz (ή overclocked στα 160 MHz). Διαθέτει 64 KB boot ROM, 64 KB instruction RAM και 96 KB RAM. Η εξωτερική μνήμη flash μπορεί να έχει πρόσβαση μέσω SPI. Η μονάδα ESP8266 είναι ένας ανεξάρτητος ασύρματος πομποδέκτης χαμηλού κόστους που μπορεί να χρησιμοποιηθεί για εξελίξεις IoT τελικού σημείου.

Για να επικοινωνήσει με τη μονάδα ESP8266, ο μικροελεγκτής πρέπει να χρησιμοποιήσει ένα σύνολο εντολών AT. Ο μικροελεγκτής επικοινωνεί με τη μονάδα ESP8266-01 χρησιμοποιώντας το UART με καθορισμένο Baudrate.

Υπάρχουν πολλοί κατασκευαστές που παράγουν διαφορετικές μονάδες με βάση αυτό το τσιπ. Έτσι, η ενότητα συνοδεύεται από διαφορετικές επιλογές διαθεσιμότητας καρφιτσών, όπως:

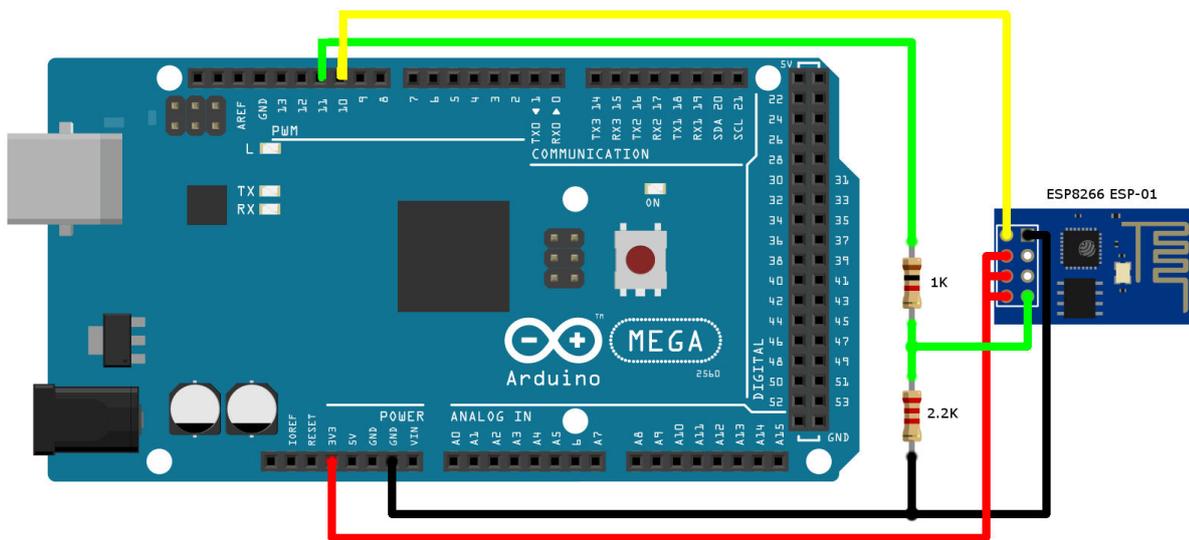
- Το ESP-01 έρχεται με 8 ακίδες (2 ακίδες GPIO)-Κεραία ιχνηλάτησης PCB.
- Το ESP-02 έρχεται με 8 ακίδες, (3 ακίδες GPIO)-Υποδοχή κεραίας U-FL.
- Το ESP-03 έρχεται με 14 ακίδες, (7 καρφίτσες GPIO)-Κεραμική κεραία.
- Το ESP-04 έρχεται με 14 ακίδες, (7 καρφίτσες GPIO)-Χωρίς κεραία.

Περιγραφή ακίδων



Εικόνα 3.10: WiFi Module Pinout

Όνομα Ακίδας	Περιγραφή
3V3	Ακίδα τροφοδοσίας 3,3 V
GND	Ακίδα γείωσης
RST	Reset ακίδα χαμηλής επαναφοράς
EN	Enable ακίδα υψηλής ενεργοποίησης
TX	Serial Transmit ακίδα του UART
RX	Serial Receive ακίδα του UART
GPIO0	Ακίδες εισόδου/εξόδου γενικού σκοπού. Αυτοί οι ακροδέκτες αποφασίζουν σε ποια λειτουργία (εκκίνηση ή κανονική) ξεκινά η μονάδα. Επίσης, αποφασίζει εάν οι ακίδες TX/RX χρησιμοποιούνται για τον προγραμματισμό της μονάδας ή για σειριακούς σκοπούς εισόδου/εξόδου
GPIO2	



Εικόνα 3.11: Σύνδεση WiFi Module

Για να προγραμματίσουμε τη μονάδα χρησιμοποιώντας το UART, συνδέουμε το GPIO0 στο GND και το GPIO2 στο VCC ή αφήνουμε το ανοιχτό. Για να χρησιμοποιήσουμε το UART για κανονικές σειριακές εισόδους/εξόδους αφήνουμε και τις δύο ακίδες ανοιχτές (ούτε VCC ούτε Ground).

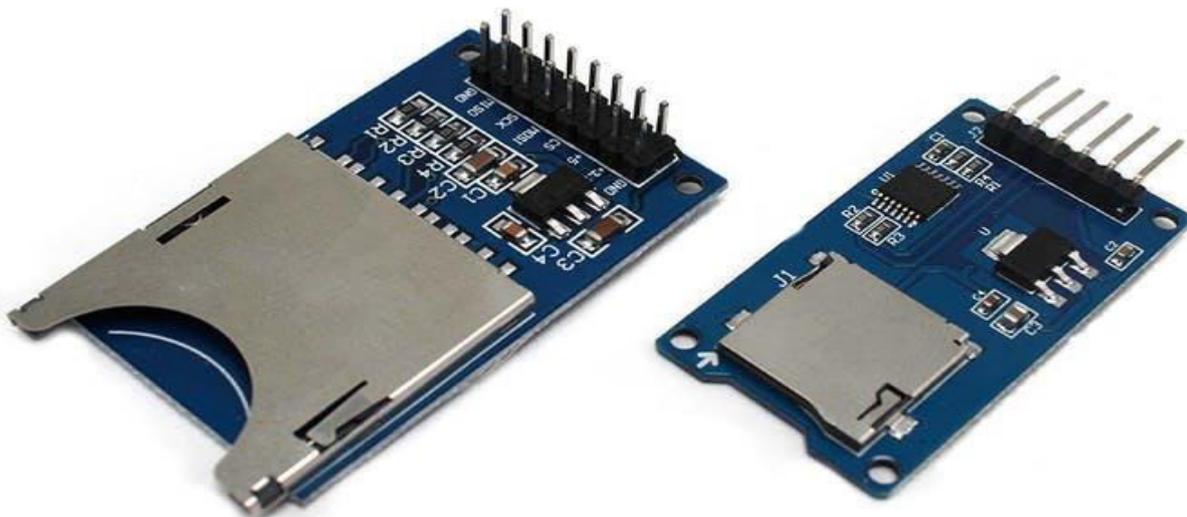
Το ESP8266-01 δίνει πολλές μεθόδους για επικοινωνία μαζί του μέσω των ακίδων RX/TX ή μέσω του αέρα (OTA) . Οι διαφορές δεν είναι μόνο στο υλικό, αλλά μπορεί να είναι και στο είδος του υλικολογισμικού που αναβοσβήνει από την πλακέτα. Ανεξάρτητα από το προεπιλεγμένο υλικολογισμικό που έχουμε εγκαταστήσει, θα πρέπει να μπορούμε να αναβοσβήσουμε το προτιμώμενο υλικολογισμικό ακολουθώντας τις οδηγίες αναβοσβήματος από το φύλλο δεδομένων. Αυτή η πλακέτα μπορεί να προγραμματιστεί χρησιμοποιώντας κώδικα LUA, κώδικα Arduino ή απευθείας μέσω εντολών AT και αυτό μας δίνει μεγαλύτερη ελευθερία κατά την ενσωμάτωση αυτής της συσκευής στα έργα μας. Υπάρχουν επίσης λίγες λειτουργίες υλικολογισμικού ρυθών, αλλά δεν είχα την ευκαιρία να τις δοκιμάσω. Προσωπικά επιλέγω να συνεργαστώ με το Arduino λόγω της προηγούμενης εμπειρίας και των διαθέσιμων βιβλιοθηκών .

Όπως έρχεται, η μονάδα είναι έτοιμη να επικοινωνήσει μέσω εντολών AT χωρίς άλλες επιπλέον ρυθμίσεις. Υπάρχουν πολλές εφαρμογές λογισμικού που μπορούμε να χρησιμοποιήσουμε για να επικοινωνία μέσω AT και πληθώρα έτοιμων εργαλείων και λειτουργιών. Χρησιμοποίησα το. Μετά την εκκίνηση, για να μπορούμε να χρησιμοποιήσετε εντολές AT. Λίγα βασικά παραδείγματα εντολών AT:

- AT - απάντηση OK
- AT+CWLAP - λίστα κοντινών διαθέσιμων δικτύων WiFi
- AT+GMR - ελέγξτε την έκδοση υλικολογισμικού
- AT+CWJAP = "<access_point_name>", "<password>" - συνδεση στο δίκτυο WiFi χρησιμοποιώντας "ssid" και "password"
- AT+CIFSR - λάβετε την τρέχουσα καταναμεμημένη διεύθυνση I

3.2.4 SD Card Module

Η αποθήκευση δεδομένων είναι ένα από τα πιο σημαντικά μέρη κάθε έργου. Υπάρχουν διάφοροι τρόποι αποθήκευσης δεδομένων ανάλογα με τον τύπο και το μέγεθος των δεδομένων. Οι κάρτες SD και micro SD είναι από τις πιο πρακτικές μεταξύ των συσκευών αποθήκευσης, οι οποίες χρησιμοποιούνται σε συσκευές όπως κινητά τηλέφωνα, μικροϋπολογιστές κ.λπ.



Εικόνα 3.12: Κάρτα SD & Micro SD

Οι μονάδες κάρτας SD και micro SD σας επιτρέπουν να επικοινωνείτε με την κάρτα μνήμης και να γράφετε ή να διαβάζετε τις πληροφορίες σε αυτές. Η ενότητα διασυνδέεται στο πρωτόκολλο SPI.

Για να χρησιμοποιήσετε αυτές τις ενότητες με το Arduino χρειάζεστε τη βιβλιοθήκη SD. Αυτή η βιβλιοθήκη είναι εγκατεστημένη στην εφαρμογή Arduino IDE από προεπιλογή.

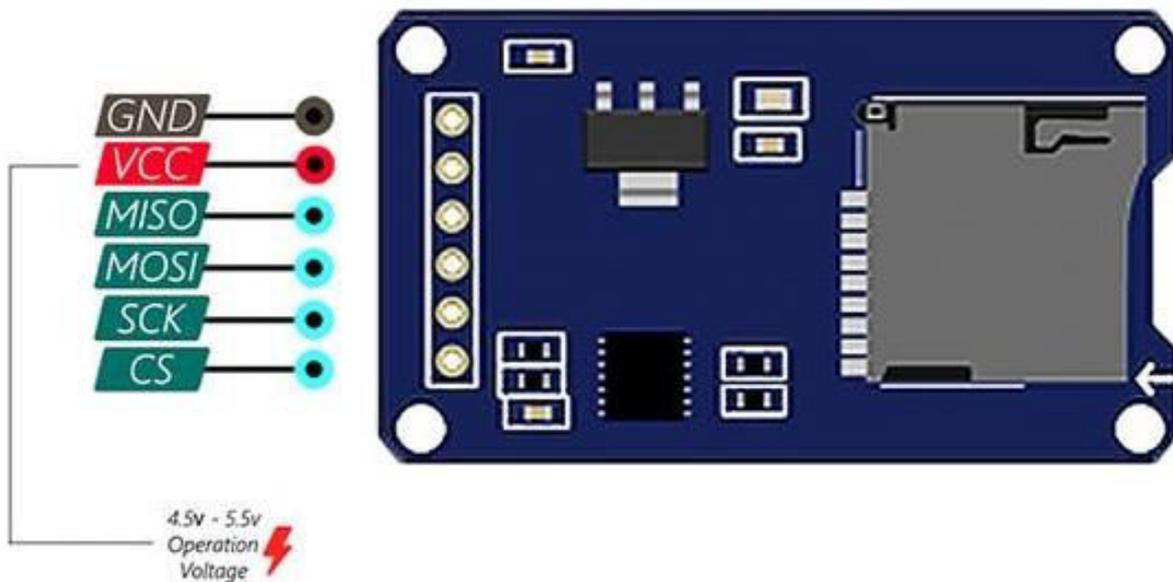
Αυτές οι μονάδες δεν μπορούν να χειριστούν κάρτες μνήμης υψηλής χωρητικότητας. Συνήθως, η μέγιστη αναγνωρίσιμη χωρητικότητα αυτών των μονάδων είναι 2 GB για κάρτες SD και 16 GB για κάρτες micro SD.

Χαρακτηριστικά

- Υποστήριξη κάρτας Micro SD, κάρτας Micro SDHC (κάρτα υψηλής ταχύτητας)
- Κύκλωμα μετατροπής επιπέδου σανίδας, δηλαδή το επίπεδο διεπαφής για 5V ή 3.3V
- Η τροφοδοσία είναι 4,5V ~ 5,5V, πλακέτα κυκλώματος ρυθμιστή τάσης 3,3V
- Η διεπαφή επικοινωνίας είναι μια τυπική διεπαφή SPI
- Τρύπες τοποθέτησης βιδών M2 για εύκολη εγκατάσταση

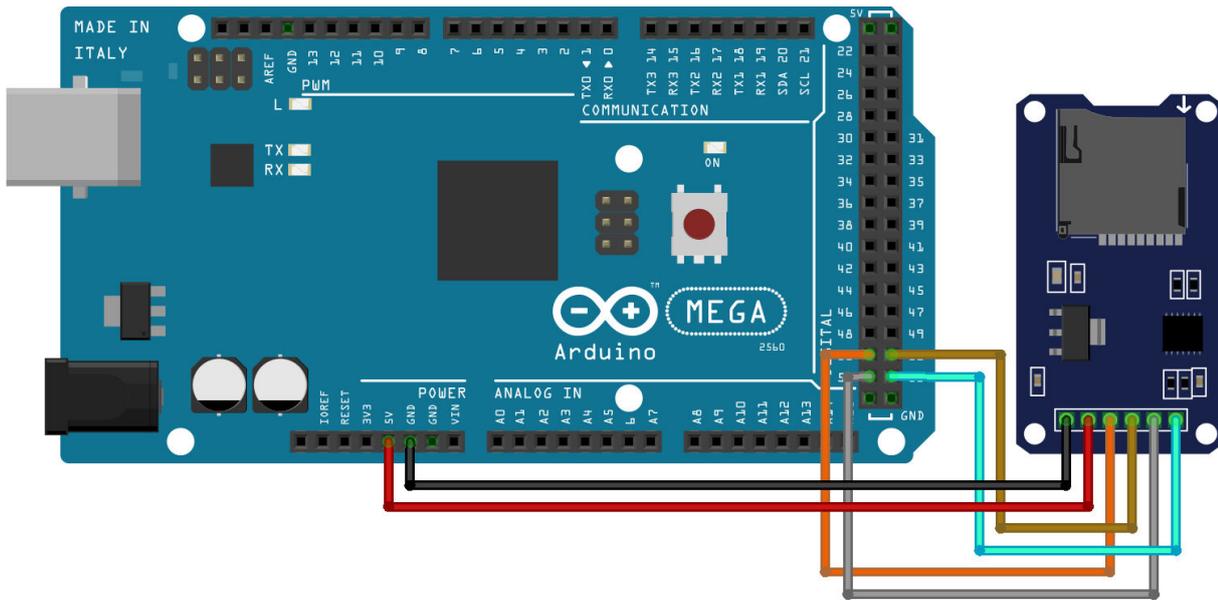
Περιγραφή

- **Διασύνδεση ελέγχου:** Συνολικά έξι ακίδες (GND, VCC, MISO, MOSI, SCK, CS), GND στη γείωση, VCC είναι η παροχή ρεύματος, MISO, MOSI, SCK για διάλογο SPI, το CS είναι ο ακροδέκτης σήματος επιλογής τσιπ.
- **Κύκλωμα ρυθμιστή 3.3V:** Έξοδος ρυθμιστή LDO 3.3V για τσιπ μετατροπής επιπέδου, τροφοδοσία κάρτας Micro SD.
- **Κύκλωμα μετατροπής επιπέδου:** Κάρτα Micro SD για να σηματοδοτήσει την κατεύθυνση των μετατροπών 3.3V, η διεπαφή κάρτας MicroSD για τον έλεγχο της κατεύθυνσης του σήματος MISO μετατρέπεται επίσης σε 3.3V, τα γενικά συστήματα μικροελεγκτών AVR μπορούν να διαβάσουν το σήμα.
- **Υποδοχή κάρτας Micro SD:** κατάστρωμα αυτο-βόμβας, εύκολη εισαγωγή κάρτας.
- **Τρύπες τοποθέτησης:** 4 βίδες M2 που τοποθετούν τρύπες με διάμετρο 2,2mm, έτσι ώστε η μονάδα να είναι εύκολη στην τοποθέτηση, για να επιτευχθεί συνδυασμός μεταξύ των μονάδων.



