



## ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

### ΘΕΜΑ

**Σύγχρονες Συσκευές Εντοπισμού Θέσης με Αυξημένη Ευφύια  
και Χαμηλό Κόστος**

### ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

**ΚΑΣΑΠΛΕΡΗ ΙΑΣΟΝΑ ΣΤΥΛΙΑΝΟΥ**

**Αριθμός Μητρώου: 03112051**

**Επιβλέπων :** Κουκούτσης Ηλίας  
Επίκουρος Καθηγητής Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2018





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ  
ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

## Σύγχρονες Συσκευές Εντοπισμού Θέσης με Αυξημένη Ευφυΐα και Χαμηλό Κόστος

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

**ΚΑΣΑΠΛΕΡΗ ΙΑΣΟΝΑ ΣΤΥΛΙΑΝΟΥ**

**Αριθμός Μητρώου: 03112051**

**Επιβλέπων :** Κουκούτσης Ηλίας  
Επίκουρος Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 22η Φεβρουαρίου 2018.

.....  
Ηλίας Κουκούτσης  
Επικ. Καθηγητής Ε.Μ.Π.

.....  
Κωνσταντίνος Παπαοδυσσεύς  
Καθηγητής Ε.Μ.Π.

.....  
Γεώργιος Καμπουράκης  
Αναπλ. Καθηγητής Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2018

.....

ΚΑΣΑΠΛΕΡΗΣ ΙΑΣΩΝ - ΣΤΥΛΙΑΝΟΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

© 2018 – All rights reserved

Copyright © Κασαπλήρης Ιάσων-Στυλιανός, 2018

Copyright © Κουκούτσης Ηλίας, 2018

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

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τους συγγραφείς και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσοβίου Πολυτεχνείου.

## ΠΕΡΙΛΗΨΗ

Ο σκοπός της διπλωματικής εργασίας είναι η πρόταση μιας ευφυούς και οικονομικής συσκευής, που αποτελείται από ενσωματωμένα συστήματα και έχει την δυνατότητα να εντοπίζει την τοποθεσία στην οποία βρίσκεται ανά πάσα στιγμή και να στέλνει τα δεδομένα αυτά με μήνυμα σε κινητό τηλέφωνο και υπολογιστή. Η συσκευή αυτή έχει τη δυνατότητα τοπικής αυτοοργάνωσης και θα ενταχθεί σε πολυλειτουργικό σταθμό βάσης για εφαρμογές στις μεταφορές επιβατών και εμπορευμάτων.

Για την κατασκευή της συσκευής αυτής έγινε έρευνα ώστε να βρεθούν τα κατάλληλα ενσωματωμένα συστήματα που θα χρησιμοποιηθούν. Αρχικά αναζητήσαμε κατάλληλο μικροελεγκτή που θα είναι ανοιχτού κώδικα και όσο το δυνατόν πιο ευφυής και οικονομικός. Από την αναζήτηση αυτή καταλήξαμε ότι θα χρησιμοποιήσουμε το Arduino Uno, που είναι ένας μικροελεγκτής μονής πλακέτας, δηλαδή μια απλή μητρική πλακέτα ανοιχτού κώδικα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring. Στη συνέχεια έγινε έρευνα για να βρούμε κατάλληλο ενσωματωμένο σύστημα που θα συνδεθεί με το Arduino και θα δίνει την τοποθεσία με παγκόσμιο σύστημα στιγματοθέτησης (GPS). Έπειτα από ενδελεχή έρευνα επιλέξαμε την πλακέτα GPS της Adafruit, την οποία συνδέσαμε με το Arduino και στην συνέχεια προγραμματίσαμε κατάλληλα προκειμένου να μας στέλνει την τοποθεσία όποτε εμείς θέλουμε. Τέλος, έπρεπε τα δεδομένα που λαμβάνουμε από το GPS να στέλνονται με γραπτό μήνυμα σε κινητό τηλέφωνο και σε υπολογιστή. Για το λόγο αυτό αναζητήσαμε κατάλληλη πλακέτα GSM με δυνατότητα εισαγωγής κάρτας sim ώστε να τη συνδέσουμε με το υπόλοιπο σύστημα προκειμένου να στέλνουμε τα δεδομένα. Επιλέξαμε, λοιπόν, το Arduino GSM Shield 2, και με τον κατάλληλο προγραμματισμό επιτύχαμε την επικοινωνία του συστήματος μας με κινητό τηλέφωνο και υπολογιστή.

Το σύστημα που κατασκευάσαμε δύναται να τοποθετηθεί σε μέσα μεταφοράς όπως σε βαγόνια τραίνου και σε εμπορεύματα πλοίων, προκειμένου αυτά να παρακολουθούνται ανά πάσα στιγμή και να στέλνουν σήμα τοποθεσίας στον πολυλειτουργικό σταθμό με αποτέλεσμα τον ορθό συντονισμό κατά την μεταφορά και την ελαχιστοποίηση των κλοπών. Η κατασκευή μας είναι οικονομική, αυτόνομη και απαιτεί πολύ μικρή παροχή ισχύος.

Λέξεις κλειδιά: μικροελεγκτές, Arduino, συσκευές εντοπισμού θέσης, GPS, GSM



## **ABSTRACT**

The purpose of this master thesis is to propose an intelligent and economical device, consisting of embedded systems and having the ability to find the location at which it is located at any time and to send this data as text message to mobile phones and computers. This device is capable of local self-organization and will be integrated into a multifunctional base station for applications in passenger and freight transport.

For the construction of this device, an important research was carried out to find the appropriate embedded systems that will be used. Initially, we searched for an appropriate microcontroller which should be open source platform, intelligent and economical. From this search, we ended up using the Arduino Uno, which is a single-board microcontroller, and more specifically a simple open source motherboard with a built-in microcontroller and inputs / outputs, which can be programmed with the Wiring language. Then we searched to find an appropriate embedded system that we will connect with the Arduino and it will give the location (GPS). After careful research, we chose Adafruit's Ultimate GPS board, which we connected with the Arduino and then we programmed it to send us the location whenever we want. Finally, the data we receive from the GPS have to be sent as text message to a mobile phone and to a computer. That's why we searched for a suitable GSM board with sim card insertion capability in order to connect it with the rest of the system and to send the location data. So, we chose Arduino GSM Shield 2, and with proper programming, we succeed our system to communicate with a mobile phone and a computer.

The system we have built can be placed on public and freight transports, such as train wagons and ship's merchandise, so that they can be tracked at any time and send a location signal to the multifunctional station, resulting to proper coordination during transport and minimizing thefts. Our construction is economical, autonomous and requires very little power.

Keywords: microcontrollers, Arduino, positioning devices, GPS, GSM





## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή της διπλωματικής μου εργασίας κ. Ηλία Κουκούτση που μου έδωσε την ευκαιρία να ασχοληθώ με αυτό το ενδιαφέρον θέμα. Η συνεργασία μας υπήρξε ιδιαίτερα εποικοδομητική και τον ευχαριστώ ιδιαίτερα για τις συμβουλές και την καθοδήγηση σε όλη τη διάρκεια εκπόνησης της διπλωματικής αυτής εργασίας. Επίσης ένα μεγάλο ευχαριστώ στους διδακτορικούς φοιτητές Μπαμπαλή Χαράλαμπο και Φωτόπουλο Ευάγγελο, για τις πολύτιμες οδηγίες και στήριξη που μου έδωσαν όλο αυτό το διάστημα. Χωρίς αυτούς και το πολύτιμο χρόνο που μου αφιέρωσαν η διεκπεραίωση της παρούσης εργασίας θα ήταν πολύ δύσκολη.

Τέλος, ένα μεγάλο ευχαριστώ στους γονείς, την οικογένειά μου και τους φίλους μου για όλη τη στήριξη, οικονομική και κυρίως ψυχολογική, που μου παρείχαν όλα αυτά τα χρόνια των σπουδών μου.

## ΠΕΡΙΕΧΟΜΕΝΑ

<b>ΚΕΦΑΛΑΙΟ 1 - ΕΙΣΑΓΩΓΗ</b> .....	13
1.1 Εισαγωγικά - Κενό στην Αγορά .....	13
1.2 Αντικείμενο Διπλωματικής .....	15
1.3 Συνεισφορά .....	15
1.4 Οργάνωση Εργασίας .....	16
<b>ΚΕΦΑΛΑΙΟ 2 – STATE OF THE ART</b> .....	17
2.1 Τηλεματική .....	17
2.2 Παρακολούθηση Στόλου (Fleet monitoring).....	17
2.3 Ενσωματωμένα Συστήματα.....	18
2.4 Έρευνα Αγοράς .....	19
2.4.1 Τι είναι κριτήριο .....	19
2.4.2 Λίστα κριτηρίων .....	19
2.4.3 Ευαισθησία GPS .....	20
2.4.4 Αυτονομία .....	20
2.4.5 Υποστήριξη GSM (περιλαμβάνεται το GPRS).....	20
2.4.6 Διαδικτυακή διασύνδεση χρήστη .....	21
2.4.7 Κόστος .....	21
2.4.8 Σύγκριση μικροελεγκτών.....	21
2.4.9 Σύγκριση συσκευών GPS .....	22
2.4.10 Επιλογή κατάλληλης πλακέτας GSM/GPRS.....	24
2.5 Arduino UNO .....	26
2.6 GPS.....	28
2.6.1 Geolocation .....	28
2.6.2 Geocoding.....	29
2.6.3 Ultimate GPS της Adafruit .....	29
2.6.4 Σχετικές Εφαρμογές GPS .....	31
2.7 GSM .....	31
2.7.1 Επικοινωνία μέσω γραπτών μηνυμάτων και φωνητικής κλήσης .....	32
2.7.2 Arduino GSM Shield 2.....	32
2.7.3 Σχετικές Εφαρμογές με GPS και GSM.....	34
2.8 Επικοινωνία Συσκευής με Ηλεκτρονικό Υπολογιστή .....	35
2.8.1 Πλατφόρμα MobileGo.....	36
2.9 Dual-Port Memory.....	37

2.9.1 Προτεινόμενες Dual-Port RAMs .....	38
2.10 Σχετικές Ενδιαφέρουσες Συσκευές .....	39
2.10.1 Κυψελοειδές σύστημα GPS για παρακολούθηση οχημάτων .....	39
2.10.2 GPS tracker με χρήση της τεχνολογίας RFID .....	40
2.10.3 Αποστολή δεδομένων τοποθεσίας μέσω Wifi .....	40
2.10.4 Αποστολή δεδομένων μέσω Ethernet (LAN).....	40
2.10.5 Συνδυασμένο σύστημα εντοπισμού GPS και σύστημα επικοινωνιών που χρησιμοποιεί κοινόχρηστο κύκλωμα .....	41
2.10.6 Παρακολούθηση οχήματος με GPS, αυτόματη μεταφόρτωση και απομακρυσμένη χειραγώγηση .....	42
2.10.7 GPS με βοήθεια E-Blocks.....	42
2.11 FPGA .....	43
2.11.1 Ασφάλεια στο hardware με τη βοήθεια FPGA.....	43
2.12 AES Algorithm (Advanced Encryption Standard) – Ασφάλεια στο Hardware .....	45
<b>ΚΕΦΑΛΑΙΟ 3 - ΠΡΟΣΕΓΓΙΣΗ</b> .....	46
3.1 Ελεύθερο Λογισμικό.....	46
3.2 Αρχική Σύνδεση του GPS με το Arduino.....	47
3.3 Αρχική Σύνδεση του GSM με το Arduino .....	49
3.4 Τελική Σύνδεση .....	49
3.4.1 Τελικές Προσθήκες στην Σύνδεση.....	51
3.5 Σύνδεση κινητού τηλεφώνου με Η/Υ με αποτέλεσμα την επικοινωνία της συσκευής με τον Η/Υ.....	52
<b>ΚΕΦΑΛΑΙΟ 4 – ΥΛΟΠΟΙΗΣΗ</b> .....	54
4.1 Λήψη Στίγματος Τοποθεσίας - Με Αποστολή Μηνύματος.....	54
4.1.1 Φωτογραφίες Αποτελεσμάτων από Πρώτη Λειτουργία Α΄ Κώδικα.....	55
4.2 Λήψη Ομάδας Στιγμάτων Τοποθεσίας Με Χρήση Κουμπιών.....	59
4.2.1 Φωτογραφίες Αποτελεσμάτων από Δεύτερη Λειτουργία Α΄ Κώδικα.....	60
4.3 Συνεχόμενη Λήψη Ομάδας Στιγμάτων Τοποθεσίας .....	63
4.4 Συνεχής Αποστολή Στιγμάτων Τοποθεσίας με Ταυτόχρονη Αποστολή του Χρόνου ....	64
4.5 Χρονικά Ελεγχόμενη και Συνεχής Αποστολή Στιγμάτων Τοποθεσίας.....	65
4.6 Χρονικά Ελεγχόμενη και Συνεχής Αποστολή Ομάδας Στιγμάτων Τοποθεσίας.....	68
4.7 Αποστολή Δεδομένων και Αποθήκευσή τους σε Η/Υ .....	72
4.8 Πακέτο Μηνυμάτων .....	77
<b>ΚΕΦΑΛΑΙΟ 5 – ΑΠΟΤΕΛΕΣΜΑΤΑ</b> .....	78
5.1 Αποτελέσματα από Α΄ και Β΄ Κώδικα .....	78
5.2 Αποτελέσματα Γ΄, Δ΄ και Ε΄ Κώδικα .....	79

5.3 Οικονομική Συσκευή – Χαμηλή Παροχή Ισχύος.....	79
<b>ΚΕΦΑΛΑΙΟ 6 – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ.....</b>	<b>80</b>
6.1 Ενσωμάτωση Διαδραστικών Χαρτών.....	80
6.2 Αισθητήρας Απόστασης.....	80
6.2.1 Υλοποίηση Αισθητήρα.....	81
6.2.2 Φωτογραφίες από τη Διάταξη και από την Εκτέλεση του Κώδικα.....	84
6.3 Επιτάχυνση-Επιβράδυνση.....	85
6.4 Βελτίωση Σήματος.....	87
<b>ΠΑΡΑΡΤΗΜΑ Α.....</b>	<b>88</b>
GPS.....	88
GSM.....	93
<b>ΠΑΡΑΡΤΗΜΑ Β.....</b>	<b>96</b>
ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Α.....	96
ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Β.....	100
ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Γ.....	104
ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Δ.....	108
ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Ε.....	112
<b>ΠΑΡΑΡΤΗΜΑ Γ.....</b>	<b>116</b>
ΚΩΔΙΚΑΣ ΓΙΑ ΕΝΣΩΜΑΤΩΣΗ ΧΑΡΤΗ.....	116
<b>ΠΑΡΑΡΤΗΜΑ Δ.....</b>	<b>117</b>
ΚΩΔΙΚΑΣ ΑΙΣΘΗΤΗΡΑ.....	117
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ.....</b>	<b>120</b>

## ΚΕΦΑΛΑΙΟ 1 - ΕΙΣΑΓΩΓΗ

### 1.1 Εισαγωγικά - Κενό στην Αγορά

Αρχικά θα αναφερθούμε στην έννοια της παρακολούθησης στόλου (fleet management). Πρόκειται ουσιαστικά για τη παρακολούθηση-διαχείριση επαγγελματικών οχημάτων όπως αυτοκίνητα, φορτηγά, τραίνα, εξειδικευμένων οχημάτων (όπως κινητά μηχανήματα κατασκευής) και ρυμουλκούμενων. Ακόμα έχει εφαρμογή και σε ιδιωτικά οχήματα που χρησιμοποιούνται για επαγγελματικούς σκοπούς, σε μηχανήματα διάσπασης όπως αεροσκάφη (αεροπλάνα και ελικόπτερα), σε πλοία και σε σιδηροδρομικά αυτοκίνητα. [1]

Η διαχείριση του στόλου (οχήματος) μπορεί να περιλαμβάνει μια σειρά από λειτουργίες όπως χρηματοδότηση οχημάτων, συντήρηση οχημάτων, τηλεματική (παρακολούθηση και διάγνωση) οχημάτων, διαχείριση οδηγών, διαχείριση ταχύτητας, διαχείριση καυσίμων και διαχείριση της υγείας και της ασφάλειας. Η διαχείριση στόλου είναι μια λειτουργία που επιτρέπει σε εταιρείες που βασίζονται σε μεταφορές στις επιχειρήσεις να απομακρύνουν ή να ελαχιστοποιήσουν τους κινδύνους που συνδέονται με τις επενδύσεις σε οχήματα, βελτιώνοντας την αποδοτικότητα, την παραγωγικότητα και μειώνοντας το συνολικό κόστος μεταφοράς και προσωπικού τους, και πολλά άλλα. Αυτές οι λειτουργίες μπορούν να αντιμετωπιστούν είτε από ένα εσωτερικό τμήμα διαχείρισης στόλου είτε από έναν εξωτερικό πάροχο διαχείρισης στόλου. Σύμφωνα με έρευνα της αγοράς από την ανεξάρτητη εταιρεία αναλύσεων Berg Insight, ο αριθμός των μονάδων διαχείρισης στόλου που αναπτύσσονται σε εμπορικούς στόλους στην Ευρώπη έχει αυξηθεί από 3,05 εκατομμύρια μονάδες στο τέλος του 2012 σε 6,40 εκατομμύρια το 2017. [1]

Ένα σύστημα πληροφοριών διαχείρισης στόλου οχημάτων προσδιορίζει την τοποθεσία και την κατεύθυνση της κίνησης κάθε οχήματος σε ένα στόλο σε πραγματικό χρόνο και αναφέρει αυτόματα τις πληροφορίες αυτές καθώς και την κατάσταση προκαθορισμένων συμβάντων στα οποία εμπλέκεται το όχημα απευθείας στον διαχειριστή στόλου. Κάθε όχημα με στόλο διαθέτει καθορισμένο χρονικό διάστημα για τη μετάδοση των πληροφοριών αναφοράς του σε ένα δίκτυο επικοινωνιών χωρίς να παρεμβαίνει στις μεταδόσεις από άλλα οχήματα στις δικές τους αντίστοιχες χρονικές θυρίδες. Ένας βρόχος κλειδώματος φάσης ελέγχου χρονισμού (PLL) παρέχει ακριβή συγχρονισμό χρόνου για διορθώσεις χρονισμού από μια αναφορά χρόνου που βασίζεται στο σύστημα παγκόσμιου εντοπισμού θέσης (GPS). Οι μονάδες επεξεργασίας χρόνου του μικροεπεξεργαστή σε στοιχεία του δικτύου εκτελούν ακριβή συγχρονισμό ρολογιού. Η διαστημική πολυμορφία που εκτελείται στα ληφθέντα μηνύματα που μεταδίδονται από το όχημα αποφεύγει τη διαφθορά δεδομένων. Τα διαφορετικά οχήματα έχουν διαφορετικά περιοδικά διαστήματα μετάδοσης, με τη δυναμική κατανομή των διαθέσιμων χρόνων για διάφορες ταχύτητες ενημέρωσης. Οι βοηθητικές θυρίδες αναφοράς επιτρέπουν την άμεση αναφορά σημαντικών δεδομένων από τους αντίστοιχους πομπούς οχήματος ανεξάρτητα από τα πιο αργά περιοδικά διαστήματα μετάδοσης [2]. Υπάρχουν πολλοί τρόποι παρακολούθησης στόλου.