Εικόνα 3.13: SD-Card Module Pinout

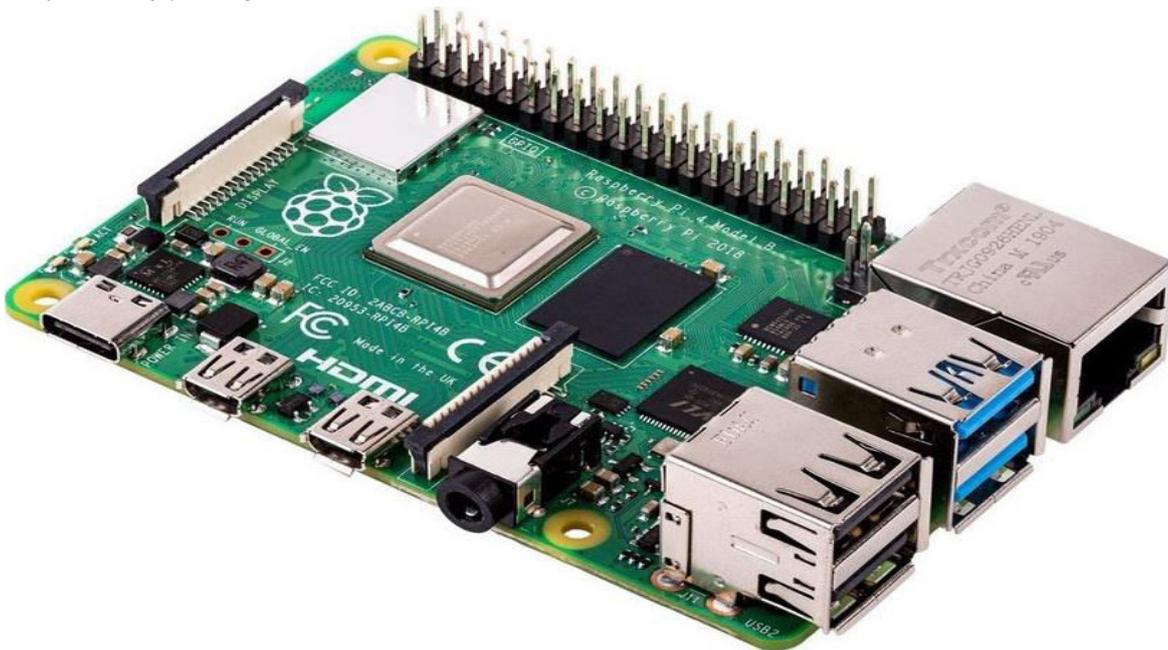
SD Card Module	Wiring Arduino Uno	Wiring Arduino Mega
VCC	3.3V / 5V	3.3V / 5V
CS	4	53
MOSI	11	51
CLK	13	52
MISO	12	50
GND	GND	GND



Εικόνα 3.14: Σύνδεση SD-Card Module

3.3 Raspberry Pi 4

Το Raspberry Pi 4 Model B είναι η τελευταία έκδοση του υπολογιστή χαμηλού κόστους Raspberry Pi . Το Pi δεν είναι σαν την τυπική συσκευή σας. στη φθηνότερη μορφή του δεν έχει θήκη και είναι απλώς μια ηλεκτρονική πλακέτα μεγέθους πιστωτικής κάρτας-του τύπου που μπορεί να βρείτε μέσα σε έναν υπολογιστή ή φορητό υπολογιστή, αλλά πολύ μικρότερη. Με το Pi 4 να είναι ταχύτερο, ικανό να αποκωδικοποιήσει βίντεο 4K, να επωφεληθεί από ταχύτερη αποθήκευση μέσω USB 3.0 και ταχύτερες συνδέσεις δικτύου μέσω πραγματικού Gigabit Ethernet, η πόρτα είναι ανοιχτή σε πολλές νέες χρήσεις. Είναι επίσης το πρώτο Pi που υποστηρίζει δύο οθόνες σε μία - έως και διπλές οθόνες 4K- ένα πλεονέκτημα για δημιουργικά που θέλουν περισσότερο χώρο στην επιφάνεια.εργασίας.



Εικόνα 3.15: Raspberry Pi 4

Το **Raspberry Pi** είναι ένας υπολογιστής χαμηλού κόστους και συνδέεται σε οθόνη υπολογιστή ή τηλεόραση και χρησιμοποιεί ένα τυπικό πληκτρολόγιο και ποντίκι. Είναι μια ικανή μικρή συσκευή που επιτρέπει σε άτομα κάθε ηλικίας να εξερευνήσουν τον υπολογιστή και να μάθουν πώς να προγραμματίζουν σε γλώσσες όπως το Scratch και το Python. Είναι σε θέση να κάνει ό, τι περιμένετε από έναν επιτραπέζιο υπολογιστή, από την περιήγηση στο διαδίκτυο και την αναπαραγωγή βίντεο υψηλής ευκρίνειας, έως την κατασκευή υπολογιστικών φύλλων, επεξεργασίας κειμένου και παιχνιδιών.

Επιπλέον, το Raspberry Pi έχει τη δυνατότητα να αλληλεπιδρά με τον έξω κόσμο και έχει χρησιμοποιηθεί σε μια ευρεία γκάμα έργων ψηφιακών δημιουργών, από μουσικές μηχανές και ανιχνευτές γονέων έως μετεωρολογικούς σταθμούς.

Λειτουργικό

Το **Raspberry Pi OS** (πρώην Raspbian) είναι ένα λειτουργικό σύστημα με βάση το **Debian** για το Raspberry Pi . Από το 2015, παρέχεται επίσημα από το ίδρυμα ως το κύριο λειτουργικό σύστημα για την οικογένεια συμπαγών υπολογιστών μονής πλακέτας Raspberry Pi . Η πρώτη έκδοση του Raspbian δημιουργήθηκε από τον Mike Thompson και τον Peter Green Η αρχική κατασκευή ολοκληρώθηκε τον Ιούνιο του 2012.

Το Raspberry Pi OS είναι εξαιρετικά βελτιστοποιημένο για τη σειρά Raspberry Pi με επεξεργαστές ARM . Λειτουργεί με κάθε Raspberry Pi εκτός από τον μικροελεγκτή Pico. Το Raspberry Pi OS χρησιμοποιεί ένα τροποποιημένο LXDE ως περιβάλλον εργασίας με τον διαχειριστή παραθύρων στοίβαξης Openbox , μαζί με ένα μοναδικό θέμα. Η διανομή αποστέλλεται με ένα αντίγραφο του προγράμματος άλγεβρας Wolfram Mathematica και μια έκδοση του Minecraft που ονομάζεται *Minecraft: Pi Edition* , καθώς και μια ελαφριά έκδοση του προγράμματος περιήγησης Chromium

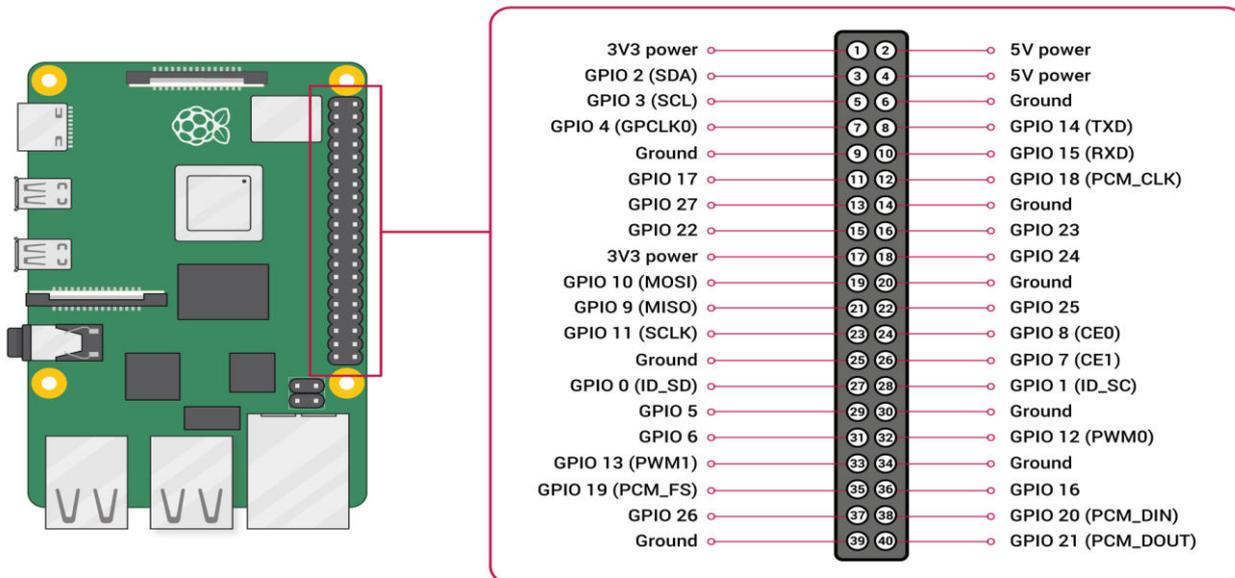
Χαρακτηριστικά

- **System-on-a-chip:** Broadcom BCM2711
- **ΕΠΕΞΕΡΓΑΣΤΗΣ:** Τετραπύρηνος επεξεργαστής 1.5GHz Arm Cortex-A72
- **GPU:** VideoCore VI
- **Μνήμη:** 1/2/4 GB LPDDR4 RAM
- **Συνδεσιμότητα:** 802.11ac Wi-Fi / Bluetooth 5.0, Gigabit Ethernet
- **Βίντεο και ήχος:** 2 x θύρες micro-HDMI που υποστηρίζουν οθόνες 4K@60Hz μέσω HDMI 2.0, θύρα οθόνης MIPI DSI, θύρα κάμερας MIPI CSI, στερεοφωνική έξοδο 4 πόλων και θύρα σύνθετου βίντεο
- **Θύρες:** 2 x USB 3.0, 2 x USB 2.0
- **Ισχύς:** 5V/3A μέσω USB-C, 5V μέσω κεφαλίδας GPIO
- **Επεκτασιμότητα:** κεφαλίδα GPIO 40 ακίδων

Τροφοδοσία

Η καλύτερη επιλογή είναι το επίσημο τροφοδοτικό Raspberry Pi Foundation USB Type-C , το οποίο έχει βαθμολογία 5,1V/3A.

Pinout



Εικόνα 3.16: Raspberry Pi 4 Pinout

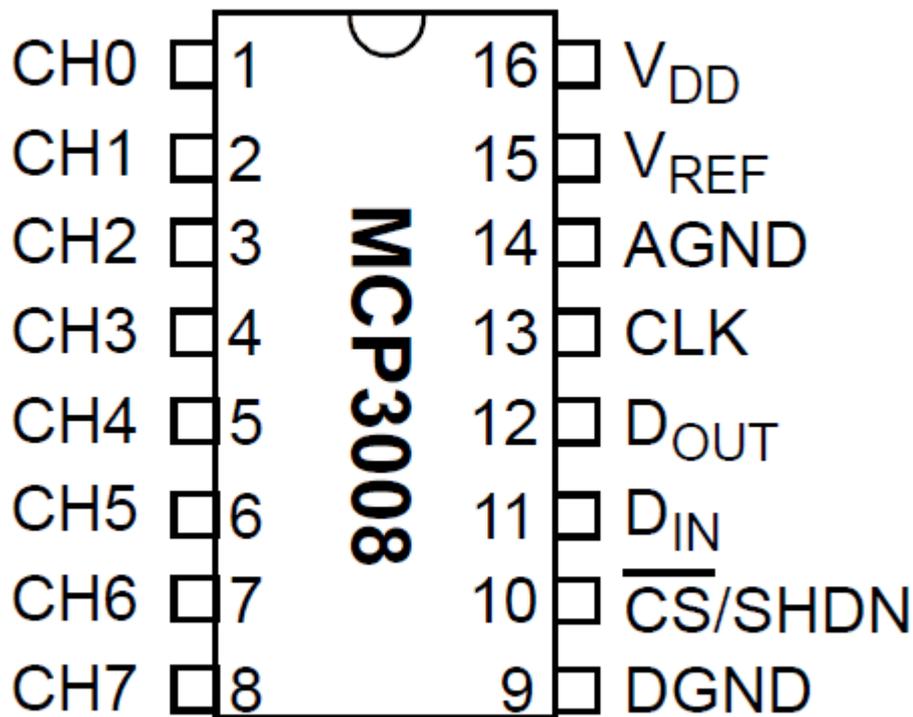
3.3.1 AnalogToDigital Converter

Το **MCP3008** είναι ένας χαμηλού κόστους μετατροπέας αναλογικού σε ψηφιακό 8 καναλιών 10 bit. Η ακρίβεια αυτού του ADC είναι παρόμοια με αυτή ενός Arduino Uno και με 8 κανάλια μπορείτε να διαβάσετε αρκετά αναλογικά σήματα από το Pi. Αυτό το τσιπ είναι μια εξαιρετική επιλογή εάν χρειάζεται απλώς να διαβάσετε απλά αναλογικά σήματα, όπως από αισθητήρα θερμοκρασίας ή φωτός.

Καλωδίωση

Το MCP3008 συνδέεται με το Raspberry Pi χρησιμοποιώντας σειριακή σύνδεση SPI. Χρησιμοποιούμε είτε το δίαυλο υλικού SPI είτε τέσσερις ακίδες GPIO και SPI λογισμικού για να μιλήσετε με το MCP3008. Το λογισμικό SPI είναι λίγο πιο ευέλικτο αφού μπορεί να λειτουργήσει με οποιαδήποτε ακίδα στο Pi, ενώ το SPI υλικού είναι ελαφρώς γρηγορότερο αλλά λιγότερο ευέλικτο επειδή λειτουργεί μόνο με συγκεκριμένες ακίδες.

Ο προσανατολισμός του τσιπ έχει σημασία! Φροντίστε να το τοποθετήσετε με την εσοχή μισού κύκλου και να τελειώνει προς την κορυφή. Μόλις το τσιπ είναι στο breadboard, τότε είστε έτοιμοι να το συνδέσετε στο Pi. Κάθε ένα από τα πόδια του τσιπ MCP3008 έχει τα ακόλουθα ονόματα:



Εικόνα 3.17: ADC-MCP3008 Pinout

Λογισμικό SPI

Για να συνδέσετε το MCP3008 στο Raspberry Pi με σύνδεση λογισμικού SPI, πρέπει να κάνετε τις ακόλουθες συνδέσεις:

- **MCP3008 VDD** σε **Raspberry Pi 3.3V**
- **MCP3008 VREF** έως **Raspberry Pi 3.3V**
- **MCP3008 AGND** στο **Raspberry Pi GND**
- **MCP3008 DGND** σε **Raspberry Pi GND**
- **MCP3008 CLK** έως **Raspberry Pi pin 18**
- **MCP3008 DOUT** στο **Raspberry Pi pin 23**
- **MCP3008 DIN** έως **Raspberry Pi pin 24**
- **MCP3008 CS/SHDN** to **Pin Raspberry Pi 25**

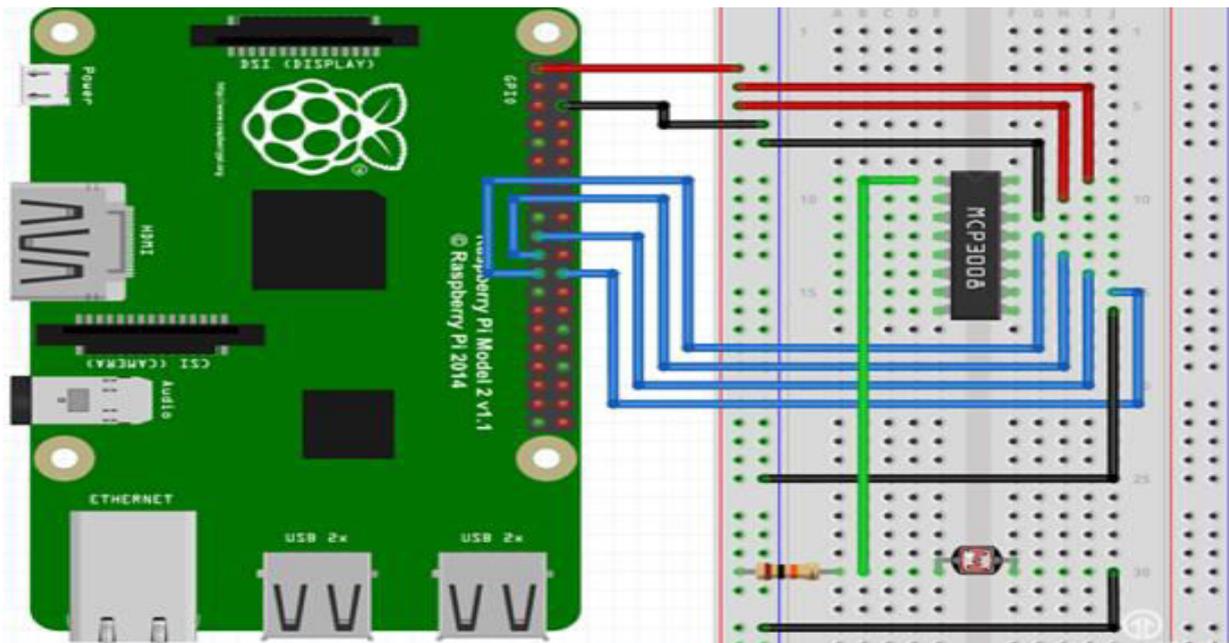
Σημείωση, μπορούμε να αλλάξουμε τις ακίδες MCP3008 CLK, DOUT, DIN και CS/SHDN σε οποιαδήποτε άλλη ψηφιακή ακίδα GPIO στο Raspberry Pi. Απλώς πρέπει να τροποποιήσουμε τον κώδικα για να τις χρησιμοποιήσουμε.

Hardware SPI

Για να χρησιμοποιήσουμε SPI υλικού, πρώτα πρέπει έχουμε ενεργοποιήσει το SPI χρησιμοποιώντας το εργαλείο διαμόρφωσης **raspi** Με την εντολή **raspi-config**.