Προτείνουμε παρακολούθηση στόλου με τη βοήθεια ενσωματωμένων συστημάτων και πιο συγκεκριμένα θα χρησιμοποιήσουμε μικροελεγκτή Arduino και θα αξιοποιήσουμε το Global Positioning System (GPS) και το Global System for Mobile Communications (GSM) ώστε να στέλνονται τα σήματα της τοποθεσίας σε υπολογιστή και κινητό τηλέφωνο. Τα ενσωματωμένα αυτά συστήματα θα ενταχθούν σε μέσα μεταφοράς επιβατών και εμπορευμάτων και θα στέλνουν σήμα τοποθεσίας στον σταθμό, προκειμένου να ελέγχεται διαρκώς η πορεία τους. Τα ενσωματωμένα αυτά συστήματα θα πρέπει να είναι αυτόνομα, οικονομικά και χαμηλής παροχής ισχύος προκειμένου να είναι λειτουργικά και να βγουν στην αγορά ώστε να επιτευχθεί η πρόοδος των συγκοινωνιών και των μεταφορών. Ενδεικτικά, παρουσιάζουμε δύο πολύ ενδιαφέροντα παραδείγματα αναγκών που θα εξυπηρετηθούν με την βοήθεια της συσκευής που θα υλοποιήσουμε:

*Βαγόνια σε τρένα:* Οι ανάγκες σε αυτήν την κατηγορία αλλάζουν συνεχώς λόγω της φύσης των υπηρεσιών και των αγαθών που μεταφέρονται. Πιο συγκεκριμένα, υπάρχουν αγαθά τα οποία χρειάζονται συγκεκριμένη μεταχείριση προκειμένου να εξασφαλιστεί η ασφαλής και απαιτούμενη μετακίνησή τους. Για παράδειγμα, αγαθά που χρειάζονται συγκεκριμένη θερμοκρασία πρέπει να βρίσκονται σε βαγόνι-ψυγείο, το οποίο θα πρέπει να βρίσκεται έτοιμο να παραλάβει τα προϊόντα τη στιγμή που φτάνουν στο σταθμό του τρένου. Χώρος σε τέτοιο βαγόνι μπορεί να υπάρχει σε τρένο το οποίο πλησιάζει τη στάση, οπότε μπορεί απλά όταν φτάσει, να τοποθετηθούν τα προϊόντα και να υπάρχουν οι επιθυμητές συνθήκες. Αν, όμως, το τρένο πρόκειται να μην φτάσει στην προκαθορισμένη ώρα, τότε θα πρέπει να βρεθεί εναλλακτική λύση, δηλαδή να ενεργοποιηθεί νέο βαγόνι-ψυγείο, να τοποθετηθούν εκεί τα προϊόντα και μόλις φτάσει το τρένο να συνδεθεί το βαγόνι σε αυτό και να συνεχιστεί η πορεία του. Όλη αυτή η διαδικασία, η οποία μπορεί να φαίνεται απλή για εύρεση λύσης, όταν προστεθεί με άλλες πολυάριθμες ανάλογες διαδικασίες, δημιουργούν ένα πολύπλοκο σύνολο προβλημάτων το οποίο έχει πολλές λύσεις. Με την συνεχή παρακολούθηση, λοιπόν, των βαγονιών του τρένου, ώστε να μπορούμε να γνωρίζουμε ανά πάσα στιγμή της τοποθεσία τους, βοηθάει στην επίλυση του προβλήματος αυτού. Επίσης, σε περιπτώσεις μη αναμενόμενων γεγονότων, όπως βλάβης κάποιου βαγονιού ή κλοπής του κλπ., μπορεί να γίνει άμεσα η ενημέρωση της κατάστασης των βαγονιών στον πολυλειτουργικό σταθμό, ότι κάποιο βαγόνι δεν ακολουθεί τον υπόλοιπο συρμό ή έχει κάποιο πρόβλημα. Αυτό που θα γίνεται πρακτικά με την συσκευή μας, είναι να παρακολουθούνται διαρκώς τα βαγόνια τραίνων και όταν κάποιο θα αποσυνδέεται από τον συρμό για τον οποιοδήποτε λόγο να ειδοποιείται ο κεντρικός σταθμός προκειμένου να λάβει δράση

*Σειρά φορτηγών μεταφορών,* το ένα πίσω από το άλλο: Σε αυτήν την περίπτωση μπορούν να δημιουργηθούν προβλήματα διαφορετικής φύσης, μερικά από τα οποία είναι τα εξής. Πρώτον, τα φορτηγά μεταξύ τους χρειάζεται να έχουν μία προκαθορισμένη απόσταση, ώστε να γνωρίζουν τις συνθήκες που βρίσκεται το καθένα. Αν αλλάξουν οι αποστάσεις, θα μπορεί εύκολα με την συσκευή που θα υλοποιήσουμε, να γίνει ενημέρωση της αλλαγής και να γίνει προσαρμογή. Δεύτερον, μπορεί να υπάρξει επιθυμία ή και ανάγκη αλλαγής δρομολογίου, για διάφορους λόγους, από αλλαγή παραλήπτη μέχρι και ατυχήματα στο δρόμο. Προκειμένου να μπορεί να παρθεί γρήγορα η απόφαση αλλαγής και να ενημερωθούν άμεσα όλα τα

φορτηγά, είναι χρήσιμο να υπάρχει άμεση ενημέρωση, ώστε μόλις παρθεί η απόφαση είτε από κάποιο φορτηγό είτε από τον κεντρικό σταθμό οργάνωσης, να μπορέσουν όλα τα φορτηγά να προσαρμόσουν την πορεία τους. Τρίτον, όπως και με τα βαγόνια του τρένου, αν υπάρξουν μη αναμενόμενα γεγονότα στα φορτηγά, όπως βλάβη κάποιου ψυγείου ή βλάβη της μηχανής του φορτηγού με αποτέλεσμα να μην είναι δυνατή η παροχή ηλεκτρικής ισχύος στο εμπορευματοκιβώτιό του, είναι σημαντικό να βρεθεί άμεση λύση ώστε να μην υπάρξουν απώλειες ή τουλάχιστο να ελαχιστοποιηθούν.

Για τους λόγους αυτούς είναι απαραίτητη η παρακολούθηση των στόλων και η αποστολή των δεδομένων τοποθεσίας ανά πάσα στιγμή σε πολυ-λειτουργικό σταθμό, με σκοπό την άμεση επίλυση τυχόν αναγκών και προβλημάτων. Με τη βοήθεια της συσκευής που θα υλοποιήσουμε, θα έχουμε τη δυνατότητα να παρακολουθούμε ανά πάσα στιγμή οποιοδήποτε στόλο και έτσι θα μπορούμε να επεμβούμε άμεσα σε τυχόν πρόβλημα που μπορεί να προκύψει. Έτσι καλύπτουμε αυτό το σημαντικό κενό της αγοράς.

### 1.2 Αντικείμενο Διπλωματικής

Στη συγκεκριμένη διπλωματική κληθήκαμε να υλοποιήσουμε μία συσκευή αποτελούμενη από ενσωματωμένα συστήματα, που θα έχει τη δυνατότητα εντοπισμού της τοποθεσίας με χρήση GPS και αποστολής των δεδομένων (συντεταγμένων) αυτών με γραπτό μήνυμα σε κινητό τηλέφωνο και υπολογιστή με τη βοήθεια GSM, με αποτέλεσμα την real-time επικοινωνία μεταξύ των συσκευών. Η αποστολή των δεδομένων στο κινητό τηλέφωνο και στον υπολογιστή είναι ουσιαστικά η προσομοίωση της αποστολής των σημάτων τοποθεσίας στον πολυλειτουργικό σταθμό. Η πρώτη δυσκολία που προκύπτει σε αυτή την υλοποίηση είναι η επιλογή κατάλληλων ενσωματωμένων συστημάτων τα οποία πρέπει πρώτα απ' όλα να είναι ευφυή, ανοιχτού κώδικα και προφανώς να είναι οικονομικά, καθώς ο αριθμός των μέσων μεταφοράς στα οποία θα ενταχθούν είναι ιδιαίτερα μεγάλος οπότε θα χρησιμοποιηθούν πολυάριθμα τέτοια συστήματα. Συνεπώς πρέπει να είναι χαμηλού κόστους ώστε να μπορεί να αγοραστεί μεγάλη ποσότητα. Επιπροσθέτως, πρέπει τα ενσωματωμένα αυτά συστήματα να λειτουργούν με χαμηλή παροχή ρεύματος, ώστε και σε αυτόν τον τομέα να έχουμε εξοικονόμηση ενέργειας, καθώς μεγάλη παροχή ηλεκτρικής ισχύος οδηγεί άμεσα σε περισσότερες δαπάνες.