MCP3008 στο Raspberry Pi ως εξής:

- **MCP3008 VDD** σε **Raspberry Pi 3.3V**
- **MCP3008 VREF** έως **Raspberry Pi 3.3V**
- **MCP3008 AGND** στο **Raspberry Pi GND**
- **MCP3008 DGND** σε **Raspberry Pi GND**
- **MCP3008 CLK** σε **Raspberry Pi SCLK**
- **MCP3008 DOUT** στο **Raspberry Pi MISO**
- **MCP3008 DIN** σε **Raspberry Pi MOSI**
- **MCP3008 CS/SHDN** έως **Raspberry Pi CE0**



Εικόνα 3.18: ADC-MCP3008 με αναλογική είσοδο ένα Photoresistor

Εγκατάσταση πηγής

Για εγκατάσταση πηγής από το Github συνδεόμαστε σε ένα τερματικό στο Raspberry Pi και εκτελούμε τις ακόλουθες εντολές:

1. `sudo apt-get update`
2. `sudo apt-get install build-essential python-dev python-smbus git`
3. `cd ~`
4. `git clone https://github.com/adafruit/Adafruit_Python_MCP3008.git`
5. `cd Adafruit_Python_MCP3008`
6. `sudo python setup.py install`

Ολοκληρωμένος οδηγός εγκατάστασης και βασικής αλληλεπίδρασης με το Raspberry Pi 4 ο οποίος ακολουθήθηκε βήμα-βήμα για την πρώτη επαφή με τον μικροελεγκτή υπάρχει στο παρακάτω link:

<https://projects.raspberrypi.org/el-GR/projects/raspberry-pi-using/9>

3.4 ECUs με ELM 327 ,OBD II & Πρωτόκολλα

Το ELM327 είναι ένας προγραμματισμένος μικροελεγκτής που παράγεται από την ELM Electronics για τη μετάφραση της διεπαφής διάγνωσης επί του σκάφους (OBD) που βρίσκεται στα περισσότερα σύγχρονα αυτοκίνητα. Το πρωτόκολλο εντολών ELM327 είναι ένα από τα πιο δημοφιλή πρότυπα διεπαφής PC-to-OBD και εφαρμόζεται επίσης από άλλους προμηθευτές.

Το αρχικό ELM327 εφαρμόζεται στον μικροελεγκτή PIC18F2480 από την τεχνολογία Microchip .

Το ELM327 ανήκει σε μια οικογένεια μεταφραστών OBD από την ELM Electronics. Άλλες παραλλαγές υλοποιούν μόνο ένα υποσύνολο των πρωτοκόλλων OBD.



Εικόνα 3.19: ELM327 Microcontroller

Χρήσεις

Το ELM327 αφαιρεί το πρωτόκολλο χαμηλού επιπέδου και παρουσιάζει μια απλή διεπαφή που μπορεί να κληθεί μέσω UART , συνήθως μέσω χειροκίνητου διαγνωστικού εργαλείου ή προγράμματος υπολογιστή που συνδέεται με USB , RS-232 , Bluetooth ή Wi-Fi.

Υπάρχει μεγάλος αριθμός διαθέσιμων προγραμμάτων που συνδέονται με το ELM327. Η λειτουργία αυτού του λογισμικού μπορεί να περιλαμβάνει συμπληρωματικά όργανα οχήματος, αναφορά και εκκαθάριση κωδικών σφαλμάτων

Λειτουργίες

- Διαγνωστικοί κωδικοί προβλημάτων, γενικοί και ειδικοί για τον κατασκευαστή.
- Διαγράφη ορισμένων κωδικών-προβλημάτων και απενεργοποίηση του MIL ("Ενδειξη ενδείξεων δυσλειτουργίας", πιο γνωστό ως "Έλεγχος φωτισμού κινητήρα")
- Εμφάνιση δεδομένων τρέχοντος αισθητήρα
- RPM κινητήρα
- Υπολογιζόμενη αξία φορτίου
- Θερμοκρασία ψυκτικού
- Κατάσταση συστήματος καυσίμου
- Ταχύτητα οχήματος
- Βραχυπρόθεσμη περικοπή καυσίμου
- Μακροπρόθεσμη περικοπή καυσίμου
- Πίεση πολλαπλής εισαγωγής
- Timing Advance
- Θερμοκρασία αέρα εισαγωγής
- Ποσοστό ροής αέρα
- Απόλυτη θέση γκαζιού
- Τάσεις αισθητήρα οξυγόνου/σχετικές βραχυπρόθεσμες περικοπές καυσίμου
- Κατάσταση συστήματος καυσίμου
- Πίεση καυσίμου

Τα διαγνωστικά επί του οχήματος (OBD)

Είναι ένας όρος αυτοκινήτου που αναφέρεται στην ικανότητα αυτοδιάγνωσης και αναφοράς ενός οχήματος. Τα συστήματα OBD δίνουν στον ιδιοκτήτη ή τον τεχνικό επισκευής πρόσβαση στην κατάσταση των διαφόρων υποσυστημάτων οχημάτων. Ο όγκος των διαγνωστικών πληροφοριών που διατίθενται μέσω του OBD έχει μεγάλη ποικιλία από την εισαγωγή του στις αρχές της δεκαετίας του 1980 εκδόσεις υπολογιστών οχημάτων επί του σκάφους. Οι πρώτες εκδόσεις του OBD θα φωτίζουν απλώς μια ενδεικτική λυχνία δυσλειτουργίας ή ένα "ηλίθιο φως" εάν εντοπιστεί πρόβλημα, αλλά δεν θα παρέχουν πληροφορίες σχετικά με τη φύση του προβλήματος. Οι σύγχρονες εφαρμογές OBD χρησιμοποιούν τυποποιημένη ψηφιακή θύρα επικοινωνιών για να παρέχουν δεδομένα σε πραγματικό χρόνο, επιπλέον μιας τυποποιημένης σειράς διαγνωστικών κωδικών προβλημάτων, ή DTC, που επιτρέπουν σε ένα άτομο να εντοπίζει γρήγορα και να διορθώνει δυσλειτουργίες εντός του οχήματος.

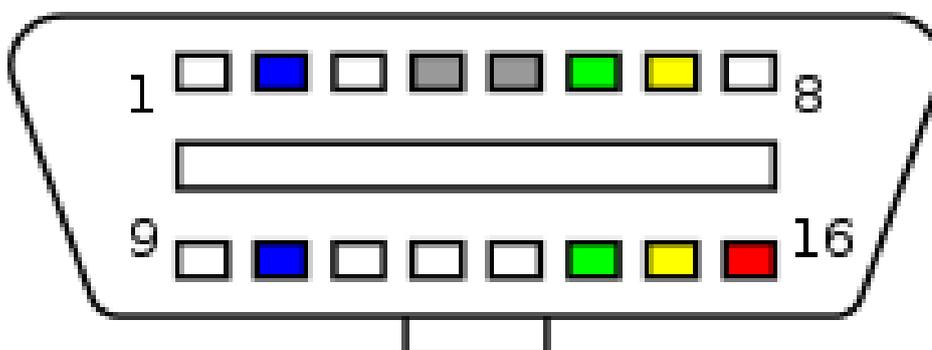
Το OBD-II είναι μια βελτίωση έναντι του OBD-I τόσο στην ικανότητα όσο και στην τυποποίηση. Το πρότυπο OBD-II καθορίζει τον τύπο του διαγνωστικού συνδέσμου και το pinout του, τα διαθέσιμα πρωτόκολλα ηλεκτρικής σηματοδότησης και τη μορφή μηνυμάτων.

Παρέχει επίσης μια υποψήφια λίστα παραμέτρων του οχήματος για παρακολούθηση μαζί με τον τρόπο κωδικοποίησης των δεδομένων για το καθένα. Υπάρχει ένας πείρος στο βύσμα που παρέχει ισχύ για το εργαλείο σάρωσης από την μπαταρία του οχήματος, γεγονός που εξαλείφει την ανάγκη ξεχωριστής σύνδεσης ενός εργαλείου σάρωσης σε μια πηγή ενέργειας. Ωστόσο, ορισμένοι τεχνικοί ενδέχεται να εξακολουθούν να συνδέουν το εργαλείο σάρωσης σε μια βοηθητική πηγή τροφοδοσίας για την προστασία των δεδομένων σε ασυνήθιστη περίπτωση που ένα όχημα παρουσιάσει απώλεια ηλεκτρικής ενέργειας λόγω δυσλειτουργίας. Τέλος, το πρότυπο OBD-II παρέχει μια επεκτάσιμη λίστα DTC. Ως αποτέλεσμα αυτής της τυποποίησης, μια μεμονωμένη συσκευή μπορεί να αναζητήσει τους ενσωματωμένους υπολογιστές σε οποιοδήποτε όχημα. Αυτό το OBD-II κυκλοφόρησε σε δύο μοντέλα OBD-IIA και OBD-IIB. Η τυποποίηση του OBD-II προήλθε από τις απαιτήσεις εκπομπών και παρόλο που απαιτείται να διαβιβάζονται μόνο κωδικοί και δεδομένα που σχετίζονται με τις εκπομπές, οι περισσότεροι κατασκευαστές έχουν κάνει το OBD-II Το **Data Link Connector** είναι το μόνο στο όχημα μέσω του οποίου διαγιγνώσκονται και προγραμματίζονται όλα τα συστήματα. Οι κωδικοί βλάβης διαγνωστικού ελέγχου OBD-II είναι τετραψήφιοι, ενώ προηγείται ένα γράμμα: P για κινητήρα και κιβώτιο ταχυτήτων (κινητήρες), B για αμάξωμα, C για σασί και U για δίκτυο.

Διαγνωστική υποδοχή OBD-II

Η προδιαγραφή OBD-II παρέχει μια τυποποιημένη διεπαφή υλικού-τη θηλυκή υποδοχή J1962 16 ακίδων (2x8). Σε αντίθεση με το βύσμα OBD-I, το οποίο βρέθηκε μερικές φορές κάτω από το καπό του οχήματος, ο σύνδεσμος OBD-II απαιτείται να βρίσκεται σε απόσταση 0,6 μέτρων από το τιμόνι (εκτός εάν ζητηθεί εξαίρεση από τον κατασκευαστή, σε περίπτωση που βρίσκεται ακόμα κάπου κοντά στον οδηγό).

Το SAE J1962 ορίζει το pinout του συνδέσμου ως εξής:



Εικόνα 3.20: Θηλυκό pinout σύνδεσης OBD-II

1	Διακριτικό Κατασκευαστή	9	Διακριτικό Κατασκευαστή
	GM: J2411 GMLAN/SWC/Single-WAR CAN.		GM: 8192 baud ALDL όπου υπάρχει.
	VW/Audi: Εναλλαγή +12 για να πει ένα εργαλείο σάρωσης εάν η ανάφλεξη είναι ενεργοποιημένη.		BMW και Toyota: σήμα RPM.
2	Θετική γραμμή διαύλου SAE J1850 PWM και VPW	10	Αρνητική γραμμή διαύλου μόνο SAE J1850 PWM (όχι SAE 1850 VPW)
3	Διακριτικό Κατασκευαστή	11	Διακριτικό Κατασκευαστή
	Ford DCL (+) Αργεντινή, Βραζιλία (προ OBD-II) 1997–2000, ΗΠΑ, Ευρώπη κ.λπ.		Ford DCL (-) Αργεντινή, Βραζιλία (προ OBD-II) 1997–2000, ΗΠΑ, Ευρώπη κ.λπ.
	Chrysler CCD Bus (+)		Chrysler CCD Bus (-)
	Ethernet TX+(Διαγνωστικά μέσω IP)		Ethernet TX- (Διαγνωστικά μέσω IP)
4	Γείωση σασί (GND)	12	Μη συνδεδεμένο
			Διακριτικό Κατασκευαστή
			Ethernet RX+ (Διαγνωστικά μέσω IP)
5	Γείωση σήματος (GND)	13	Διακριτικό Κατασκευαστή
			Ford: FEPS - Προγραμματισμός τάσης PCM
			Ethernet RX- (Διαγνωστικά μέσω IP)
6	CAN High (ISO 15765-4 και SAE J2284)	14	CAN Low (ISO 15765-4 και SAE J2284)
7	K-γραμμή ISO 9141-2 και ISO 14230-4	15	Γραμμή L ISO 9141-2 και ISO 14230-4
8	Διακριτικό Κατασκευαστή	16	Τάση μπαταρίας (12V)
	Πολλές BMW: Μια δεύτερη γραμμή K για μη OBD-II συστήματα (αμάξωμα/πλαίσιο/πληροφορίες ψυχαγωγίας).		
	Ενεργοποίηση Ethernet (Διαγνωστικά μέσω IP)		

Πρωτόκολλα

Τα πρωτόκολλα που υποστηρίζονται από το ELM327 μέσω του OBD II είναι:

- SAE J1850 PWM (41,6 kbit/s)
- SAE J1850 VPW (10,4 kbit/s)
- ISO 9141-2 (5 baud init, 10,4 kbit/s)
- ISO 14230-4 KWP (5 baud init, 10,4 kbit/s)
- ISO 14230-4 KWP (γρήγορη εκκίνηση, 10,4 kbit/s)
- ISO 15765-4 CAN (11 bit ταυτότητας, 500 kbit/s)
- ISO 15765-4 CAN (29 bit ταυτότητας, 500 kbit/s)
- ISO 15765-4 CAN (11 bit ταυτότητας, 250 kbit/s)
- ISO 15765-4 CAN (29 bit ταυτότητας, 250 kbit/s)
- SAE J1939 (250kbit/s)
- SAE J1939 (500kbit/s)

Κωδικοί βλάβης

Κάθε ένας από τους κωδικούς σφάλματος OBD αποτελείται από πέντε χαρακτήρες: ένα γράμμα, ακολουθούμενο από τέσσερις αριθμούς. Το γράμμα αναφέρεται στο σύστημα που ανακρίνεται π.χ. το Pxxxx θα αναφέρεται στο σύστημα μετάδοσης κίνησης. Ο επόμενος χαρακτήρας θα είναι 0 αν συμμορφώνεται με το πρότυπο OBD. Θα πρέπει λοιπόν να μοιάζει με P0xxx.

Ο επόμενος χαρακτήρας θα αναφέρεται στο υποσύστημα.

- P00xx - Μέτρηση καυσίμου και αέρα και έλεγχοι βοηθητικών εκπομπών.
- P01xx - Μέτρηση καυσίμου και αέρα.
- P02xx - Μέτρηση καυσίμου και αέρα (Κύκλωμα μπεκ ψεκασμού).
- P03xx - Σύστημα ανάφλεξης ή σφάλμα.
- P04xx - Βοηθητικοί έλεγχοι εκπομπών.
- P05xx - Έλεγχος ταχύτητας οχήματος και σύστημα ελέγχου ρελαντί.
- P06xx - Κύκλωμα εξόδου υπολογιστή.
- P07xx - Μετάδοση.
- P08xx - Μετάδοση.

3.4.1 OBD II - PID Commands

Οι εντολές OBD αποτελούνται από δεκαεξαδικούς κωδικούς γραμμένους με χαρακτήρες ASCII. Γενικά, αυτές οι εντολές περιέχουν 2 ή περισσότερα ζεύγη δεκαεξαδικών αριθμών, ωστόσο υπάρχουν μερικές εντολές που απαιτούν μόνο ένα εξάγωνο ζεύγος.

Το πρώτο hex ζεύγος στην εντολή OBD αντιπροσωπεύει τη λειτουργία OBD που πρέπει να χρησιμοποιηθεί. Τυχόν επόμενα ζεύγη hex μετά από αυτό αντιπροσωπεύουν το αναγνωριστικό παραμέτρου (PID) που πρέπει να διαβαστεί από την καθορισμένη λειτουργία. Υπάρχουν 10 λειτουργίες OBD, αλλά δεν θα χρησιμοποιούν όλα τα οχήματα και τις 10 λειτουργίες.

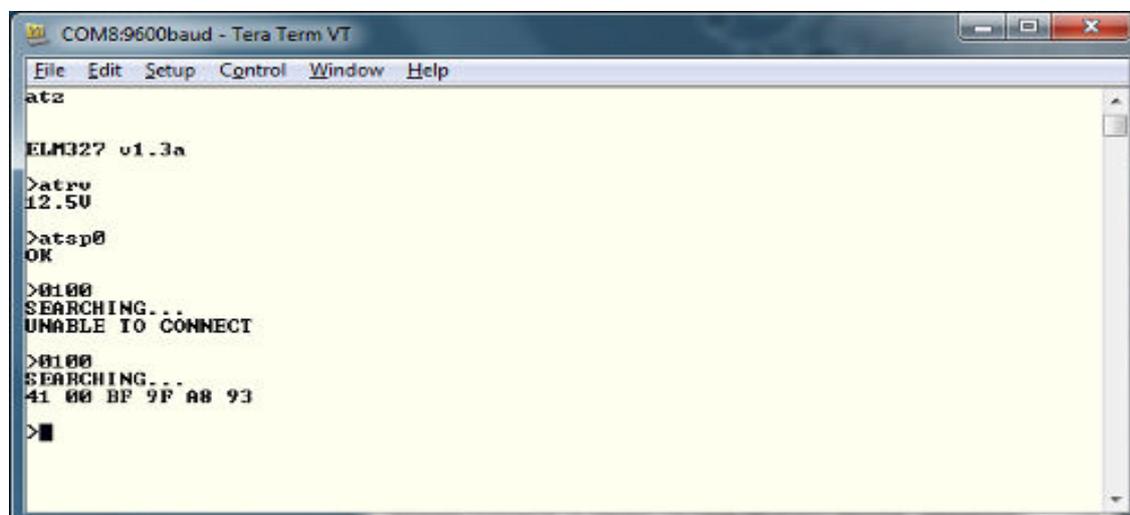
Τα πρωτόκολλα του κάθε οχήματός διαφέρουν για να δούμε ποιες λειτουργίες OBD και αναγνωριστικά παραμέτρων υποστηρίζονται χρειάζεται έλεγχος.