### 1.3 Συνεισφορά

Η συνεισφορά της διπλωματικής συνοψίζεται ως εξής:

1. Μελετήσαμε μικροελεγκτές προκειμένου να βρούμε τον καταλληλότερο ώστε να είναι ευφυής και οικονομικός.
2. Επιλέξαμε το Arduino Uno.
3. Αναζητήσαμε κατάλληλο σύστημα εντοπισμού θέσης (GPS) που θα συνδεθεί με το Arduino και θα μας δίνει την τοποθεσία.
4. Επιλέξαμε το GPS της Adafruit.

5. Προγραμματίσαμε το Arduino με τον κατάλληλο αλγόριθμο με αποτέλεσμα το GPS να μας δίνει την τοποθεσία.
6. Αναζητήσαμε στη συνέχεια κατάλληλη πλακέτα GSM που θα συνδεθεί με το υπόλοιπο σύστημα και θα στέλνει τα σήματα τοποθεσίας σε κινητό τηλέφωνο και υπολογιστή.
7. Επιλέξαμε το Arduino GSM shield 2.
8. Συμπεριλάβαμε στον ήδη υπάρχον κώδικα τον κατάλληλο αλγόριθμο ώστε να πετύχουμε την επικοινωνία του συστήματός μας με κινητό τηλέφωνο και υπολογιστή.
9. Συγχωνεύσαμε με κατάλληλο τρόπο και αλλαγές τους δύο κώδικες, προκειμένου τα στίγματα τοποθεσίας να στέλνονται με γραπτό SMS σε κινητό τηλέφωνο.
10. Υλοποιήσαμε συνολικά πέντε (5) διαφορετικούς κώδικες, καθένας από τους οποίους υλοποιεί και μία διαφορετική, πολύ χρήσιμη λειτουργία για την real-time επικοινωνία της συσκευής μας και του κινητού τηλεφώνου.
11. Μεταφορά των SMS από το κινητό στον Ηλεκτρονικό Υπολογιστή με μια επιφάνεια εργασίας λογισμικού του κινητού (Android ή iOS) και ταυτόχρονη αποθήκευση των SMS σε αρχεία txt. Αποτέλεσμα ήταν ότι τελικά επιτύχαμε αποστολή των δεδομένων της συσκευής μας και αποθήκευσή των αντίστοιχων συντεταγμένων στον υπολογιστή.

#### 1.4 Οργάνωση Εργασίας

Κεφάλαιο 2: State of Art: Στο κεφάλαιο αυτό περιέχονται πληροφορίες σχετικά με το τεχνολογικό, επιστημονικό και θεωρητικό προσκήνιο των αντικειμένων της παρούσας εργασίας, κυρίως πληροφορίες που έχουν βρεθεί μετά από αναζήτηση και έρευνα στο διαδίκτυο.

Κεφάλαιο 3: Προσέγγιση: Στο κεφάλαιο αυτό αναλύουμε λεπτομερώς τα ενσωματωμένα συστήματα που χρησιμοποιήσαμε για την υλοποίηση της συσκευής αυτής και το πως προσεγγύσαμε το θέμα.

Κεφάλαιο 4: Υλοποίηση: Στο κεφάλαιο αυτό δείχνουμε αναλυτικά την υλοποίηση που κάναμε με αποτέλεσμα την ορθή λειτουργία της συσκευής μας και την επίλυση του θέματος

Κεφάλαιο 5: Αποτελέσματα: Στο κεφάλαιο αυτό παρουσιάζουμε τα αποτελέσματα της εργασίας που κάναμε και τα συμπεράσματα που εξήγαμε από την υλοποίηση της συσκευής αυτής.

Κεφάλαιο 6: Μελλοντική Δουλειά: Στο κεφάλαιο αυτό παρουσιάζουμε μελλοντικές επεκτάσεις της διπλωματικής αυτής, καθώς και πιθανές μελλοντικές ιδέες.

#### Βιβλιογραφία



## ΚΕΦΑΛΑΙΟ 2 – STATE OF THE ART

Σε αυτό το κεφάλαιο θα γίνει μία αναφορά στην τηλεματική, στα ενσωματωμένα συστήματα, στο Arduino, στο GPS και στο GSM. Επίσης θα πούμε λίγα λόγια για τις τεχνολογίες που χρησιμοποιήθηκαν για την εκπόνηση της εργασίας και τέλος για τις σχετικές εφαρμογές που υπάρχουν ήδη.

### 2.1 Τηλεματική

Η τηλεματική, δηλαδή η παρακολούθηση στόλου είναι μια ερευνητική περιοχή άμεσα συνυφασμένη με την συγκεκριμένη διπλωματική εργασία, καθώς ασχολείται με την παρακολούθηση οχημάτων από απομακρυσμένο σημείο με στόχο τη βελτιστοποίηση κάποιων λειτουργιών.

Για αρχή θα πρέπει να γίνει μια αναφορά στην τηλεματική, καθώς η συγκεκριμένη διπλωματική αποτελεί μια σημαντική εφαρμογή της. Η τηλεματική είναι ο διεπιστημονικός τομέας που περιλαμβάνει τις τηλεπικοινωνίες, τις τεχνολογίες των οχημάτων, τις οδικές μεταφορές, την οδική ασφάλεια, την ηλεκτρολογία (αισθητήρες, όργανα, ασύρματες επικοινωνίες κ.λπ.) και την επιστήμη των υπολογιστών (πολυμέσα, Διαδίκτυο κ.λπ.), με σκοπό την παρακολούθηση στόλων/οχημάτων. Οι τηλεματικές εφαρμογές κερδίζουν συνεχώς έδαφος στο σύγχρονο κόσμο αλλάζοντας ριζικά τους τρόπους επικοινωνίας και μετάδοσης πληροφοριών. [6] Η τηλεματική μπορεί να περιλαμβάνει οποιοδήποτε από τα υπόλοιπα:

- Την τεχνολογία της αποστολής, λήψης και αποθήκευσης πληροφοριών μέσω συσκευών τηλεπικοινωνιών σε συνδυασμό με την πραγματοποίηση ελέγχου σε απομακρυσμένα αντικείμενα
- Την ολοκληρωμένη χρήση των τηλεπικοινωνιών και της πληροφορικής για εφαρμογή σε οχήματα και με έλεγχο των οχημάτων σε κίνηση
- Τη τεχνολογία GNSS που ενσωματώνεται με υπολογιστές και τεχνολογία κινητών επικοινωνιών σε συστήματα πλοήγησης για αυτοκίνητα
- Τη χρήση τέτοιων συστημάτων στα οδικά οχήματα, που ονομάζεται επίσης τηλεματική των οχημάτων.[7]

### 2.2 Παρακολούθηση Στόλου (Fleet monitoring)

Η χρήση κατάλληλου συστήματος τηλεματικής μας δίνει την δυνατότητα να αποκτήσουμε τον πλήρη έλεγχο της δραστηριότητας των οχημάτων και των οδηγών, από τον υπολογιστή μας από όπου και αν βρισκόμαστε. Έτσι γνωρίζουμε κάθε στιγμή τη θέση του κάθε οχήματος για να ξέρουμε δηλαδή κάθε στιγμή που βρίσκεται το όχημα, την διαδρομή του σε πραγματικό χρόνο, οποιαδήποτε απόκλιση από το πρόγραμμα δρομολογίων, πότε ακριβώς θα φτάσει το όχημα κάπου και πότε θα επιστρέψει το όχημα ώστε να προγραμματίσουμε την επόμενη του διαδρομή. Αυτό έχει ορισμένα πολύ χρήσιμα αποτελέσματα όπως:

- Με τον καλύτερο προγραμματισμό να πραγματοποιούνται περισσότερα δρομολόγια σε λιγότερο χρόνο
- Μπορούμε να γνωρίζουμε για ποιο λόγο σταμάτησε το όχημα κατά τη διάρκεια της διαδρομής του
- Μπορούμε να ελέγξουμε την διάρκεια της στάσης του και την ακριβή διεύθυνση στάσης
- Εξάλειψη άσκοπων διαδρομών, μείωση διανυθέντων χιλιομέτρων, μείωση καυσίμων, άρα οικονομία
- Τέλος, επιτυγχάνεται η βελτίωση τρόπου οδήγησης οδηγών, η μείωση ασφαλιστρών, ο άμεσος εντοπισμός και ακινητοποίηση οχήματος σε περίπτωση κλοπής, η προστασία οχήματος και φορτίου [8].