Αναλυτή λίστα σχετικά με τη λειτουργία των OBD PIDs υπάρχει στη [Wikipedia](https://en.wikipedia.org/wiki/OBD-II_PIDs) η οποία χρησιμοποιήθηκε συμβουλευτικά.

https://en.wikipedia.org/wiki/OBD-II_PIDs

Αριθμός λειτουργίας	Περιγραφή λειτουργίας
01	Τρέχοντα Δεδομένα
02	Πάγωμα δεδομένων πλαισίου
03	Κωδικοί διαγνωστικών προβλημάτων
04	Διαγραφή κώδικα προβλήματος
05	Αποτελέσματα δοκιμών/Αισθητήρες οξυγόνου
06	Αποτελέσματα δοκιμών/Μη συνεχής δοκιμή
07	Εμφάνιση κωδικών προβλημάτων σε εκκρεμότητα
08	Ειδική λειτουργία ελέγχου
09	Ζητήστε πληροφορίες οχήματος
0A	Αίτηση μόνιμων κωδικών προβλημάτων

Ίσως το πιο σημαντικό PID είναι το 00. Αυτό λειτουργεί σε οποιοδήποτε όχημα υποστηρίζει OBD και δίνει μια λίστα με άλλα PID που υποστηρίζει το αυτοκίνητο. Στο παράθυρο του τερματικού μας πληκτρολογούμε "0100" και πατάμε "enter". Αυτή η εντολή μεταφράζεται σε "Στη λειτουργία 01, ποια PID υποστηρίζονται;"

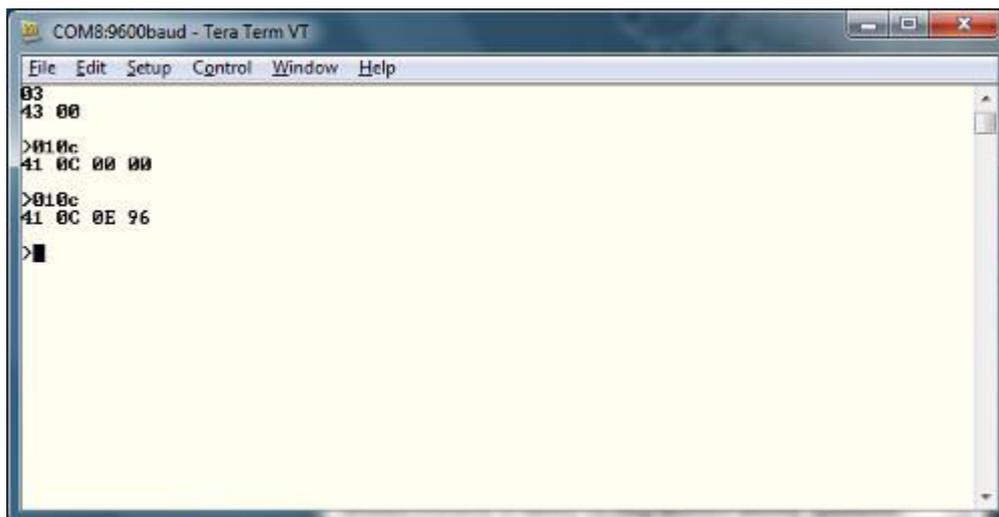


```
COM8:9600baud - Tera Term VT
File Edit Setup Control Window Help
atz
ELM327 v1.3a
>atrv
12.50
>atsp0
OK
>0100
SEARCHING...
UNABLE TO CONNECT
>0100
SEARCHING...
41 00 BF 9F AB 93
>
```

Εικόνα 3.21: OBD-II & PIDs

Υπάρχει μια γενική δομή που όλες οι απαντήσεις του OBD έχουν κοινό. Το πρώτο byte απόκρισης (στην περίπτωση αυτή 0x41) παραθέτει τη λειτουργία που ζητήθηκε στην εντολή. Έτσι ο πίνακας στέλνει 0x40 + 0x01. Το δεύτερο byte είναι η παράμετρος που ζητήθηκε, οπότε στην περίπτωσή μας, 0x00. Τα ακόλουθα byte είναι οι απαντήσεις στην εντολή. Σε αυτήν την περίπτωση, τα byte 0xBF, 0x9F, 0xA8 και 0x93 είναι τα PID που υποστηρίζονται από το όχημα.

Μια άλλη συνήθως υποστηριζόμενη παράμετρος είναι η «Ανάγνωση στροφών ανά λεπτό». Πληκτρολογούμε την εντολή "010C" και πατάμε "enter". Το Board θα απαντήσει με μια τιμή που αναφέρεται στο δεκαεξαδικό.



Εικόνα 3.22: OBD-II & PIDs

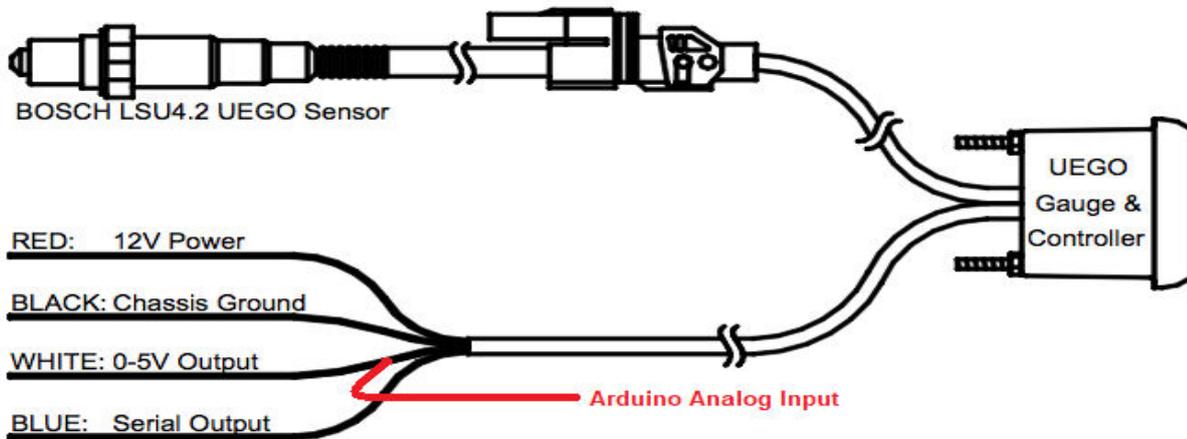
Η δομή απόκρισης είναι η ίδια όπως πριν. 0x41 για να δηλώσουμε ότι το Board είναι στη λειτουργία 01, ακολουθούμενο από 0x0C για να δείξουμε ότι το Board κοιτάζει την παράμετρο RPM. Η επιστρεφόμενη τιμή 0x0E 0x96 μπορεί στη συνέχεια να μετατραπεί σε δεκαδική τιμή 3734. Αυτό είναι στην πραγματικότητα 4 φορές το πραγματικό RPM, καθώς αυτή η τιμή παρατίθεται σε τέταρτα RPM. Μόλις η τιμή διαιρεθεί με 4, έχουμε ένα RPM στο ρελαντί 933.

3.5 After-Market Όργανα & Αισθητήρες

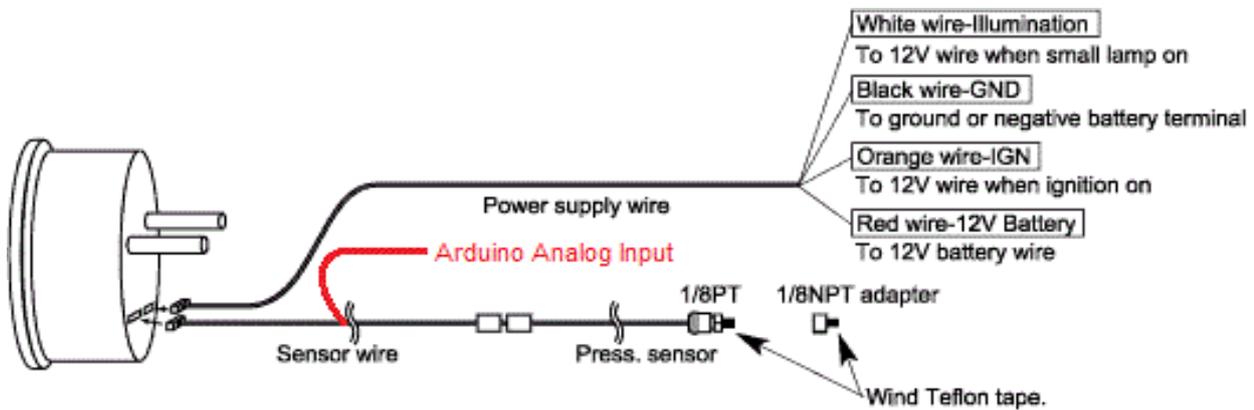
Τα **After-Market Όργανα** μετά την αγορά του είναι συχνά μια αποτελεσματική στρατηγική για τη συμπλήρωση του πίνακα οργάνων του οχήματός. Είτε για να έχουμε όσο το δυνατόν περισσότερες επιδόσεις ή απλώς για να ελαχιστοποιήσουμε τις πιθανότητες σας για καταστροφική μηχανική βλάβη, η εγκατάσταση ενός οργάνου μετά την αγορά μπορεί να είναι η σωστή προσέγγιση.

Όπως και με άλλα εξαρτήματα, οι μετρητές αυτοκινήτων μετά την αγορά, διαφέρουν ως προς την αξιοπιστία και την ποιότητα. Με τα κατάλληλα εργαλεία, έναν δείκτη ποιότητας και λίγη προσπάθεια, μπορούμε πιθανώς να λάβουμε τις πληροφορίες που χρειαζόμαστε για να αξιολογήσουμε στο έπακρο το όχημά μας. Στο πλαίσιο της εργασίας θα χρησιμοποιηθούν σήματα από όργανα και αισθητήρες που έχουν ήδη εγκατασταθεί στο όχημα ανάγοντας το σήμα εισόδου στο όργανο μέσω του αισθητήρα σε **Analog Input Value** για το Arduino.

Διαγράμματα καλωδίωσης

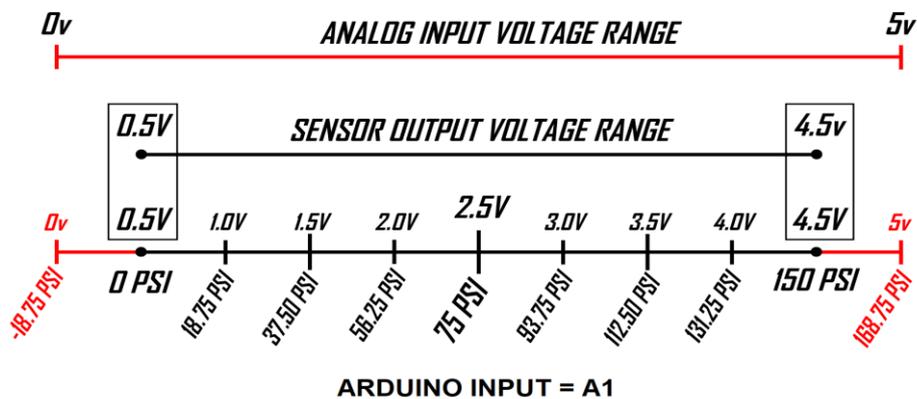


Εικόνα 3.23: Διάγραμμα Οργάνου A/F Ratio



Εικόνα 3.24: Διάγραμμα Οργάνου Πίεσης Λαδιού

ANALOG INPUT - (0V - 5V (MIN-MAX VALUE)) OIL PRESSURE SENSOR - (0.5V - 4.5V (0-150 PSI))



0.5V = 0 PSI / 4.5V 150 PSI

ANALOG INPUT LO/HI

0V = -19* PSI / 5V = 169* PSI

A1.LO = -19* A1.HI = 168*

Εικόνα 3.25: Κωδικοποίηση Αναλογικού Σήματος

3.6 ThinkSpeak

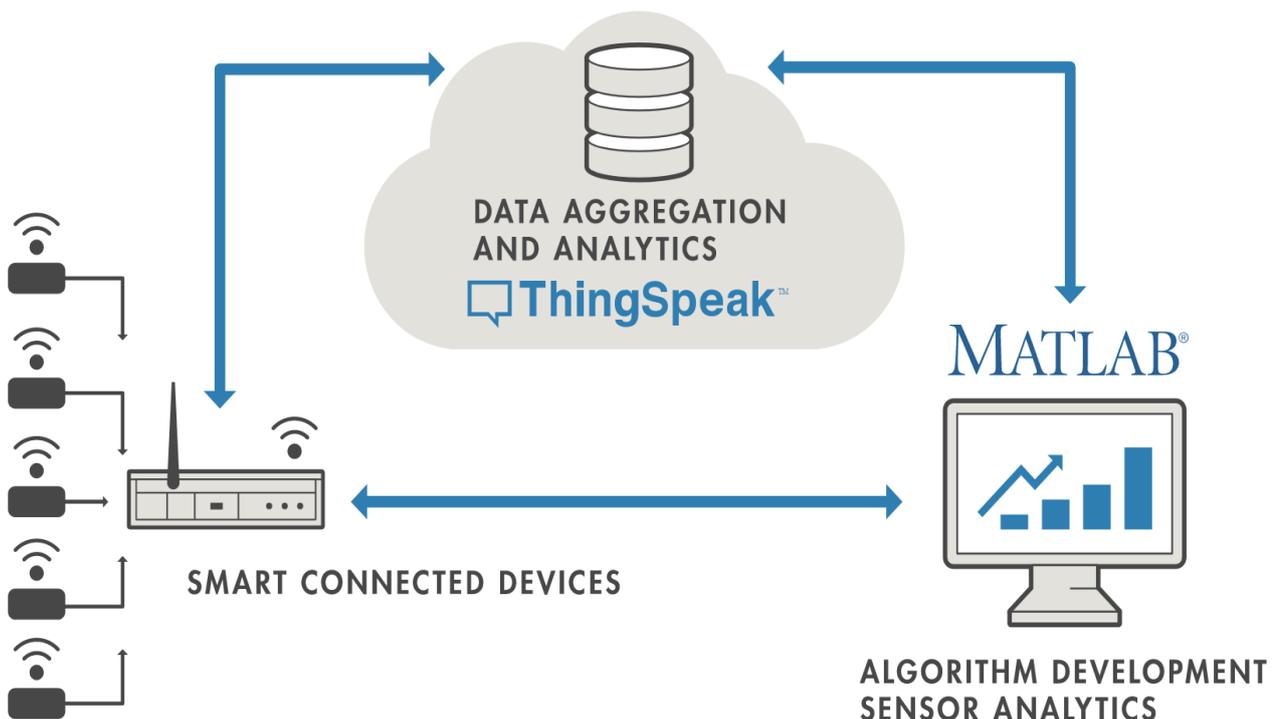
Το ThingSpeak είναι μια υπηρεσία πλατφόρμας αναλύσεων IoT που μας επιτρέπει να συγκεντρώνουμε, να οπτικοποιούμε και να αναλύουμε ζωντανές ροές δεδομένων στο cloud. Το ThingSpeak παρέχει άμεσες απεικονίσεις δεδομένων που δημοσιεύονται από τις συνδεδεμένες συσκευές μας. Με τη δυνατότητα εκτέλεσης κώδικα MATLAB® στο ThingSpeak, πραγματοποιούμε διαδικτυακή ανάλυση και επεξεργασία των δεδομένων όπως έρχονται. Το ThingSpeak χρησιμοποιείται συχνά για την δημιουργία πρωτοτύπου και την απόδειξη εννοιολογικών συστημάτων IoT που απαιτούν αναλύσεις.

Internet of Things (IoT)

Το Internet of Things (IoT) περιγράφει μια αναδυόμενη τάση όπου ένας μεγάλος αριθμός ενσωματωμένων συσκευών (αντικειμένων) συνδέονται στο Διαδίκτυο. Αυτές οι συνδεδεμένες συσκευές επικοινωνούν με ανθρώπους και άλλα πράγματα και συχνά παρέχουν δεδομένα αισθητήρων σε αποθηκευτικούς χώρους και πόρους υπολογιστικού νέφους όπου τα δεδομένα επεξεργάζονται και αναλύονται για να αποκτήσουν σημαντικές γνώσεις. Η φθηνή υπολογιστική ισχύς στο cloud και η αυξημένη συνδεσιμότητα συσκευών καθιστούν δυνατή αυτήν την τάση.

Οι λύσεις IoT έχουν δημιουργηθεί για πολλές κάθετες εφαρμογές όπως παρακολούθηση και έλεγχος περιβάλλοντος, παρακολούθηση της υγείας, παρακολούθηση στόλου οχημάτων, βιομηχανική παρακολούθηση και έλεγχος και αυτοματισμός σπιτιού.

Σε υψηλό επίπεδο, πολλά συστήματα IoT μπορούν να περιγραφούν χρησιμοποιώντας το παρακάτω διάγραμμα:



Εικόνα 3.26: ThinkSpeak Flowchart

Στα αριστερά, έχουμε τις έξυπνες συσκευές (τα «πράγματα» στο IoT) που ζουν στην άκρη του δικτύου. Αυτές οι συσκευές συλλέγουν δεδομένα και περιλαμβάνουν πράγματα όπως φορητές συσκευές, αισθητήρες ασύρματης θερμοκρασίας, οθόνες καρδιακών παλμών και αισθητήρες υδραυλικής πίεσης και μηχανές στο πάτωμα του εργοστασίου.

Στη μέση, έχουμε το σύννεφο όπου τα δεδομένα από πολλές πηγές συγκεντρώνονται και αναλύονται σε πραγματικό χρόνο, συχνά από μια πλατφόρμα ανάλυσης IoT σχεδιασμένη για αυτόν τον σκοπό.

Η δεξιά πλευρά του διαγράμματος απεικονίζει την ανάπτυξη αλγορίθμων που σχετίζεται με την εφαρμογή IoT. Εδώ ένας μηχανικός ή επιστήμονας δεδομένων προσπαθεί να αποκτήσει εικόνα για τα δεδομένα που συλλέγονται πραγματοποιώντας ιστορική ανάλυση των δεδομένων. Σε αυτήν την περίπτωση, τα δεδομένα αντλούνται από την πλατφόρμα IoT σε περιβάλλον λογισμικού για επιτραπέζιους υπολογιστές για να επιτρέψουν στον μηχανικό ή τον επιστήμονα να δημιουργήσει πρωτότυπα αλγόριθμους που μπορεί τελικά να εκτελεστούν στο cloud ή στην ίδια την έξυπνη συσκευή.

Ένα σύστημα IoT περιλαμβάνει όλα αυτά τα στοιχεία. Το ThingSpeak ταιριάζει στο τμήμα σύννεφων του διαγράμματος και παρέχει μια πλατφόρμα για τη γρήγορη συλλογή και ανάλυση δεδομένων από αισθητήρες συνδεδεμένους στο Διαδίκτυο.

Βασικά χαρακτηριστικά

Το **ThingSpeak** επιτρέπει να συγκέντρωση, να οπτικοποίηση και να ανάλυση σε ζωντανές ροές δεδομένων στο **cloud**. Μερικές από τις βασικές δυνατότητες του είναι:

- Εύκολη διαμόρφωση συσκευών για αποστολή δεδομένων στο ThingSpeak χρησιμοποιώντας δημοφιλή πρωτόκολλα IoT.
- Οπτικοποίηση για τα δεδομένα των αισθητήρων σε πραγματικό χρόνο.
- Συγκεντρωτικά δεδομένα κατά παραγγελία.
- Χρήση του MATLAB για να κατανόηση των δεδομένων στο IoT.
- Αυτόματη ανάλυση των στοιχείων IoT με βάση χρονοδιαγράμματα ή συμβάντα.
- Δημιουργία συστημάτων IoT χωρίς τη δημιουργία server ή την ανάπτυξη λογισμικού ιστού.

Βασικές Ρυθμίσεις

Για να δημιουργήσουμε ένα τελικό σημείο στο οποίο μπορούμε να στείλουμε κάποια δεδομένα για τη δοκιμή μας.

- Δημιουργούμε έναν λογαριασμό στο <https://thingspeak.com/>
- Δημιουργούμε ένα νέο κανάλι με μια ετικέτα πεδίου
- Αποκτούμε το κλειδί API
- Ανατρέχουμε στη διεύθυνση URL "Ενημέρωση ροής καναλιού"

Το ThingSpeak είναι εξαιρετικά εύκολο στη χρήση. Μόλις έχουμε έτοιμο το κανάλι και το κλειδί μας, μπορούμε απλά να υποβάλουμε ένα αίτημα HTTP στη διεύθυνση:

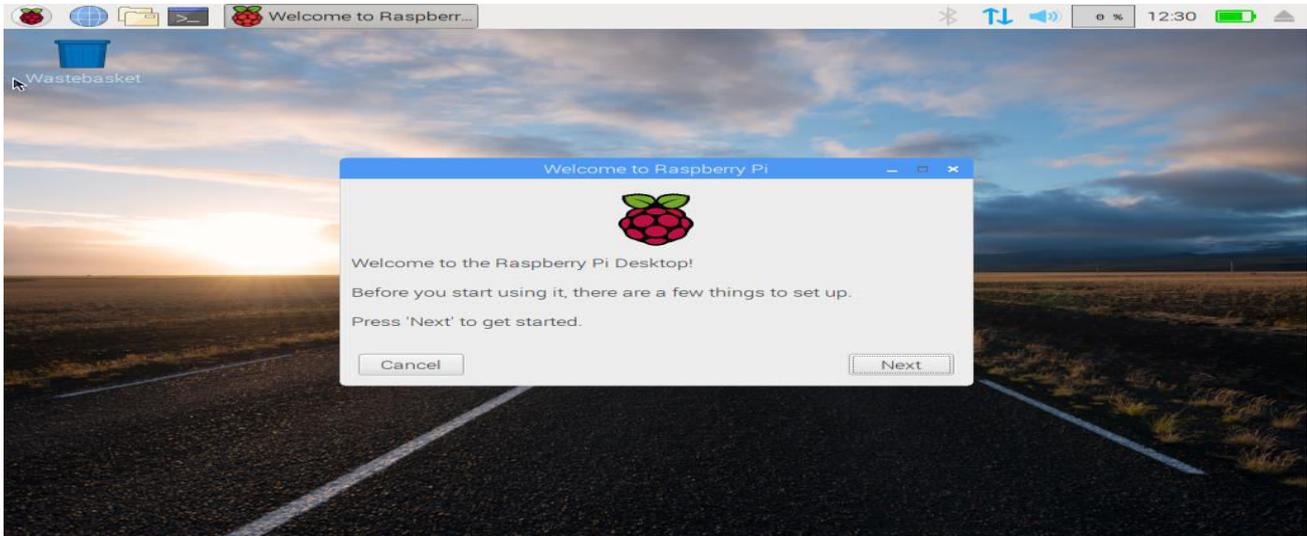
https://api.thingspeak.com/update?api_key='YOUR_KEY_HERE'&field1=4 για να στείλουμε την τιμή 4 στο πεδίο 1.

Έτσι έχουμε στα πεδία που έχουμε δημιουργήσει τις τιμές που θέλουμε με σχηματική ανάλυση σε χρονοδιάγραμμα και τελικά άμεσες απεικονίσεις δεδομένων.

Κεφάλαιο 4^ο - Υλοποίηση Συστήματος

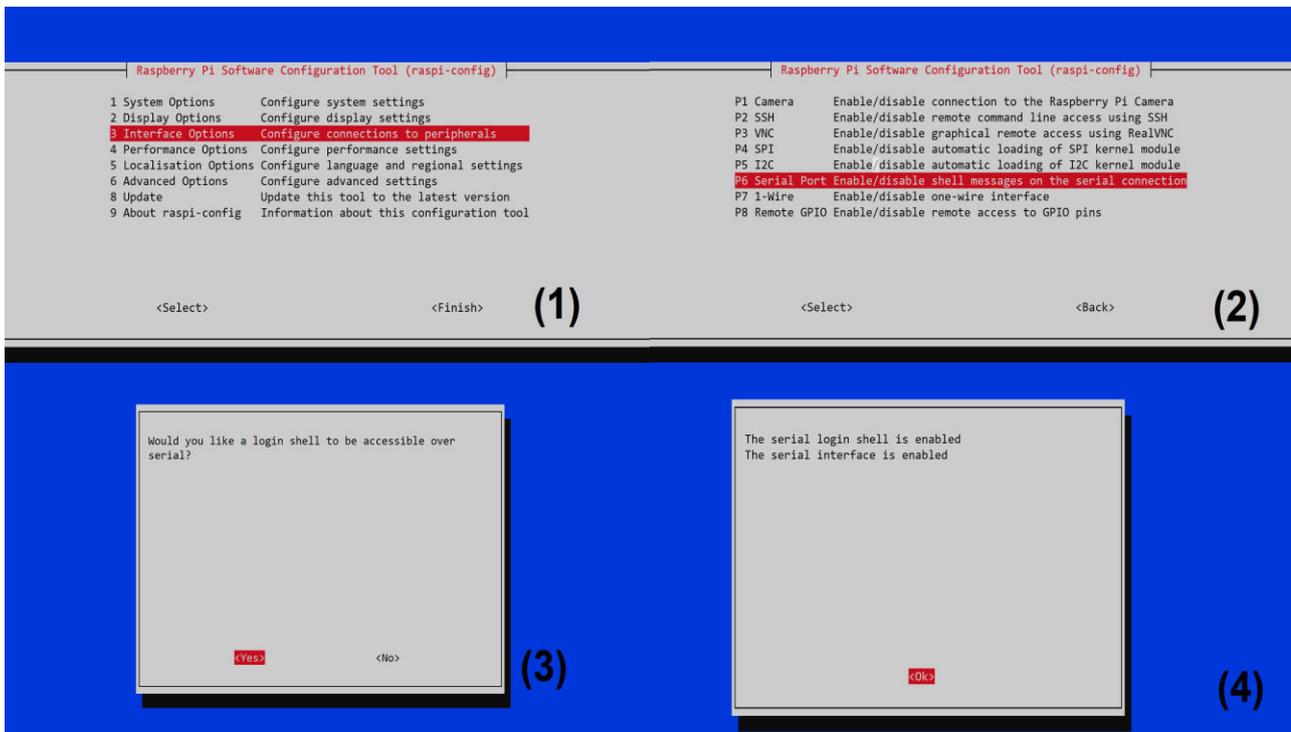
4.1 Σχεδίαση του (Car Real Time IoT Data Logger) Based On Raspberry Pi

Σε αρχικό στάδιο και με μια πρώτη μελέτη το ενσωματωμένο σύστημα παρακολούθησης είχε σαν κύριο μικροελεγκτή το Raspberry Pi 4. Έπειτα από το download του λογισμικού και την αποθήκευση του σε sd κάρτα έγινε το πρώτο boot.



Εικόνα 4.1: Raspberry Pi Desktop

Για να ενεργοποιήσουμε την σειριακή κονσόλα μέσω του **terminal** χρησιμοποιούμε την εντολή **sudo raspi-config** και ακολουθούμε τα 4 παρακάτω βήματα:

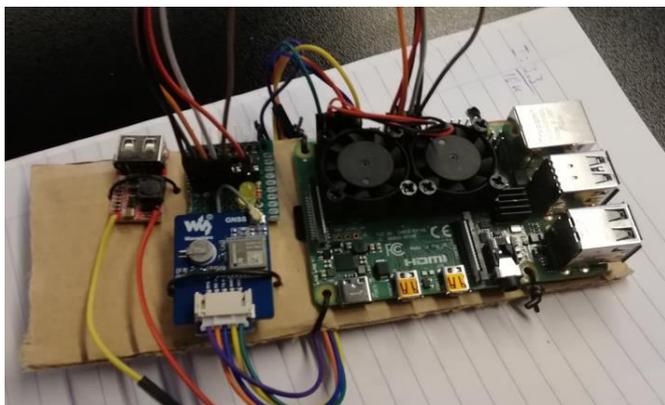


Εικόνα 4.2: Raspberry Pi raspi-config

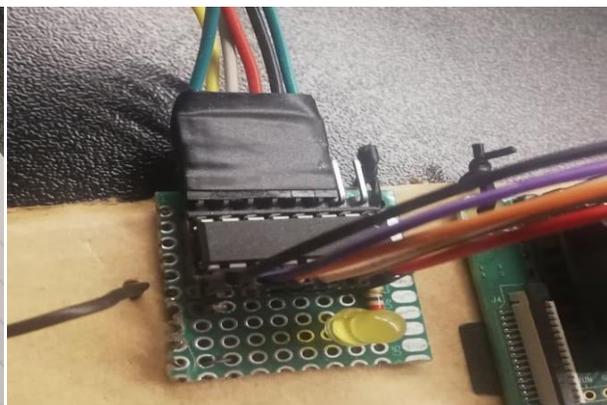
Έχοντας ενεργοποιήσει την σειριακή κονσόλα κάνουμε τις πρώτες συνδέσεις για το **gps**, το **down stepper** και την δημιουργία ενός **AnalogToDigital Converter**. Παρακάτω ακολουθούν τα βήματα που ακολουθηθήκαν.

DownStepper είναι ένα board το οποίο μέσω ενός ψαλιδιστή ή ενός transistor μπορεί να κάνει πτώση τάσης από 12v σε 5v που είναι η τάση λειτουργίας του Pi 4.

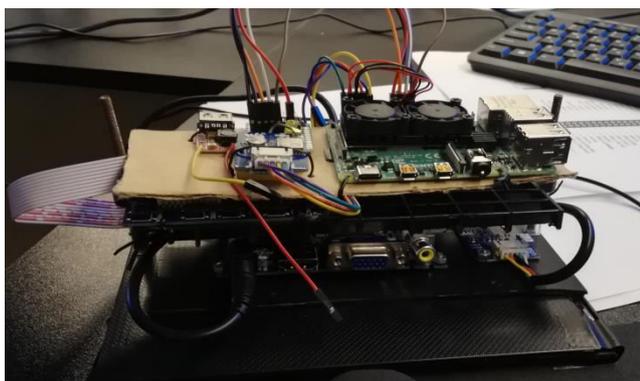
Τέλος έχουμε την οθόνη του συστήματος και όλο το κύκλωμα σε μια βάση.



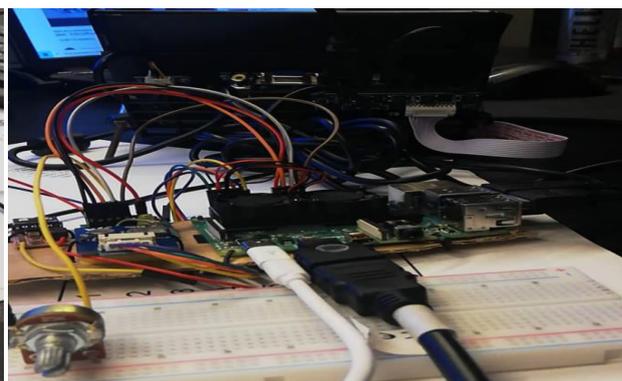
Εικόνα 4.2: Αρχική Διάταξη



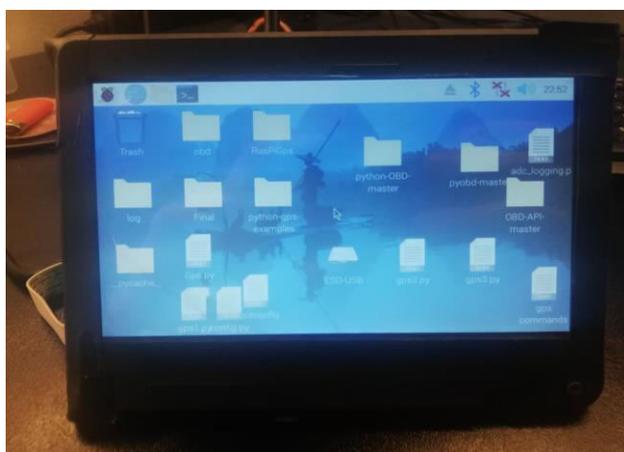
Εικόνα 4.3: AnalogToDigital Converter



Εικόνα 4.4: Τελικός σχεδιασμός



Εικόνα 4.5: Προσομοίωση αναλογικών σημάτων από After-Market όργανα μέσω Python



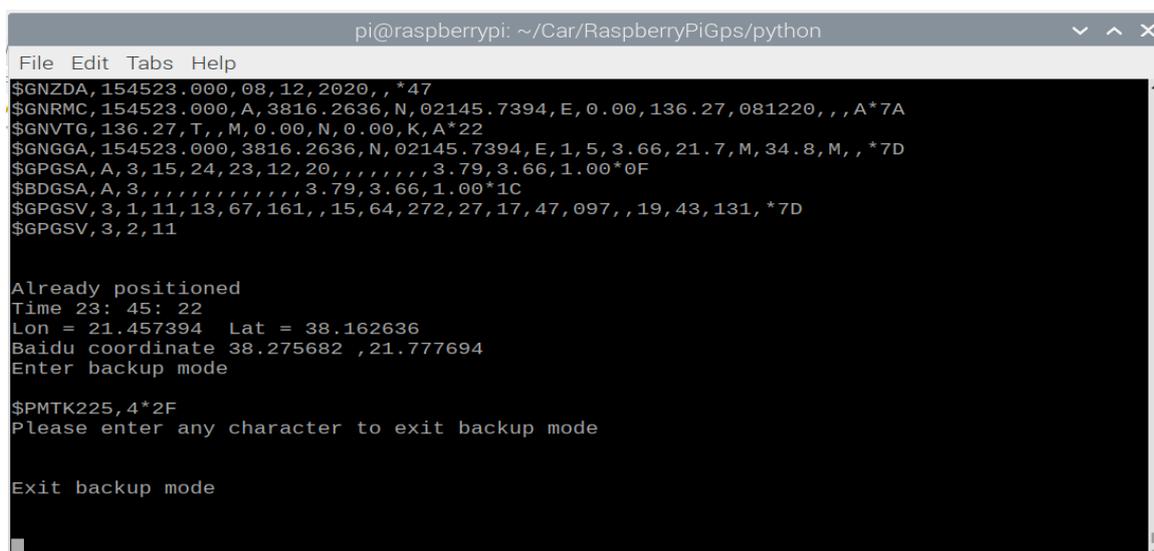
Εικόνα 4.5: Πρόσψη οθόνης

```

adc_onlist_done.py - /ho... *Python 3.7.3
*Python 3.7.3 Shell*
File Edit Shell Debug Options Window Help
===== RESTART: /home/pi/Desktop/Final/Adc_Files/adc_onlist_done.py =====
Reading MCP3008 values, press Ctrl-C to quit...
-----
| 011Temp | 011Press | ExtTemp | FuelPress | Boost | A/F R | 6 | 7 |
| 2.3 °C| 0.4 Bar| 7.4 °C| 6.5 Bar| 0.8 Bar| 15.0 o2| 1.08 | 0.76 |
-----
| 011Temp | 011Press | ExtTemp | FuelPress | Boost | A/F R | 6 | 7 |
| 18.2 °C| 0.7 Bar| 0.0 °C| 0.0 Bar| -1.0 Bar| 10.0 o2| 0.00 | 0.16 |
-----
| 011Temp | 011Press | ExtTemp | FuelPress | Boost | A/F R | 6 | 7 |
| 28.9 °C| 1.6 Bar| 0.8 °C| 0.0 Bar| -1.0 Bar| 10.0 o2| 0.00 | 0.00 |
-----
| 011Temp | 011Press | ExtTemp | FuelPress | Boost | A/F R | 6 | 7 |
| 19.9 °C| 0.6 Bar| 0.0 °C| 0.0 Bar| -1.0 Bar| 10.0 o2| 0.00 | 0.10 |
-----
| 011Temp | 011Press | ExtTemp | FuelPress | Boost | A/F R | 6 | 7 |
| 29.7 °C| 1.7 Bar| 4.2 °C| 1.5 Bar| -1.0 Bar| 10.0 o2| 0.00 | 0.00 |
-----

```

Εικόνα 4.6: PythonShell με δεδομένα από ADC



```
pi@raspberrypi: ~/Car/RaspberryPIGps/python
File Edit Tabs Help
$GNZDA,154523.000,08,12,2020,,*47
$GNRMC,154523.000,A,3816.2636,N,02145.7394,E,0.00,136.27,081220,,A*7A
$GNVTG,136.27,T,M,0.00,N,0.00,K,A*22
$GNGGA,154523.000,3816.2636,N,02145.7394,E,1,5,3.66,21.7,M,34.8,M,,*7D
$GPGSA,A,3,15,24,23,12,20,,,,,3.79,3.66,1.00*0F
$BDGSA,A,3,,,,,3.79,3.66,1.00*1C
$GPGSV,3,1,11,13,67,161,,15,64,272,27,17,47,097,,19,43,131,*7D
$GPGSV,3,2,11

Already positioned
Time 23: 45: 22
Lon = 21.457394 Lat = 38.162636
Baidu coordinate 38.275682 ,21.777694
Enter backup mode

$PMTK225,4*2F
Please enter any character to exit backup mode

Exit backup mode
```

Εικόνα 4.7: “NMEA” δεδομένα μέσω gps

Η διαχείριση του Pi 4 μέσω της Python ήταν αρκετά δύσκολη. Μετά από διαρκή αποσφαλμάτωση κώδικα με στόχο την λήψη δεδομένων OBD-II μέσω της ECU του αυτοκίνητου και συνεχή αναζήτηση πηγών δεν έφτασα σε κάποιο περαιτέρω αποτέλεσμα ούτε σε κάποια βιβλιοθήκη που να μπορώ να επεξεργαστώ σύμφωνα με τις ανάγκες της εργασίας.