Εμείς χρησιμοποιώντας ενσωματωμένα συστήματα, θα φτιάξουμε μία κατάλληλη συσκευή παρακολούθησης στόλου η οποία θα έχει αρχική εφαρμογή σε βαγόνια τραίνων, και θα μας δίνει ανά πάσα στέλνει ανά πάσα στιγμή την τοποθεσία με γραπτό μήνυμα σε κινητό τηλέφωνο και σε Ηλεκτρονικό Υπολογιστή. [11]

### 2.3 Ενσωματωμένα Συστήματα

Ενσωματωμένο σύστημα είναι το σύστημα το οποίο περιλαμβάνει έναν προγραμματιζόμενο υπολογιστή ο οποίος δεν είναι υπολογιστής γενικού σκοπού. Μπορεί δηλαδή να θεωρηθεί ως ένα σύστημα υλικού υπολογιστή που έχει ενσωματωμένο λογισμικό σε αυτό. Ένα ενσωματωμένο σύστημα μπορεί να είναι ένα ανεξάρτητο σύστημα ή μπορεί να είναι μέρος ενός μεγάλου συστήματος.

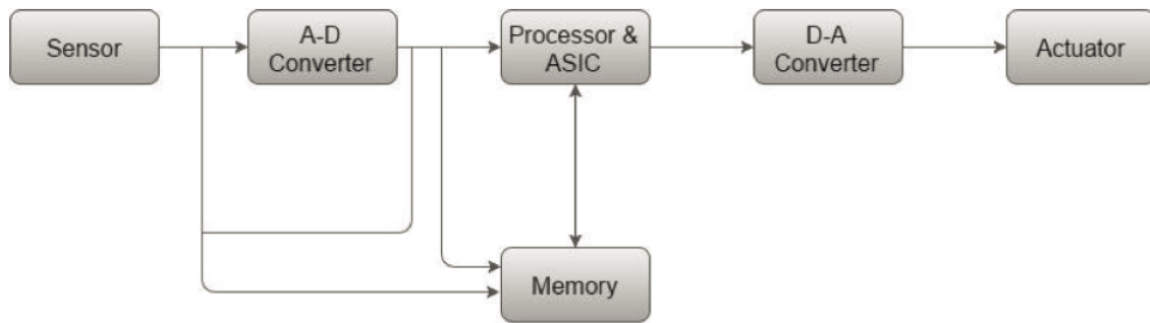
Ο πυρήνας του ενσωματωμένου συστήματος αποτελείται από ένα τουλάχιστον μικροελεγκτή ή μικροεπεξεργαστή. Το μικρό φυσικό μέγεθος, το χαμηλό κόστος, η χαμηλή κατανάλωση ενέργειας είναι μερικά χαρακτηριστικά ενσωματωμένου συστήματος.

Αφού τα ενσωματωμένα συστήματα είναι αφιερωμένα σε συγκεκριμένες εργασίες, δίνει τη δυνατότητα στους μηχανικούς που τα σχεδιάζουν να τα βελτιώνουν, να μειώνουν το μέγεθος και το κόστος, και να αυξάνουν την απόδοση και την αξιοπιστία τους.

Η γενική λοιπόν λογική πίσω από αυτά τα συστήματα είναι να εκτελούν σωστά και αποδοτικά την εφαρμογή για την οποία έχουν σχεδιαστεί. Όσον αφορά τις αρχιτεκτονικές που χρησιμοποιούνται στο σχεδιασμό των ενσωματωμένων συστημάτων, ενδεικτικά αναφέρουμε ότι μερικές από αυτές είναι οι ARM, MIPS, POWERPC, X86.

Αν εξετάσουμε την καθημερινή ζωή μας θα διαπιστώσουμε πως αλληλοεπιδρούμε με ενσωματωμένα συστήματα συνέχεια. Μερικά παραδείγματα ενσωματωμένων συστημάτων είναι οι κονσόλες παιχνιδιών (π.χ. Sony PlayStation, Xbox), οι οικιακές συσκευές(κουζίνα, ψυγείο, τηλεοράσεις), συστήματα στο αυτοκίνητο (ABS, ESP), ψηφιακή φωτογραφική μηχανή, tablets, κινητά τηλέφωνα και πολλά άλλα. [12]

## Basic Structure of an Embedded System



### 2.4 Έρευνα Αγοράς

Στο σημείο διεξάγαμε μια εκτενή έρευνα αγοράς για την επιλογή κατάλληλου μικροελεκτη σε συνδυασμό με κατάλληλη πλακέτα GPS και κατάλληλη πλακέτα GSM. Η επιλογή αυτή έγινε με βάση κάποια συγκεκριμένα κριτήρια.

#### 2.4.1 Τι είναι κριτήριο

Προκειμένου να παρουσιάσουμε μια πλήρη, λεπτομερή σύγκριση, πρέπει πρώτα να αντιμετωπίσουμε τα κριτήρια στα οποία η συσκευή μπορεί να ανταποκριθεί. Τα κριτήρια δεν είναι ένα σύνολο χαρακτηριστικών που μπορεί να έχει μια συσκευή. Ένα κριτήριο είναι μία αρχή ή ένα πρότυπο με βάση το οποίο μπορεί να κριθεί ή να αποφασιστεί κάτι. Με βάση τα κριτήρια θα γίνει, λοιπόν, η επιλογή της συσκευής. Συνεπώς, τα κριτήρια πρέπει να είναι παγκόσμια για όλες τις συσκευές. Επιπλέον, όλες οι συσκευές θα πρέπει να οδηγούν σε βαθμολογία σχετικά με τα κριτήρια επιλογής.

#### 2.4.2 Λίστα κριτηρίων

Οι συσκευές γίνονται αποδεκτές αν πληρούν τα κριτήρια που περιγράφονται παρακάτω. Με λίγα λόγια, τα κριτήρια (ταξινομημένα από τη σημασία) είναι:

1. Υποστήριξη GSM
2. Διαδικτυακή διασύνδεση χρήστη
3. Αυτονομία
4. Ευαισθησία GPS
5. Θέση ακρίβειας
6. Τιμή

Στο σημείο αυτό θα παρουσιάσουμε αναλυτικά την σημασία των παραπάνω κριτηρίων. [31]

### 2.4.3 Ευαισθησία GPS

Η ευαισθησία GPS είναι η δυνατότητα ενός δέκτη GPS να χρησιμοποιεί μεγάλες τράπεζες συσχετιστών και να εκτελεί ψηφιακή επεξεργασία σήματος για πολύ γρήγορη αναζήτηση σημάτων GPS. Αυτό είναι ένα κριτήριο για μια συσκευή GPS. Ένα GPS υψηλής ευαισθησίας:

- μπορεί να υπολογίσει πολύ γρήγορα μια λύση θέσης όταν είναι σε εξωτερικούς χώρους.
- μπορεί να χρησιμοποιήσει μεγαλύτερη ισχύ επεξεργασίας από άλλες συσκευές GPS για να παρέχει μια θέση λύση σε εσωτερικούς χώρους.

Πιστεύουμε ότι η ευαισθησία GPS είναι ένα κριτήριο, επειδή είναι η δυνατότητα που έχει μια συσκευή GPS να εντοπίζει και να ανιχνεύει πολύ αδύναμα σήματα GPS. Μας ενδιαφέρει ιδιαίτερα η παρακολούθηση πολύ αδύναμων σημάτων GPS, γιατί στα τρένα οι διαδρομές στόλου περνούν μέσα από σήραγγες ή κάτω από δέντρα ή άλλα εμπόδια που εμποδίζουν την ορατότητα του δορυφόρου και επομένως το σήμα εκεί είναι αποδυναμωμένο. Έτσι, πρέπει να εξασφαλίσουμε τη συνεχή παρακολούθηση στόλου. Διασφαλίζουμε, λοιπόν, τη συνεχή και ανοιχτή παρακολούθηση επιλέγοντας μια συσκευή GPS με υψηλή ευαισθησία.

### 2.4.4 Αυτονομία

Αυτός ο όρος δεν αναφέρεται μόνο στα χαρακτηριστικά και τις ιδιότητες της μπαταρίας, αλλά στις ανάγκες σε ενέργεια της συσκευής. Με άλλα λόγια, μια συσκευή GPS χρειάζεται να εφαρμόσει τεχνολογία εξαιρετικά χαμηλής ισχύος. Αυτό είναι άλλο σημαντικό κριτήριο, διότι:

- Η ισχύς κατά τη ζήτηση δεν είναι πάντα παρούσα. Τα βαγόνια συνήθως δεν παρέχουν καμία τροφοδοσία.
- Οι στόλοι των αμαξοστοιχιών ταξιδεύουν για πολλές ημέρες.
- Οι τελικοί χρήστες απαιτούν συνεχή παρακολούθηση και καταγραφές ταξιδιών σε οποιοδήποτε σημείο του ταξιδιού.

Πρέπει, λοιπόν, να διασφαλίσουμε ότι μια συσκευή είναι σε θέση να συνεχίσει να παρακολουθεί τη διαδρομή και να στέλνει μηνύματα σε ολόκληρο το ταξίδι. Η συσκευή πρέπει να είναι σε θέση να λειτουργεί περισσότερο από ό, τι προβλέπεται, καθώς μπορεί να συμβούν ατυχή περιστατικά (τυχαία ανοίγματα ή ληστείες εμπορευματοκιβωτίων κ.λπ.) τα οποία πρέπει να αντιμετωπιστούν απολύτως αποτελεσματικά.