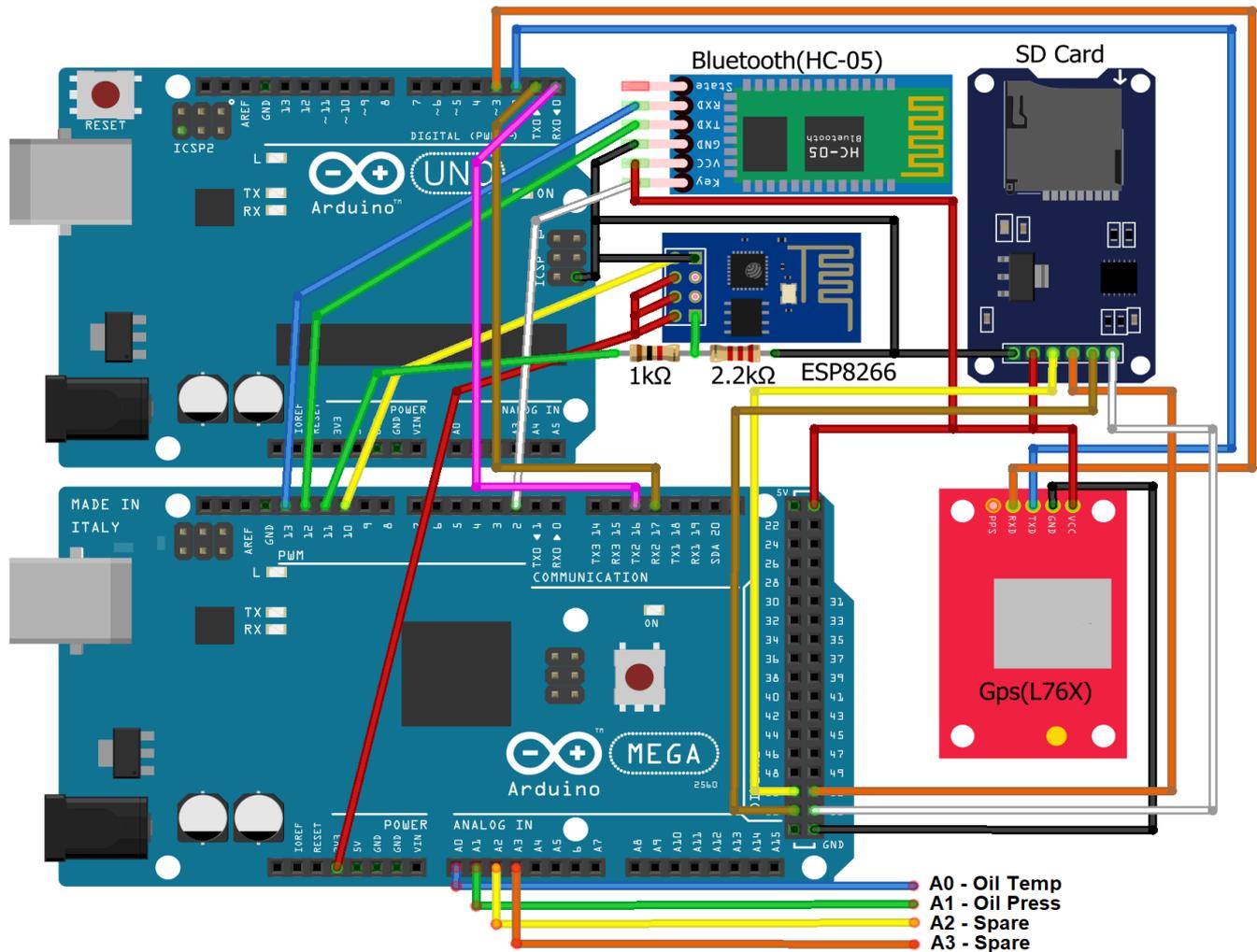
Έτσι την λύση στο πρόβλημα ήρθε να δώσει το Arduino με τα modules που έχω αναφέρει στο [Κεφάλαιο \(3.2\)](#).

Σε πρώτη φάση δούλεψα πάνω στο **UNO** όμως οι βιβλιοθήκες και οι δυναμικές μεταβλητές που χρησιμοποιήθηκαν εξαντλήσαν τους πόρους του board και ειδικότερα την μνήμη **SRam**. Οπότε έπειτα από διεξοδικότερη μελέτη έγινε αναδιάταξη του Project σε μικρότερα και σε ξεχωριστές διεργασίες. Επιλέχτηκε το Arduino **MEGA** διότι σύμφωνα με τα χαρακτηριστικά του τα όποια έχουν αναλυθεί στο [Κεφάλαιο \(3.1.2\)](#) η **SRam** που διαθέτει είναι 4 φορές μεγαλύτερη από του UNO.

Η διαχείριση πλέον γίνεται συνδυάστηκα και από το UNO και από το MEGA το οποίο είναι και η κυρία μονάδα επεξεργασίας δεδομένων και υπεύθυνο για όλα τα modules και τις επιμέρους διεργασίες. Αυτό έχει σαν αποτέλεσμα, έχοντας υπόψη και την οποιαδήποτε μελλοντική επέκταση του project, οι βιβλιοθήκες και οι δυναμικές μεταβλητές να μοιράζονται και να μην εξαντλούνται οι πόροι των Boards.

4.2 Σχεδίαση του (Car Real Time IoT Data Logger) Based On Arduino

Η νέα και τελική διάταξη του IoT συστήματος είναι η εξής:



Εικόνα 4.8: Τελικό Σύστημα & Συνδέσεις

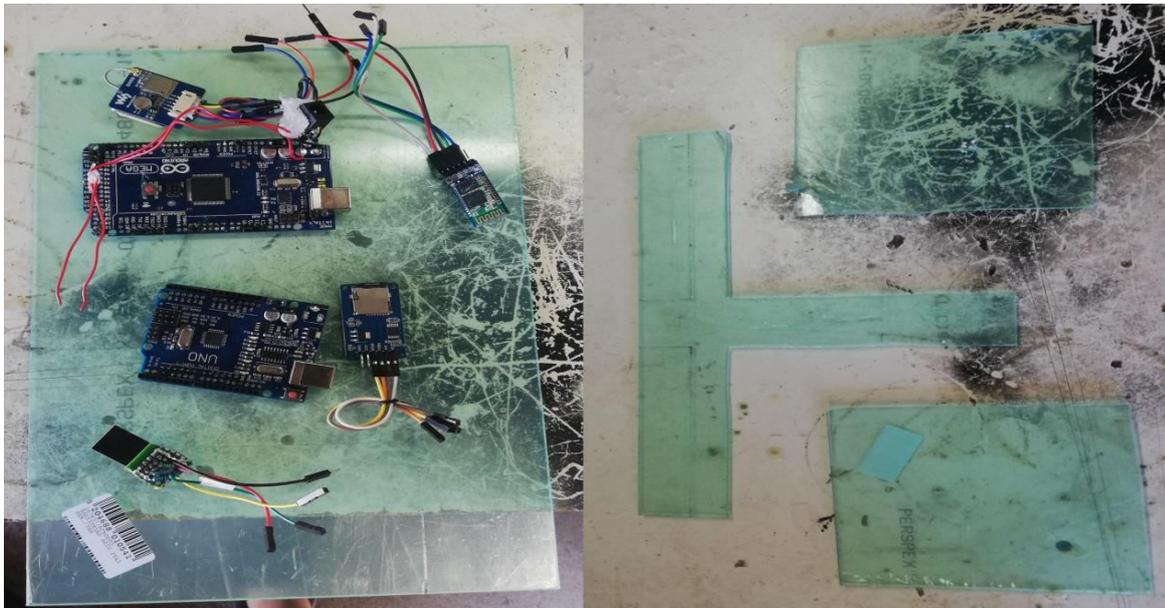
Έχοντας πλέον καταλήξει στην διάταξη αυτή, έγιναν τα απαραίτητα τεστ μέσω debugging codes που η συγγραφή τους και η τροποποίηση τους για το συγκεκριμένο project έχουν σκοπό την απροβλημάτιστη και ορθή λειτουργία του κάθε module ξεχωριστά.

Πινάκας διεργασιών του κάθε Arduino

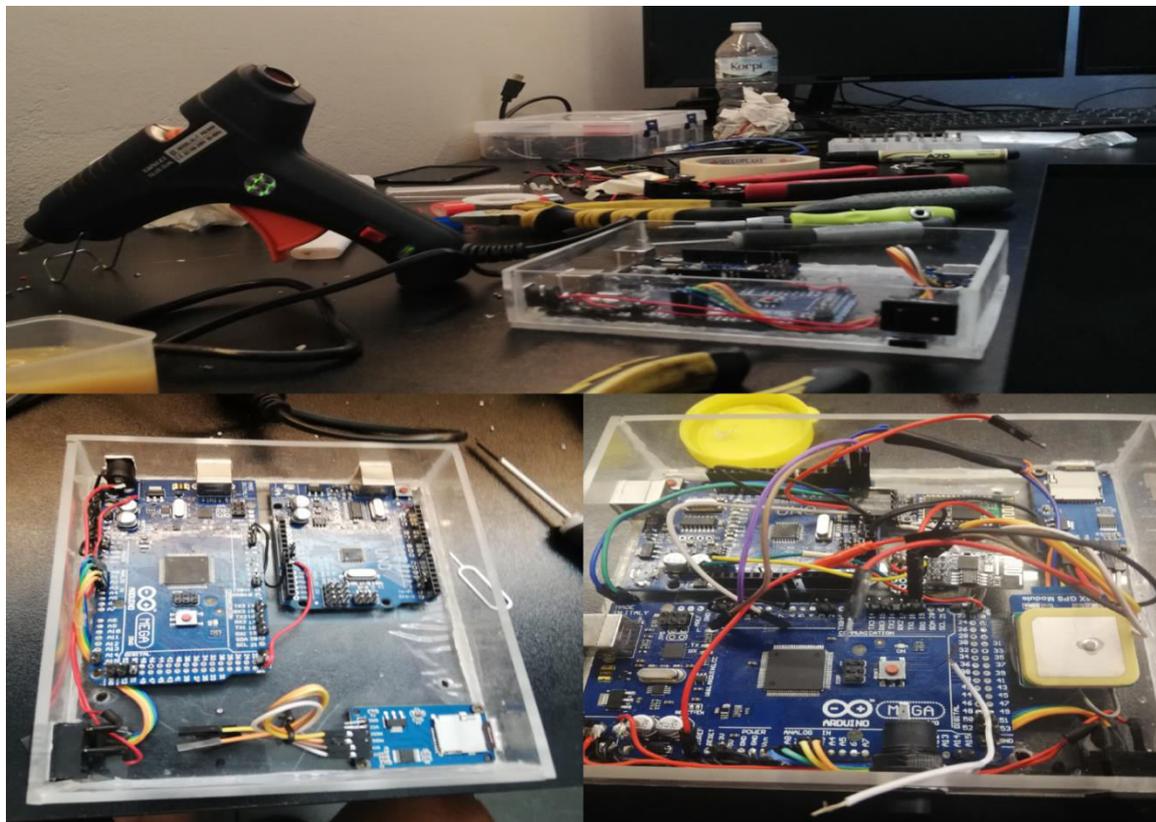
<u>ARDUINO UNO</u>	<u>ARDUINO MEGA</u>
Gps(L76X)	Αποδοχή δεδομένων gps από το Arduino Uno
	Λήψη και ανάλυση δεδομένων από αναλογικούς αισθητήρες
Διαχείριση απαραίτητων βιβλιοθηκών για την καταγραφή "NMEA" δεδομένων	Bluetooth (HC-05) - Συλλογή OBD-II δεδομένων μέσω της ECU
	WiFi (Esp-8266) - Upload των ζητούμενων δεδομένων στο ThinkSpeak
Μεταφορά δεδομένων στην κύρια μονάδα επεξεργασίας δεδομένων Arduino Mega	SD Card - Αποθήκευση των ζητούμενων δεδομένων και λειτουργία σαν Backup για περιπτώσεις έλλειψης δικτύου

4.2.1 Σχεδίαση και Υλοποίηση Βάσης και Σειριακής Επικοινωνίας

Η βάση της κατασκευής έγινε από Plexiglass το οποίο κόπηκε σε κομμάτια του απαραίτητου μήκους και πλάτους ώστε να χωρέσουν όλα τα boards σε συνδυασμό με το ότι το μέγεθος πρέπει να είναι όσο πιο compact γίνεται για να μπορεί να αποθηκευτεί σε κάποιο μέρος της καμπίνας του αυτοκινήτου. Το Plexiglass επιδέχθηκε έναντι κάποιου άλλου υλικού για να μπορεί να γίνει διάγνωση των λειτουργιών του κάθε board μέσω των led τους.

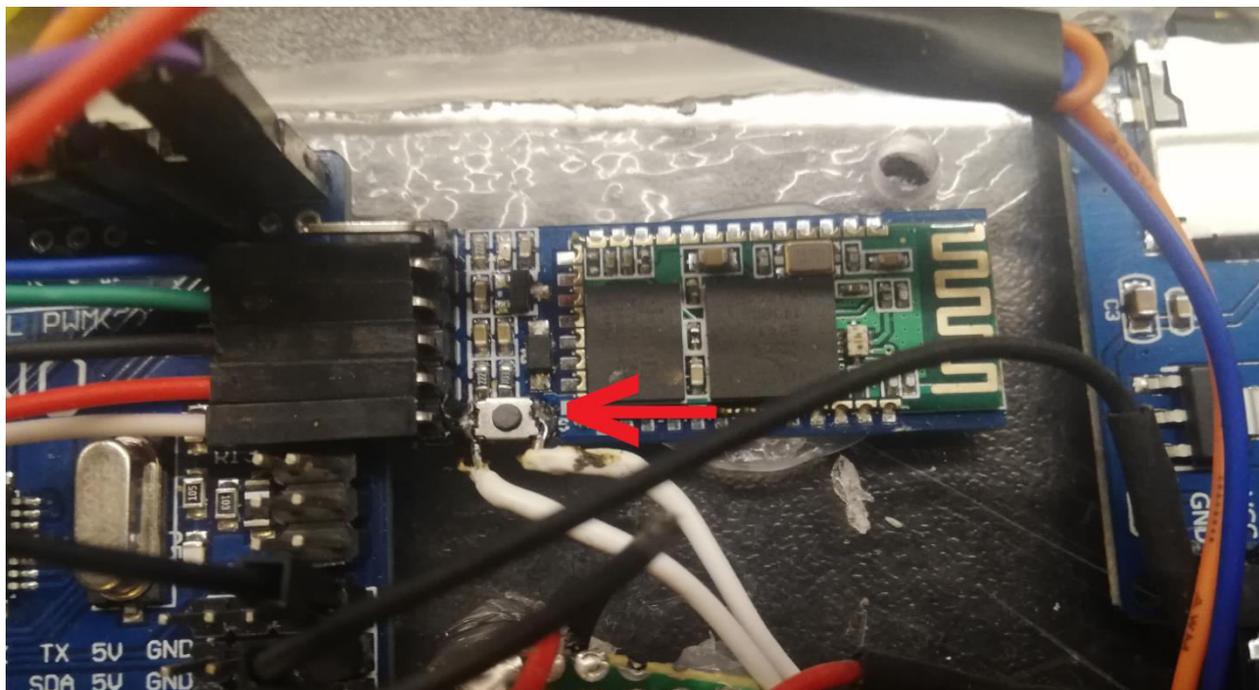


Εικόνα 4.9: Κατασκευή Βάσης Με Plexiglass



Εικόνα 4.10: Σύνδεση Των Arduino Με Τα Υπόλοιπα Στοιχεία

Αφού δημιουργήθηκαν οι κατάλληλες υποδοχές για τροφοδοσία και σειριακή επικοινωνία, προσδεθήκαν 2 ακόμα διακόπτες και 4 analog inputs. Ο 1^{ος} είναι On/Off για να μπορεί το σύστημα να δουλεύει συμφωνά με την θέληση μας και όχι κάθε φορά που ενεργοποιείται το κύκλωμα του αυτοκίνητου και ο 2^{ος} (push-button) είναι υπεύθυνος για το pairing του Bluetooth. Κάθε φορά που θέλουμε να κάνουμε pair με κάποια άλλη συσκευή πιέζουμε τον διακόπτη και μετά βάζουμε το σύστημα σε τροφοδοσία. Κάτω από τον διακόπτη τροφοδοσίας υπάρχουν τα 4 analog inputs για τα δεδομένα από τα After-Market όργανα.

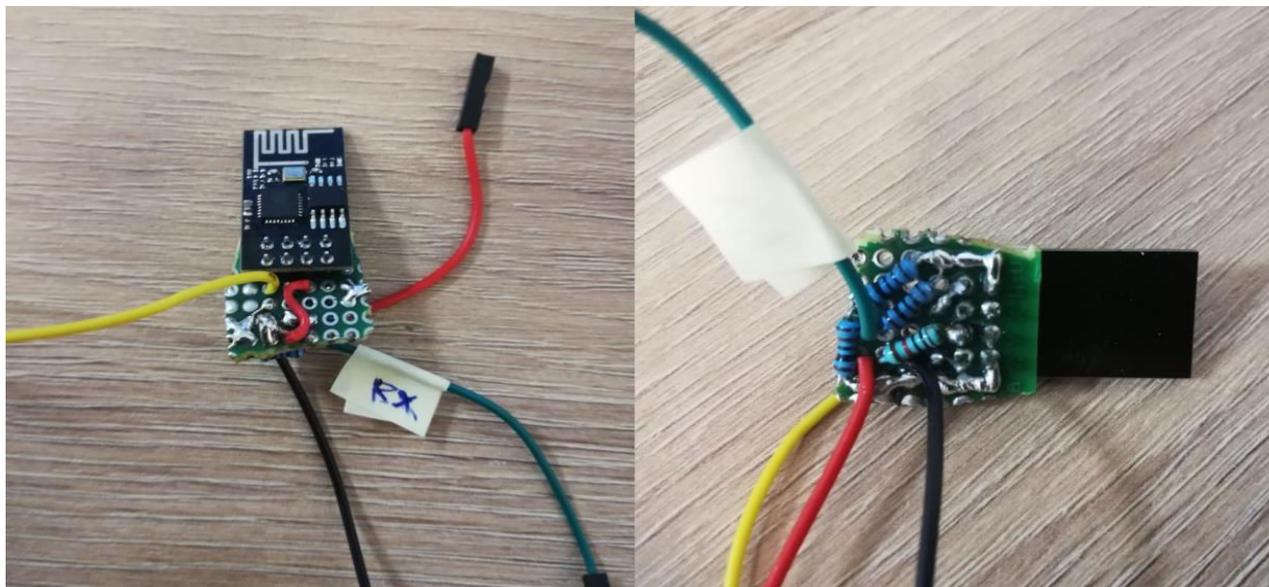


Εικόνα 4.11: Παράκαμψη Εργοστασιακού Pairing-Διακόπτη Του HC-05



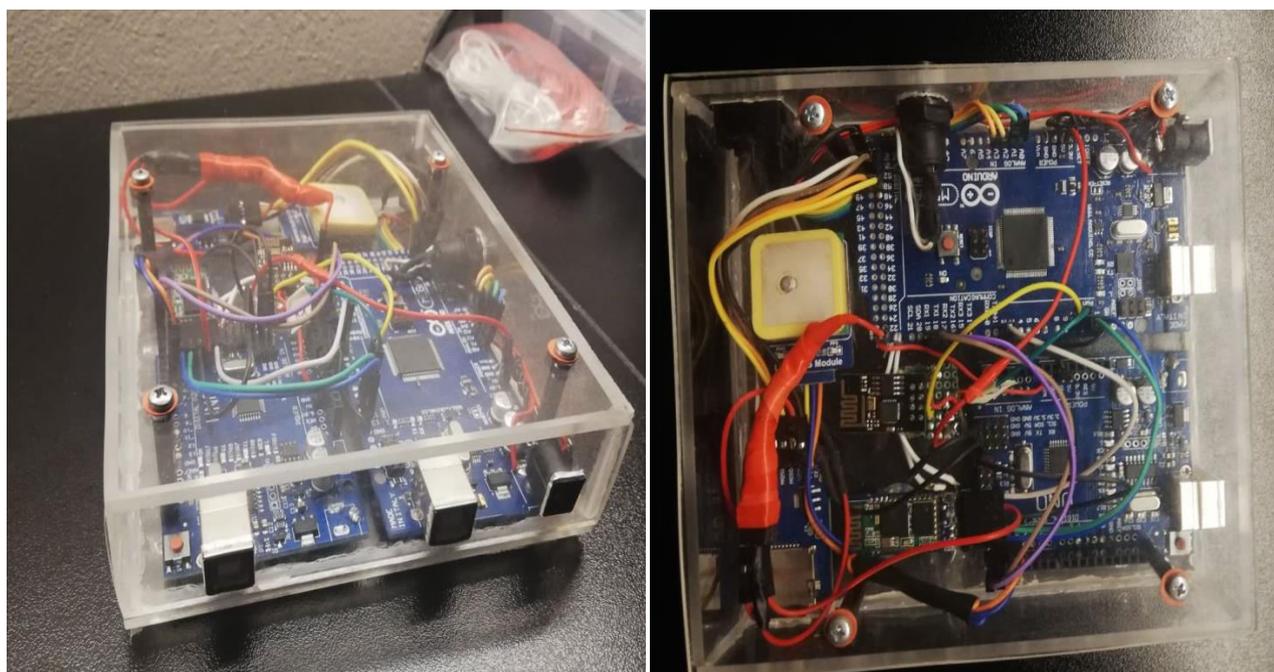
Εικόνα 4.12: Pairing, On/Off Διακόπτες & 4 Analog Inputs

Άλλη μια παρατήρηση που χρειάζεται να γίνει είναι ότι το **Wi-Fi (Esp-8266)** χρειάζεται μια κυκλωματική διάταξη για να ολοκληρωθεί η ορθή λειτουργία του, η οποία έχει αναφερθεί στο **Κεφάλαιο (3.2.3)**. Η υλοποίηση της γίνεται σε prototype board με μια σειρά αντιστάσεων διότι το Vref της συσκευής είναι 3.3V.



Εικόνα 4.13: Esp-8266 Δημιουργία Voltage Divider Για Το RX Pin

Έχοντας πλέον ολοκληρώσει την βάση που θα φιλοξενεί τα **boards** και τα **modules** και όλες τις συνδέσεις μεταξύ αυτών, το τελικό αποτέλεσμα στο οποίο φαίνονται οι σειριακές θύρες, η θύρα τροφοδοσίας και όλη η συνδεσμολογία είναι το παρακάτω:



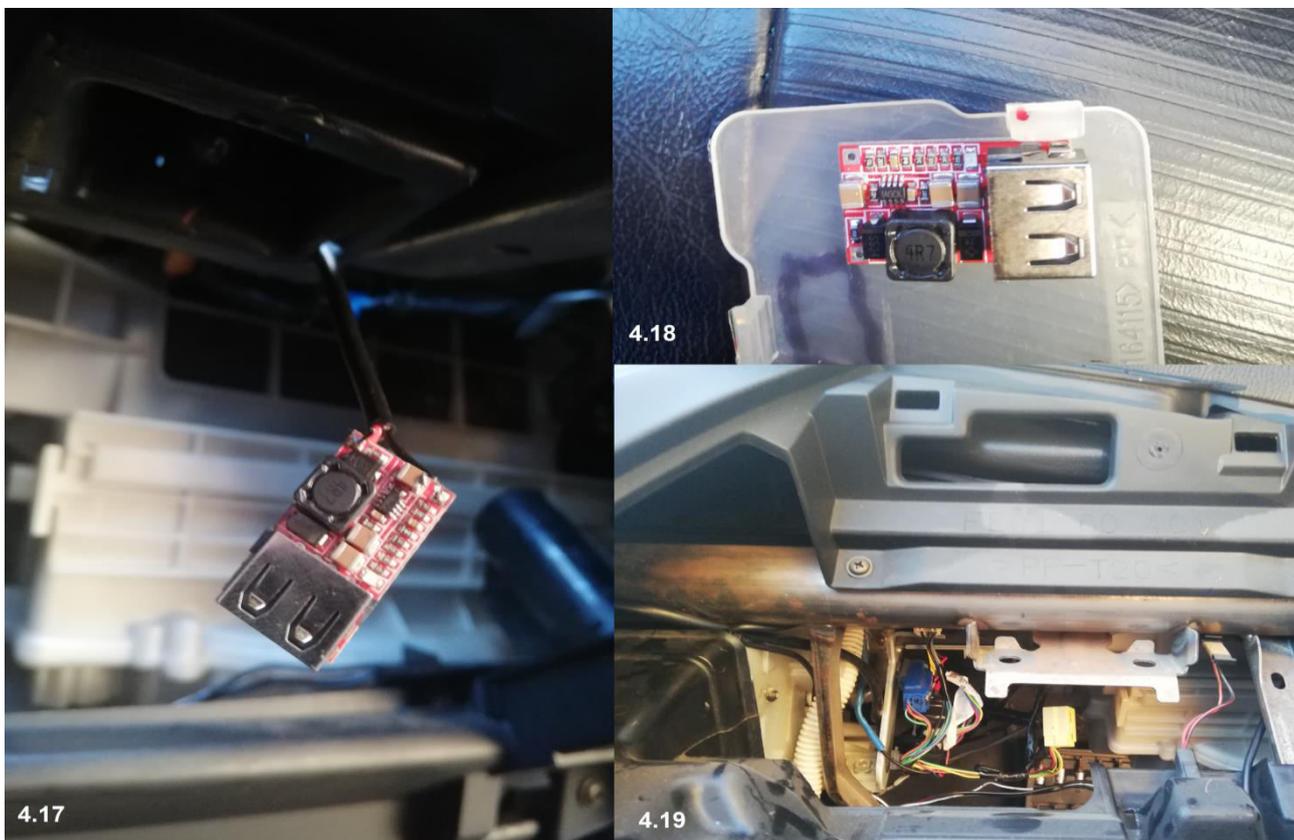
Εικόνα 4.14 & 4.15: Τελική Διάταξη Ολοκληρωμένου Project

4.3 Σχεδίαση και Υλοποίηση Συνδέσεων Συστήματος & Αισθητήρων

Αρχικά πρέπει να εξασφαλίσουμε την τροφοδοσία του συστήματος μας. Χρησιμοποιώντας το **DownStepper** ρίχνουμε την τάση του αυτοκίνητου από 12 σε 5 Volt που είναι η τάση λειτουργίας του συστήματος μας. Η θέση του ενσωματωμένου συστήματος θα είναι στο ντουλαπάκι του συνοδηγού και για αυτό το λόγο τα σήματα των αισθητήρων πρέπει να φτάσουν ως εκεί καθώς και η τροφοδοσία. Η διαδικασία που ακολουθήθηκε περιγράφεται παρακάτω.



Εικόνα 4.16 & 4.17: Μάτση Καλωδίου Τροφοδοσίας & Αισθητήρα Για Data Input



Εικόνα 4.18, 4.19 & 4.20: Εγκατάσταση DownStepper

Αφαίρεσα αρχικά τα απαραίτητα πλαστικά πάνελ και τον αερόσακο του συνοδηγού για γίνουν οι συνδέσεις και έτσι το θηλυκό USB τροφοδοσίας βρίσκεται πάνω από το ντουλαπάκι του συνοδηγού καθώς και τα καλώδια των αναλογικών σημάτων. Η μάτιση έγινε στο καλωδίου του αναπτήρα το οποίο ασφαρίζεται με 10^A ασφάλεια που είναι υπέρ αρκετά για την κατανάλωση του ενσωματωμένου συστήματος.

Τέλος τοποθετούμε τον **ELM327 Adapter** στην OBD θύρα που βρίσκεται συνήθως αριστερά κάτω από το τιμόνι για να ολοκληρωθεί η εγκατάσταση και η επικοινωνία του συστήματος.



Εικόνα 4.21 & 4.22: Θύρα OBD2 & ELM 327 Adapter

Η τελική εικόνα στο εσωτερικό του οχήματος είναι η παρακάτω, με το ενσωματωμένο σύστημα να βρίσκεται στο ντουλαπάκι και τα αναλογικά σήματα των αισθητήρων στα Inputs.



Εικόνα 4.23 & 4.24: Τελικό Αποτέλεσμα

4.4 Ανάπτυξη του Software

Για να λειτουργήσουν όλα τα παραπάνω θα πρέπει να προγραμματίσουμε το Arduino να εκτελεί κάποιες συγκεκριμένες διεργασίες. Αρχικά για να προγραμματίσουμε το Arduino θα πρέπει να κατεβάσουμε και να εγκαταστήσουμε στον υπολογιστή μας το **Arduino Software (IDE)**.

Η έκδοση που χρησιμοποιείται είναι **Arduino1.8.15** σε αυτό πρέπει να δώσουμε προσοχή διότι σε επόμενες εκδόσεις ο κώδικας μπορεί να χρειάζεται τροποποιήσεις για να είναι λειτουργικός.

Ξεκινάμε με τις βιβλιοθήκες που χρειάζονται για τα διάφορα **modules** που έχουμε συνδεδεμένα στο Arduino.

- SoftwareSerial.h
- SPI.h
- SD.h
- MapFloat.h
- ELMduino.h
- L76X.h

Οι 3 πρώτες βρίσκονται προεγκατεστημένες στο Arduino IDE και αφορούν την σειριακή επικοινωνία, την αποθήκευση και αναζήτηση δεδομένων. Η MapFloat και η ELMduino βρίσκονται μαζί με παραδείγματα στο <https://github.com/>. Η MapFloat χρησιμοποιείται για να οριστεί εκ νέου το φάσμα των τιμών των αναλογικών εισόδων στο σύστημα ενώ η ELMduino βρίσκει εφαρμογή στα **OBD-II** δεδομένα που αντλούνται από τον εγκέφαλο του αυτοκινήτου. Τέλος έχουμε την L76X την οποία βρίσκουμε στο https://www.waveshare.com/wiki/L76X_GPS_Module και έχει στόχο την συλλογή των **NMEA δεδομένων** του **Gps**.

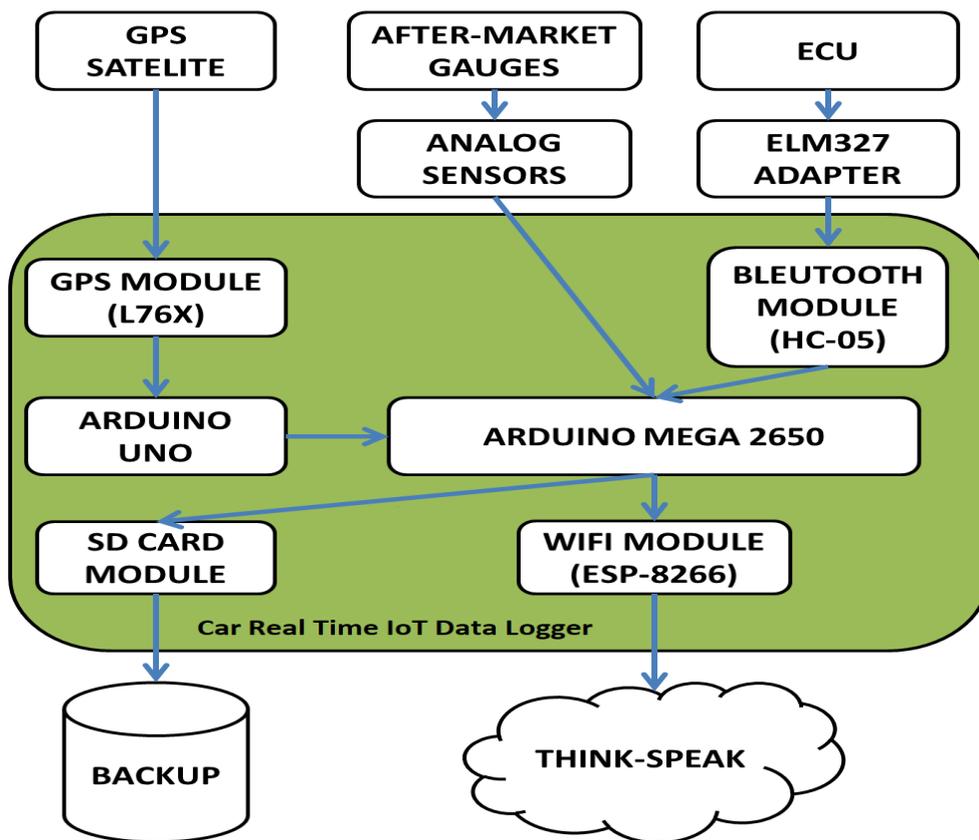
Έπειτα ορίζουμε την λειτουργία του κάθε **Pin**, δηλώνουμε και αρχικοποιούμε κάποιες σταθερές μεταβλητές όπου είναι συνδεδεμένες στα Digital και στα Analog Pin του Arduino και ξέρουμε ότι δεν πρόκειται χρησιμοποιηθούν για κάτι άλλο. Στην συνέχεια ορίζουμε το όνομα και κωδικό του δικτύου καθώς και το μοναδικό **API key** του καναλιού μας. Ο καθορισμός των πεδίων και το σε ποιες τιμές αντιστοιχούν έρχεται να ολοκληρώσει την επικοινωνία με το σύννεφο.

Στην **setup()** καλούμε τις συναρτήσεις που χρειάζονται για την συνέχεια, ορίσουμε το **baud rate** και σετάρουμε τις αρχικές ρυθμίσεις των σειριακών θυρών που είναι συνδεδεμένα τα **modules**.

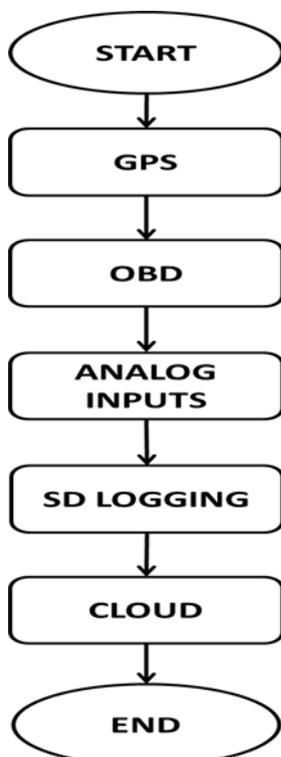
Μέσα στην **loop()** γράφουμε ότι εκτελείται επανειλημμένα όπως οι μετρήσεις των αισθητήρων για τα A0, A1, A2 & A3 και η λήψη των **OBD & GPS** δεδομένων. Αυτά θα πρέπει να σταλούν στο **ThinkSpeak** όπου εκεί θα γίνει και η γραφική τους απεικόνιση.