### 2.4.5 Υποστήριξη GSM (περιλαμβάνεται το GPRS)

Μια συσκευή GPS θα πρέπει να υποστηρίζει απολύτως το GSM (Global System for Mobile Communication) προκειμένου να επικοινωνούν με τον τελικό χρήστη. Μια συσκευή που υποστηρίζει το GSM είναι σε θέση να επικοινωνήσει με τον χρήστη, χρησιμοποιώντας SMS ή ακόμα και φωνητικές εντολές. Ένα άλλο πλεονέκτημα αυτού του τύπου είναι η επικοινωνία και η άμεση ανταπόκριση σε κάθε ανάγκη του χρήστη. Τέτοιο χαρακτηριστικό είναι επίσης απαραίτητο λόγω του γεγονότος ότι

καθυστερήσεις ή άλλα απρόβλεπτα γεγονότα μπορεί να συμβούν κατά τη διάρκεια ενός ταξιδιού. Ως αποτέλεσμα, ο χρήστης θα πρέπει να είναι σε θέση να επικοινωνεί αμέσως με τον στόλο προκειμένου να δώσει μια διαφορετική πληροφορία (όπως αλλαγή διαδρομής) ή να αποκτήσει ο ίδιος μερικές χρήσιμες πληροφορίες (όπως που είναι ο στόλος τώρα). Επιπλέον, απαιτείται GPRS (Υπηρεσίες Πακέτου Ραδιοφώνου). Κατά τη διάρκεια ενός ταξιδιού, ο στόλος μπορεί να ταξιδέψει σε πολλές διαφορετικές χώρες. Ακόμη, μπορεί να προκύψει συχνή επικοινωνία με υπηρεσίες GSM, οι οποίες απαιτούν πολύ μεγάλες αμοιβές, όταν ο στόλος διασχίζει διάφορες χώρες (και ως εκ τούτου, πολλοί πάροχοι GSM συμμετέχουν). Το GPRS είναι μια τεχνολογία που επιτρέπει στον χρήστη να αποφύγει τέτοια ατυχή γεγονότα.

#### 2.4.6 Διαδικτυακή διασύνδεση χρήστη

Μια διεπαφή είναι απαραίτητη για να μπορεί ο χρήστης να παρακολουθεί το στόλο του και να τον παρακολουθεί σε πραγματικό χρόνο. Εκτός από μια απλή διεπαφή, μια διαδικτυακή διεπαφή έχει ακόμη περισσότερα πλεονεκτήματα καθώς δεν είναι μόνο να παρακολουθεί τη συσκευή on-line, από κάθε πιθανή συσκευή (tablet, smartphone ή pc) αλλά και να επιτρέπει επίσης την απεικόνιση του στόλου στο άθροισμα. Χωρίς web-based UI, ο χρήστης λαμβάνει μόνο τις συντεταγμένες της συσκευής. Από την άλλη πλευρά, έχοντας μια διεπαφή χρήστη που βασίζεται στον ιστό, οι συντεταγμένες τοποθετούνται σε ένα ηλεκτρονικό χάρτη και σε πραγματικό χρόνο και ο χρήστης είναι σε θέση να παρακολουθεί το στόλο του. Έτσι, μια ισχυρή, διαδικτυακή διεπαφή χρήστη αποτελεί ένα κριτήριο.

#### 2.4.7 Κόστος

Ένα άλλο ιδιαίτερα σημαντικό κριτήριο είναι η τιμή μιας συσκευής GPS. Η τιμή είναι πολύ σημαντική γιατί οι συσκευές αυτές πρέπει να αγοράζονται χύμα λόγω του αριθμού των στόλων που συνήθως χρειάζεται ο χρήστης για την παρακολούθηση. Ο χρήστης δεν είναι ένα άτομο, αλλά ένας μεγάλος οργανισμός με δεκάδες (ή εκατοντάδες) στόλους που είναι διαθέσιμοι για παρακολούθηση. Έτσι πρέπει να δοθεί ιδιαίτερη προσοχή στην τιμή. Ακόμα κι αν μια συσκευή GPS δεν είναι τόσο καλή όσο οι άλλες ή δεν έχει τόσα χαρακτηριστικά όπως οι άλλες, είναι προτιμότερο δεδομένης της ανταγωνιστικής (ή χαμηλής) του τιμής.

#### 2.4.8 Σύγκριση μικροελεγκτών

Πρώτα απ' όλα η καρδιά της συσκευής μας, η βάση δηλαδή πάνω στην οποία θα μπουν το GPS και η πλακέτα GSM, είναι ένας μικροελεγκτής. Πρέπει, λοιπόν, να κάνουμε μια πρώτη έρευνα αγοράς στους μικροελεγκτές και με βάση τα κριτήρια που αναφέραμε παραπάνω να επιλέξουμε τον καταλληλότερο. Ας δούμε, λοιπόν, τους μικροελεγκτές που υπάρχουν στην αγορά και τα κριτήρια τα οποία εκπληρούν.

Είδη Μικροελεγκτών	Κόστος (\$)	Αυτονομία (Τάση, Ένταση)	Ταχύτητα	Μνήμη Συστήματος (RAM)
Arduino Uno	9.99	5V, 20-50mA	16MHz	2kB
Raspberry Pi 3	39.99	5V, 200-400mA	900MHz	1GB
Raspberry Pi Zero	5.99	5V, 80-200mA	1GHz	512MB
BeagleBone	59.99	5V, 300mA	1GHz	512MB
pcDuino	60	5V, 2A	2GHz	1GB
Goldilocks	45	5V, 2A	20MHz	16kB
Intel Galileo	84.99	5V, 800mA	400MHz	512kB
SparkCore	39.99	5V, 500mA	72MHz	20kB
DigiSpark	7.99	5V, 500mA	16.5MHz	8kB
Teensy 3.6	29.99	5V, 50mA	180MHz	256kB
Microchip PIC	4	2- 5.5V	48MHz	2kB
MSP430	4	5V	16MHz	64kB
Intel Edison	99.99	50mA	500MHz	1GB
8051 Microcontroller KIT	85	5V, 1.5A	12MHz	32kB

[32-43]

Όπως, φαίνεται και στον παραπάνω πίνακα, τα κριτήρια που χρησιμοποιήσαμε για την επιλογή του κατάλληλου μικροελεγκτή είναι λίγο διαφορετικά από τα κριτήρια με βάση τα οποία θα επιλέξουμε το GPS και το GSM. Με βάση, λοιπόν, τα παραπάνω κριτήρια καταλήγουμε ότι η καταλληλότερη επιλογή για την εργασία μας είναι το Arduino Uno, καθώς έχει την μικρότερη παροχή ισχύος σε σχέση με τα υπόλοιπα, παράλληλα είναι ιδιαίτερα οικονομικό και έχει και ταχύτητα και μνήμη που είναι αποτελεσματικά για την εργασία μας. Επομένως, εργαστήκαμε με το Arduino Uno του οποίου οι λειτουργίες θα αναφερθούν σε επόμενο κεφάλαιο.

#### 2.4.9 Σύγκριση συσκευών GPS

Σε αυτό το σημείο της διπλωματικής κάναμε μια καλή έρευνα αγοράς στις συσκευές GPS και στις πλακέτες GSM που υπάρχουν και με βάση τα κριτήρια που αναφέρθηκαν παραπάνω θα επιλέξαμε τον καταλληλότερο συνδυασμό για την εργασία μας. Στον παρακάτω πίνακα φαίνονται οι συσκευές που βρήκαμε στην αγορά και τα αντίστοιχα κριτήρια που εκπληρούν.

GPS trackers	Υποστήριξη GSM	Διαδικτυακή διασύνδεση χρήση	Αυτονομία	Ευαισθησία GPS (dBm)	Θέση ακρίβειας (feet)	Τιμή (\$)
Amber Alert GPS	Yes	Yes	2	-160	5-10	125
BrickHouse Security Spark Nano	Yes	Yes	14	-160	9-10	99.99
Trackimo	Yes	Yes	4	-162	50	139.99
AngelSense Guardian	Yes	Yes	5	-159	10	59.99
Spy Tec GL-300	Yes	Yes	14	-148	15	49.95
Trax	Yes	No	3	-162	10-20	98.99
Spot Gen 3	Yes	Yes	180	-	30	149.95
Yepzon	Yes	No	14	-159	5	129
Ultimate Adafruit	Yes	Yes	240	-165	6	39.95
Amcrest AM-GL300	Yes	Yes	7-10	-147	30	69.99
Vectu	Yes	Yes	-	-162	200	49.99
PAM-7Q	Yes	Yes	-	-130	8	49.99
PMB-648	Yes	No	-	-159	16	19.99
SUP500F	Yes	Yes	-	-161	8	29.99
EM406	Yes	Yes	3	-163	8	39.95
Copernicus	Yes	Yes	-	-160	8	74.95
U-Blox 6M	Yes	Yes	-	-162	8	15
Adafruit FONA 808 - GPS Breakout	Yes	Yes	-	-165	8	49.95

[41-62]

Έτσι από τις παραπάνω συσκευές GPS καταλήξαμε ότι αυτή που εκπληρεί περισσότερο τα κριτήρια μας είναι το Ultimate GPS της Adafruit καθώς διαθέτει την μεγαλύτερη αυτονομία, είναι πάρα πολύ οικονομικό, με μεγάλη ακρίβεια και υποστηρίζει και GSM. Αξίζει επίσης να σημειωθεί ότι ένα μεγάλο πλεονέκτημα του συγκεκριμένου GPS, πέρα από τα παραπάνω κριτήρια, είναι και το ότι λαμβάνει σήμα από 22 δορυφόρους. Η δομή και οι λειτουργίες του συγκεκριμένου module θα εξεταστούν παρακάτω.