Στο τελικό σκέλος του κώδικα, εκτός της **loop()**, υπάρχουν οι επιμέρους συναρτήσεις για την ενσωμάτωση τις κάθε λειτουργίας των modules στο **main** πρόγραμμα.

Και τέλος έχουμε το **Flowchart** και το **Component Diagram** για το πως λειτουργεί συνολικά το σύστημα με την αλληλεπίδραση όλων των στοιχείων και τελικά ο κώδικας.



Εικόνα 4.25: Component Diagram



Εικόνα 4.26: FlowChart Diagram

Κεφάλαιο 5^ο - Συμπεράσματα & Μελλοντικές Προεκτάσεις

5.1 Περίληψη

Στόχος της παρούσας πτυχιακής εργασίας ήταν μια εισαγωγή στην μονάδα ECU των αυτοκινήτων, στα ενσωματωμένα συστήματα και στις λειτουργίες των μικροελεγκτών ανάλογα με την εκάστοτε εφαρμογή, καθώς επίσης και η σχεδίαση και ανάπτυξη IoT ενσωματωμένου συστήματος real-time παρακολούθησης και καταγραφής αισθητήρων και OBD-II δεδομένων.

Πιο συγκεκριμένα έγινε μια εισαγωγή στον τρόπο λειτουργίας της ECU και την εξέλιξη της μέχρι σήμερα, ενώ από την πλευρά των μικροελεγκτών έγινε μια πιο εκτεταμένη ανάλυση στις λειτουργίες και τα χαρακτηριστικά τους.

Ακολούθησε η παρουσίαση των microcontrollers **Arduino Uno, Mega & Raspberry Pi 4**, πάνω στους οποίους μελέτησα εντατικά. Ανέλυσα του **pinout** τους και τα χαρακτηριστικά των **modules** που χρησιμοποιήθηκαν και μέσω όλων αυτών αναπτύχθηκαν και υλοποιήθηκαν εφαρμογές με στόχο την ολοκλήρωση του **Car Real Time IoT Data Logger**.

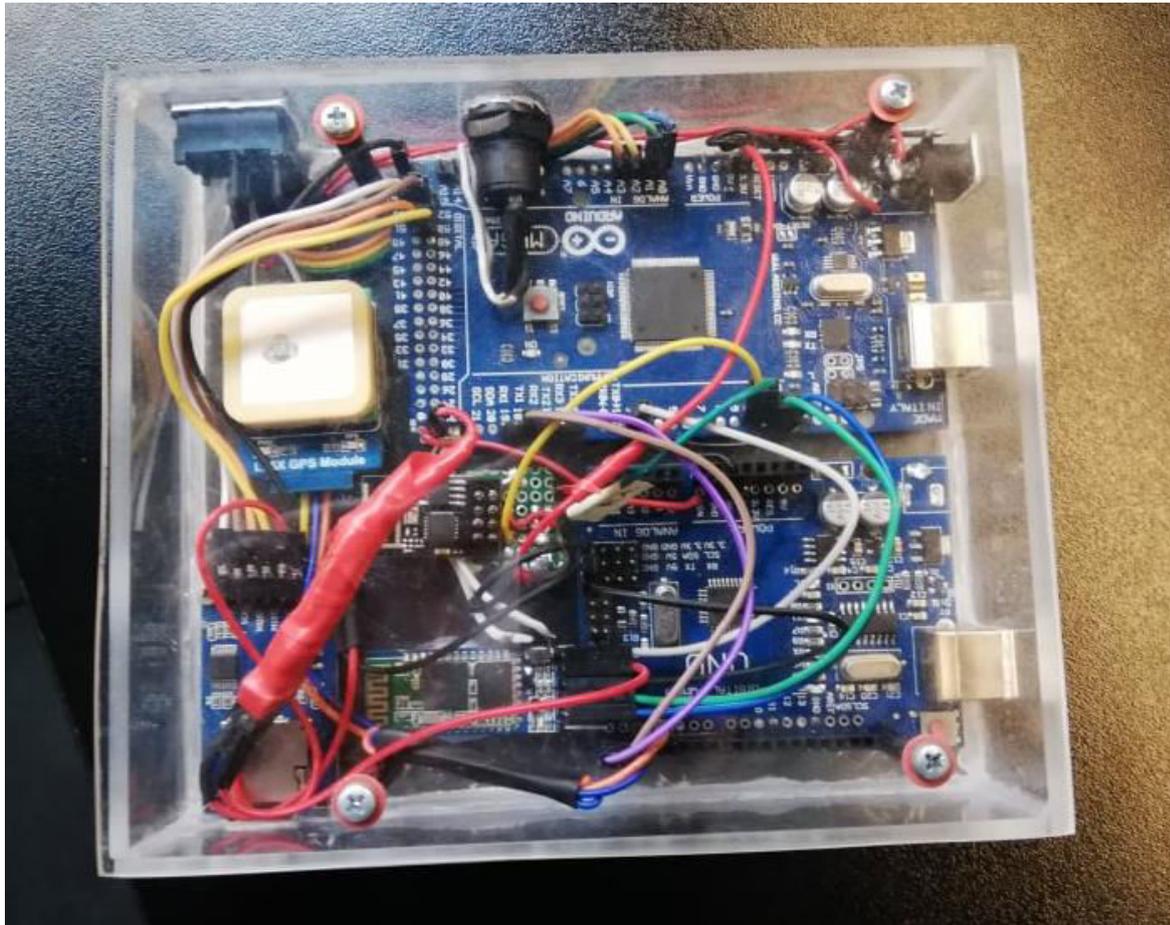
Έπειτα αναφέρθηκα εκτενεστέρα στην λογική των ECUs και στο με ποιο τρόπο μπορούμε να αντλήσουμε δεδομένα που αφορούν την λειτουργία, την συμπεριφορά και την συντήρηση του οχήματος μας. Ειδικότερα έγινε ανάλυση του **ELM327** ενός ακόμα μικροελεγκτή που είχε καταλυτική σημασία στην επικοινωνία του συστήματος με την ECU. Ύστερα ανέλυσα την λογική της θύρας **OBD** και των εντολών **PID** για την μεταξύ τους επικοινωνία.

Ένας ακόμα παράγοντας του λαμβάνει υπόψη το τελικό σύστημα είναι τα **After-Market όργανα** που κατά προτίμηση του οδηγού τοποθετούνται στο όχημα. Πλέον μπορούμε να καταγράψουμε τις μετρήσεις και αυτών για μια πιο ολοκληρωμένη εικόνα της λειτουργίας του οχήματος.

Τέλος έγινε ανάλυση του **IoT** (Internet of Things) και πως μέσω του **ThinkSpeak** τα δεδομένα θα απεικονίζονται στο σύννεφο. Οι δυνατότητες που παρέχονται από αυτή την πλατφόρμα, είναι γραφικές παραστάσεις και προσομοίωση των μετρητών του οχήματος.

Όσον αφορά στο κατασκευαστικό κομμάτι έγινε αναλυτική παρουσίαση βήμα προς βήμα του πλαισίου που φιλοξενεί το **Car Real Time IoT Data Logger**, των συνδέσεων μεταξύ των **boards** και των **modules** καθώς επίσης και της εγκατάστασης του στο όχημα, συμπεριλαμβάνοντας τις απαραίτητες τροποποιήσεις και τις αλλαγές σε κάποια από τα κυκλώματα του οχήματος.

Το συγκεκριμένο Project έχει υλοποιηθεί με την δυνατότητα επεκτασιμότητας και προσθήκης ακόμα περισσότερων λειτουργιών, διότι το αυτοκίνητο είναι στις πρώτες θέσεις ενδιαφέροντος για ανάπτυξη IoT εφαρμογών.



Εικόνα 5.1: Car Real Time IoT Data Logger

Παρακάτω βλέπουμε μερικά από τα γραφήματα στο **ThinkSpeak** και μια προσομοίωση των μετρητών του οχήματος.



Εικόνα 5.2: ThinkSpeak Διαγράμματα & Μετρητές

5.2 Προβλήματα - Παρατηρήσεις

Ολοκληρώνοντας την εργασία υπήρξαν δεν υπήρξαν ιδιαίτερα προβλήματα. Με αρκετή μελέτη και αναζήτηση της εκάστοτε βιβλιογραφίας έφτασα στο επιθυμητό αποτέλεσμα. Βεβαία κάποιες από τις παρατηρήσεις που σημειώθηκαν κατά την υλοποίηση βρίσκονται παρακάτω:

- Μια παρατήρηση είναι ότι τις μέρες που έχει συννεφιά το **Gps** αργεί να βρει την τοποθεσία και να μας δώσει δεδομένα, καθώς και για το λόγο ότι το σύστημα μας βρίσκεται στο ντουλαπάκι του συνοδηγού έχουμε μια ακόμα δυσκολία στην λήψη των **Gps** σημάτων. Η πιο ορθή εγκατάσταση γίνεται με την κεραία του Gps να βρίσκεται τελικά στο ταμπλό του οχήματος και σε συνδυασμό με ηλιοφάνεια, έχουμε ιδανικές συνθήκες λειτουργίας.
- Επίσης η τροφοδοσία του συστήματος για το λόγο ότι τροφοδοτούμε το **Uno** μέσω του **Mega** έχουμε μια πτώση τάσης της τάξης του 0,5V τουλάχιστον μέχρι να σετάρουμε τις αρχικές παραμέτρους. Αυτό αντιμετωπίστηκε τροφοδοτώντας ξεχωριστά και το Uno και το Mega.
- Το **WiFi module** που χρησιμοποιώ από την άλλη, λειτουργεί κάνοντας Hotspot από τα δεδομένα του κινητού. Όποτε είναι προϋπόθεση για την λειτουργία του συστήματος ένας πάροχος δικτύου που στην περίπτωση μας είναι το κινητό.
- Πρέπει επίσης να λάβουμε υπόψη το γεγονός ότι στην κατασκευή το υλικό που χρησιμοποιήθηκε είναι **Plexiglass** και όχι τυχαία. Είναι διάφανο και δίνει την δυνατότητα αναγνώρισης των λειτουργιών των **Boards** και των **Modules** μέσω των **Debugging Led** που διαθέτουν. Αρχικά είχα σκεφτεί ένα μαύρο ηλεκτρολογικό κουτί, αλλά δεν ήταν καθόλου προσιτή η **Debugging** διαδικασία.
- Όσο αναφορά στον κώδικα ένα μέρος του που χρειάστηκε παραπάνω μελέτη ήταν η επικοινωνία μεταξύ των 2 Arduino για την σειριακή αποθήκευση των **NMEA** δεδομένων μέσω του GPS. Υπάρχον αρκετές σειριακές θύρες που ενεργοποιούνται στην τελική φάση της εργασίας και ο συγχρονισμός και η διαχείριση τους ήταν αρκετά απαιτητική, αν λάβουμε υπόψη ότι δουλεύουν και σε διαφορετικά **Baud rate**.
- Τέλος για την εγκατάσταση του συστήματος στο όχημα χρειάστηκε να γίνουν κάποιες τροποποιήσεις τόσο σε κυκλωματικό επίπεδο όσο και σε επίπεδο διάταξης. Για παράδειγμα στο συγκεκριμένο μοντέλο δεν υπάρχει παροχή 5V μέσω USB, για αυτό το λόγο χρησιμοποίησα το Down Stepper. Χρειάστηκε να επέμβω και στην διάταξη στο πλαίσιο του συνοδηγού για να βρει μια θέση η νέα παροχή ρεύματος.

Συνοψίζοντας όλες οι παρατηρήσεις και τα προβλήματα με την κατάλληλη μελέτη και με διαφορές δόκιμες βρέθηκε τελικά ο τρόπος είτε να λυθούν είτε να παρακαμφθούν.

5.3 Μελλοντικές Προεκτάσεις

Μελλοντικές προεκτάσεις και εφαρμογές υπάρχουν και είναι αρκετές και ίσως κάποιες από αυτές υλοποιηθούν σε επόμενο επίπεδο. Το πλέον τελικό IoT σύστημα λαμβάνει δεδομένα και έπειτα από επεξεργασία τροφοδοτεί το Think Speak με αυτά.

- Μια επομένη υλοποίηση είναι να προσδεθεί οθόνη για να ενημερώνεται και ο οδηγός για την συνολική κατάσταση του οχήματος Real-time. Το Arduino είναι μια πλατφόρμα που υποστηρίζει την χρήση LCD οθόνης και εύκολα μπορεί να γίνει αυτή η επέκταση.
- Επίσης μπορεί μέσω Android εφαρμογής να γίνει σύνδεση με την βάση του ThinkSpeak και να εμπλακεί έτσι στην ενημέρωση του οδηγού. Επομένως μέσω κινητού κάνοντας χρήση της εφαρμογής **ThingView - ThingSpeak viewer** μπορεί ο οδηγός να έχει πλήρη εικόνα των δεδομένων που ανεβαίνουν στο Cloud.
- Μέσω ενός **Smart Bracelet** μπορεί να γίνει καταγραφή δεδομένων όπως, καρδιακοί παλμοί, αρτηριακή πίεση και εφίδρωση. Σε συνάρτηση με τα “g” της επιτάχυνσης που δέχεται το όχημα τα όποια μπορούμε να πάρουμε από το **Gps Module** γίνεται αναγωγή του συστήματος σε σύστημα προσομοίωσης οδηγού αγώνων τύπου Formula 1. Έτσι θα έχουμε καταγραφή της συμπεριφοράς του οχήματος και του οδηγού σε συγκεκριμένο κομμάτι διαδρομής όπως είναι μια πιστά αγώνων.
- Σύνδεση του τελικού συστήματος μπορεί να υπάρξει και με τον χώρο του tuning, απαιτεί βέβαια τεραστία γνώση μηχανολογίας αλλά σίγουρα σε πρώιμα στάδια μπορεί να γίνει βελτίωση σε κάποιες από τις εργοστασιακές ρυθμίσεις-λειτουργίες προς την κατεύθυνση που θέλουμε.

Αυτές ήταν κάποιες από τις μελλοντικές προεκτάσεις που είναι προσιτό να υλοποιηθούν. Όσο αναφορά γενικότερα στην αυτοκινητοβιομηχανία, τα IoT συστήματα ενσωματώνονται ολοένα και περισσότερο σε όλα τα οχήματα βρίσκοντας παρά πολλές εφαρμογές. Όπως είναι η αυτόνομη οδήγηση, Self-Healing συστήματα, ψυχαγωγία μέσω multimedia, ακόμα και την ενσωμάτωση αυτών στο οδικό δίκτυο για άριστη διαχείριση της κίνησης και την αποφυγή ατυχημάτων .

Εν κατακλείδι, οι επεκτάσεις είναι πραγματικά τόσο μεγάλες, όσο και η φαντασία μας και όσο υπάρχει όρεξη για γνώση και εργασία πάνω στην ανάπτυξη εφαρμογών, θα γίνονται ακόμα και περισσότερες.

5.4 Βιβλιογραφία

- [1] **Car ECUs**
(<https://www.caroto.gr/2009/03/03/ecu/>)
- [2] **Workings of Ecu**
(https://en.wikipedia.org/wiki/Engine_control_unit)
- [3] **Automotive MicroChips**
(<https://www.chipsetc.com/computer-chips-inside-the-car.html>)
- [4] **IoT4Car**
(https://create.arduino.cc/projecthub/frankzhao/iot4car-1b07f1?ref=similar&ref_id=18269&offset=0)
- [5] **Smart Car**
(https://create.arduino.cc/projecthub/pilgrimbill/carsmart-88164a?ref=tag&ref_id=vehicle&offset=12)
- [6] **Definition of Microcontrollers**
(<https://en.wikipedia.org/wiki/Microcontroller>)
- [7] **Applications of Microcontrollers**
(<https://www.eit.edu.au/resources/types-and-applications-of-microcontrollers/>)
- [8] **Arduino Introduction**
(<https://www.electronicshub.org/arduino-introduction/>)
- [9] **Arduino Uno Pinout**
(<https://www.electronicshub.org/arduino-uno-pinout/>)
- [10] **Arduino Mega Pinout**
(<https://www.electronicshub.org/arduino-mega-pinout/>)
- [11] **Gps Module**
(<https://www.waveshare.com/l76x-gps-module.htm>)
- [12] **Bluetooth Module**
(<https://components101.com/wireless/hc-05-bluetooth-module>)
- [13] **WiFi Module**
(<https://www.electronicwings.com/sensors-modules/esp8266-wifi-module>)

- [14] **WiFi & Arduino Interaction**
(<https://www.electronicshub.org/esp8266-arduino-interface/>)

- [15] **SD Card Module**
(<https://randomnerdtutorials.com/guide-to-sd-card-module-with-arduino/>)

- [16] **SD Card Module & Arduino interaction**
(<https://create.arduino.cc/projecthub/electropeak/sd-card-module-with-arduino-how-to-read-write-data-37f390>)

- [17] **Raspberry Pi 4**
(https://en.wikipedia.org/wiki/Raspberry_Pi)

- [18] **Raspberry Pi 4 Basics**
(<https://projects.raspberrypi.org/en/projects/raspberry-pi-getting-started>)

- [19] **Analog To Digital Converter**
(<https://learn.adafruit.com/raspberry-pi-analog-to-digital-converters/mcp3008>)

- [20] **Elm 327**
(<https://en.wikipedia.org/wiki/ELM327>)

- [21] **OBD-II & PIDs**
(https://en.wikipedia.org/wiki/On-board_diagnostics)

- [22] **OBD-II Commands**
(<https://learn.sparkfun.com/tutorials/obd-ii-uart-hookup-guide/obd-commands>)

- [23] **After-Market Gauges**
(http://www.bugsandbuggies.com/Defi/dg_install.html)