#### 2.4.10 Επιλογή κατάλληλης πλακέτας GSM/GPRS

Στο σημείο αυτό, αφού έχουμε επιλέξει κατάλληλο μικροελεγκτή και κατάλληλο GPS με βάση τα κριτήρια που αναφέρθηκαν, έπρεπε να γίνει επιλογή κατάλληλης πλακέτας GSM/GPRS. Η επιλογή αυτή θα καθοριστεί με τρία πολύ βασικά κριτήρια. Το ένα είναι προφανώς το κόστος που επιθυμούμε να είναι χαμηλό, το δεύτερο είναι η παροχή ισχύος που απαιτεί η συγκεκριμένη πλακέτα και το τελευταίο είναι το εύρος συχνότητας, καθώς πρέπει να επιλέξουμε πλακέτα που να λειτουργεί στις συχνότητες που υπάρχουν στην Ελλάδα. Επίσης προσέξαμε, προφανώς, οι πλακέτες αυτές να είναι συμβατές με Arduino.

Αρχικά, κάνοντας μια έρευνα καταλήξαμε στο GSM/GPRS shield της SPARKFUN, το μοντέλο SM5100B-D. Το κόστος της συγκεκριμένης πλακέτας ανέρχεται στα 35 ευρώ, στοιχείο που το καθιστά ιδιαίτερα οικονομικό και επιπλέον συνδέεται απευθείας πάνω στο Arduino. Ξεκινώντας, λοιπόν να εργαζόμαστε με αυτή την πλακέτα δημιουργήθηκαν κάποια προβλήματα. Πρώτα απ' όλα έπρεπε να δώσουμε στην πλακέτα την πληροφορία της συχνότητας της Ελλάδας, το οποίο μας πήρε αρκετό χρόνο. Έπειτα ενώ ήταν όλα έτοιμα, η κάρτα SIM που εισάγαμε δεν μπορούσε να συνδεθεί στο δίκτυο και να επικοινωνήσει. Τότε, αναρωτηθήκαμε μήπως το πρόβλημα έχει να κάνει με την κάρτα SIM που χρησιμοποιούμε. Έπειτα απο επικοινωνία με την ίδια τη Sparkfun καταλήξαμε ότι η κοινή κάρτα SIM που χρησιμοποιούμε δεν είναι συμβατή με την συσκευή μας, καθώς η συγκεκριμένη πλακέτα απαιτεί κάρτα SIM 2G και παράλληλα να είναι και Machine to Machine (M2M). [63]

#### Τι είναι το Machine to Machine

Το machine to machine αναφέρεται στην απευθείας επικοινωνία μεταξύ συσκευών χρησιμοποιώντας οποιοδήποτε κανάλι επικοινωνίας, συμπεριλαμβανομένων των ενσύρματων και ασύρματων. Η επικοινωνία μηχανής με μηχανή μπορεί να περιλαμβάνει βιομηχανικά όργανα, επιτρέποντας σε έναν αισθητήρα ή έναν μετρητή να μεταδίδει τα δεδομένα που καταγράφει (όπως θερμοκρασία, επίπεδο αποθέματος κ.λπ.) στο λογισμικό εφαρμογών που μπορεί να το χρησιμοποιήσει. Αυτή η επικοινωνία ολοκληρώθηκε αρχικά με το να έχει ένα απομακρυσμένο δίκτυο μηχανών πληροφοριών ρελέ πίσω σε έναν κεντρικό κόμβο για ανάλυση, το οποίο στη συνέχεια θα μετατοπίζεται σε ένα σύστημα όπως ένας προσωπικός υπολογιστής.

Η πιο πρόσφατη επικοινωνία μηχανής προς μηχανή έχει μετατραπεί σε ένα σύστημα δικτύων που μεταδίδει δεδομένα σε προσωπικές συσκευές. Η επέκταση των δικτύων IP σε όλο τον κόσμο έχει κάνει την επικοινωνία μηχανής με μηχανή ταχύτερη και ευκολότερη, ενώ χρησιμοποιεί λιγότερη ενέργεια. [64]

Εισάγοντας, λοιπόν, την καινούρια κάρτα SIM που αγοράσαμε, εκτελέσαμε την ίδια διαδικασία με πριν προκειμένου να συνδεθεί με το δίκτυο. Μάταιη, όμως, αποδείχτηκε και αυτή η προσπάθεια μας, καθώς και πάλι δεν γινόταν. Επικοινωνώντας ξανά με την Sparkfun μας είπαν ότι το μοντέλο της συγκεκριμένης πλακέτας απαιτεί και εξωτερική παροχή ρεύματος ίσης με 2.5A (στοιχείο που δεν



αναφερόταν πουθενά στο αντίστοιχο datasheet), ποσό ιδιαίτερα μεγάλο, πράγμα το οποίο δεν μας εξυπηρετεί και πολύ. Κάναμε, όμως, και μια τελευταία προσπάθεια τροφοδοτώντας την πλακέτα μας με εξωτερική τροφοδοσία με μπαταρία 9V που απέδιδε ρεύμα έως και 3A, 2 από τα οποία θα τα τραβούσε η πλακέτα. Και πάλι το αποτέλεσμα ήταν το ίδιο, η κάρτα δεν μπορούσε να μπει στο δίκτυο. Σε μια τελευταία απικοινωνία με την Sparkfun καταλήξαμε στο πιο ήταν το βασικό πρόβλημα με το συγκεκριμένο μοντέλο και επομένως έπρεπε να το απορίψουμε και να αναζητήσουμε άλλο. Η συγκεκριμένη πλακέτα λειτουργεί αυστηρά με κάρτα SIM 2G και όχι με 3G, παρότι η Vodafone μας είχε διαβεβαιώσει ότι αφού η κάρτα είναι 3G υποστηρίζει αυτόματα και το δίκτυο 2G.

Κάνοντας μια έρευνα αγοράς με βάση τα κριτήρια που θέσαμε παραπάνω εντοπίσαμε στην αγορά τα παρακάτω GSM/GPRS shields τα οποία λειτουργούν αποτελεσματικά στον Ελλαδικό χώρο. Και όταν αναφέρουμε τον όρο αποτελεσματικότητα στον Ελλαδικό χώρο, εννοούμε ότι υποστηρίζουν τις συχνότητες της κινητής τηλεφωνίας της Ελλάδος που κειμούνται στα 800-2600 MHz.

GSM/GPRS shields	Κόστος (\$)	Εξωτερική Παροχή Ισχύος (πέρα από την τροφοδοσία που έχει το Arduino)
Arduino GSM shield 2	60	No
Sparkfun GSM/GPRS SM5100B-D	29.95	2.5A
Keystudio SIM900 GSM/GPRS Module	25	9V, 2A
SIM900 Quad-band GSM/GPRS Shield	24.99	9V, 2A
SIMCOM SIM900 Quad-band GSM GPRS Shield	24.66	No
ITEAD SIM5216E 3G Module WCDMA	76.13	No
Adafruit FONA 800 Shield - Voice/Data Cellular GSM	39.95	No
SIM900 850/900/1800/1900 MHz GPRS/GSM Development Board Module Kit For Arduino	32.26	9V, 2A
SIM800H GPRS IOT Shield For Arduino	29.00	6-12V, >500mA

[65-72]

Έτσι από τον παραπάνω πίνακα θεωρήσαμε ότι η καταλληλότερη επιλογή είναι η πλακέτα Arduino GSM shield 2. Ο βασικότερος λόγος είναι ότι δεν απαιτεί εξωτερική παροχή ισχύος, γεγονός πολύ σημαντικό, μιας και το ενσωματωμένο σύστημα που υλοποιούμε θα χρησιμοποιηθεί σε μέσα μεταφοράς που εκτελούν μεγάλα ταξίδια και συνεπώς χρειαζόμαστε μικρή παροχή, ώστε να διατηρείται η λειτουργία τους καθόλη τη διάρκεια του ταξιδιού. Ένας άλλος εξίσου σημαντικός λόγος που επιλέξαμε τη συγκεκριμένη πλακέτα είναι ότι μετά από εκτενή έρευνα αγοράς και επικοινωνίας με τις εταιρίες που φτιάχνουν αυτές τις πλακέτες καταλήξαμε στο ότι αυτή είναι η πιο απλή, εύχρηστη και αυτόνομη επιλογή, καθώς οι περισσότερες από τις παραπάνω πλακέτες του πίνακα απαιτούν την ύπαρξη ενός χρήστη ο οποίος θα εκτελεί κάποιες συγκεκριμένες εντολές (AT+command), κάθε φορά, στο περιβάλλον του Arduino ώστε να λαμβάνουμε το σήμα τοποθεσίας του. Αντιθέτως, η Arduino GSM shield 2 είναι αυξημένης ευφυίας και απολύτως αυτόνομη, καθώς μπορεί να μας στέλνει τα σήματα τοποθεσίας κάθε στιγμή χωρίς την παρουσία κάποιου χρήστη.

## 2.5 Arduino UNO

Το Arduino είναι ένας μικροελεγκτής μονής πλακέτας, δηλαδή μια απλή μητρική πλακέτα ανοικτού κώδικα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring (ουσιαστικά πρόκειται για τη γλώσσα προγραμματισμού C++ και ένα σύνολο από βιβλιοθήκες, υλοποιημένες επίσης στην C++ ). Το Arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε Processing, Max/MSP, Pure Data, SuperCollider.



Η πλακέτα αυτή χρησιμοποιεί το μικροελεγκτή ATmega328 της Atmel, ο οποίος έχει τα εξής χαρακτηριστικά:

- 32KB flash program memory
- SRAM 2KB data memory
- EEPROM 1KB data memory
- I/O pins: 23
- clock speed: 16 MHz
- 8-bit αρχιτεκτονική

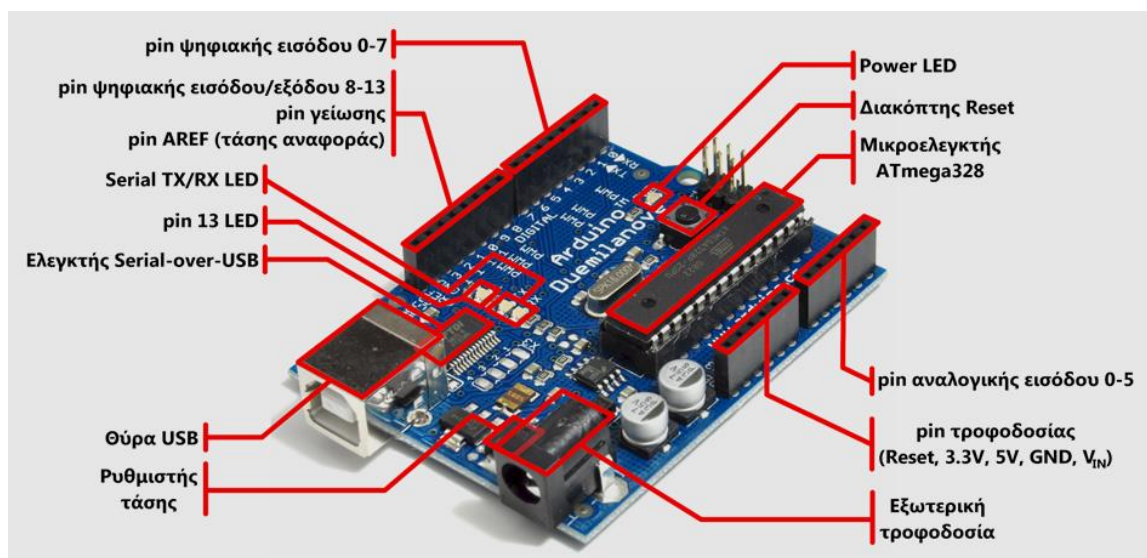
Η σχετική πλακέτα έχει:

- 14 digital I/O pins
- 6 analog input pins
- operating voltage: 5V
- DC current per I/O pin: 40mA
- DC current: 46.5mA

Το περιβάλλον ανάπτυξης που χρησιμοποιείται για τον προγραμματισμό της πλακέτας αυτής είναι το Arduino IDE.

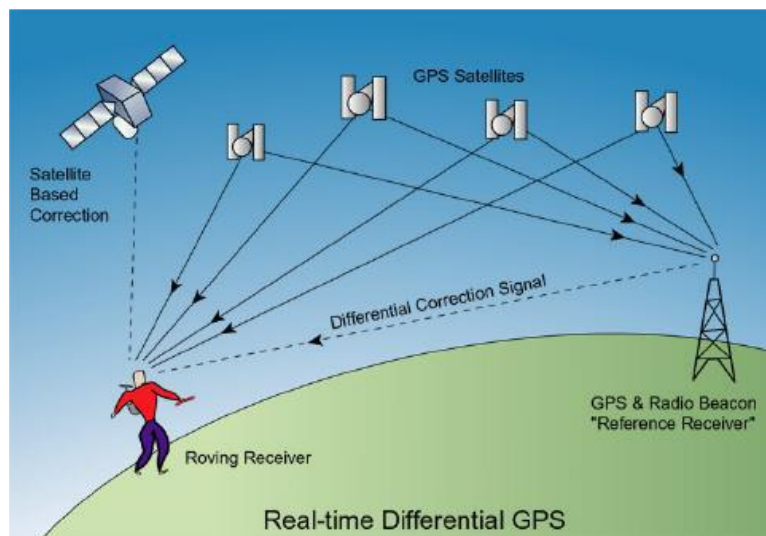
Το Arduino βέβαια, δεν είναι ούτε ο μοναδικός, ούτε και ο καλύτερος δυνατός τρόπος για την δημιουργία μιας οποιασδήποτε διαδραστικής ηλεκτρονικής συσκευής. Όμως το κύριο πλεονέκτημά του είναι η τεράστια κοινότητα που το υποστηρίζει και η οποία έχει δημιουργήσει, συντηρεί και επεκτείνει μια ανάλογοι μεγέθους online γνωσιακή βάση. [9]

Παρακάτω θέτουμε μία φωτογραφία του Arduino που εξηγεί τις λειτουργίες όλων των θυρών του:



## 2.6 GPS

Το GPS (Global Positioning System) Παγκόσμιο Σύστημα Στιγματοθέτησης ή Θεσιθεσίας είναι ένα παγκόσμιο σύστημα εντοπισμού θέσης το οποίο βασίζεται σε ένα πλέγμα 24 δορυφόρων της Γης στους οποίους υπάρχουν ειδικές συσκευές που ονομάζονται «πομποδέκτες GPS». Οι πομποδέκτες παρέχουν ακριβείς πληροφορίες για τη θέση ενός σημείου, το υψόμετρο, την ταχύτητα και την κατεύθυνση της κίνησης όπου με τη βοήθεια λογισμικού χαρτογράφησης (Google Maps) μπορούν να απεικονίσουν γραφικά τις πληροφορίες. Το GPS σχηματίζει ένα παγκόσμιο δίκτυο με εμβέλεια που καλύπτει ξηρά, θάλασσα και αέρα, εξαιτίας αυτής της έκτασης χωρίζεται σε 3 τμήματα: α) το διαστημικό τμήμα, β) το επίγειο τμήμα και γ) το τμήμα των χρηστών.



*Σχηματίζει ένα παγκόσμιο δίκτυο με εμβέλεια που καλύπτει ξηρά, θάλασσα και αέρα, εξαιτίας αυτής της έκτασης χωρίζεται σε 3 τμήματα: α) το διαστημικό τμήμα, β) το επίγειο τμήμα και γ) το τμήμα των χρηστών.*

Το διαστημικό τμήμα αποτελείται από το πλέγμα των δορυφόρων που αναφέραμε παραπάνω. Όπως είναι αναμενόμενο οι δορυφόροι είναι πολύ πιθανό να αντιμετωπίσουν ανά πάσα στιγμή προβλήματα στη λειτουργία τους. Για να διασφαλιστεί η ομαλή λειτουργία των δορυφόρων υπάρχουν τα τμήματα επίγειου ελέγχου. Τέλος το τμήμα του τελικού χρήστη απαρτίζεται από χιλιάδες χρήστες ανά την υφήλιο. [13]

### 2.6.1 Geolocation

Ο όρος Geolocation είναι ο προσδιορισμός της γεωγραφικής θέσης σε πραγματικό χρόνο ενός αντικειμένου όπως μιας πηγής ραντάρ, κινητό τηλέφωνο, υπολογιστής ή οποιαδήποτε άλλη συσκευή μπορεί να συνδεθεί στον διαδίκτυο. Το Geolocation είναι στενά συνδεδεμένο με το σύστημα εντοπισμού θέσης, και μας παρέχει τις συντεταγμένες που λαμβάνει. [14]

### 2.6.2 Geocoding

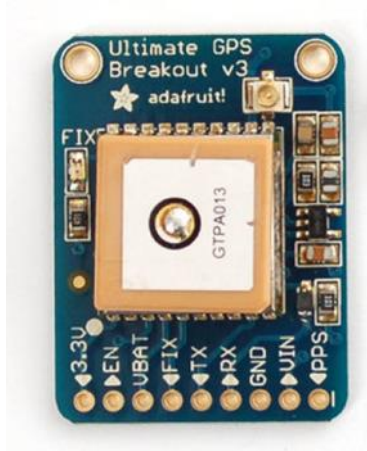
Ο όρος Geocoding ( γεωκωδικοποίηση ) είναι η μετατροπή πληροφορίας σε συντεταγμένες σε κάποιο γεωγραφικό σύστημα αναφοράς. Μια απλή μέθοδος γεωκωδικοποίησης είναι η παρεμβολή διευθύνσεων η οποία χρησιμοποιεί δεδομένα από τα γεωγραφικά συστήματα πληροφοριών στα οποία το οδικό δίκτυο αλλά και σημεία όπως μουσεία , αξιοθέατα και άλλα είναι αντιστοιχισμένα σε γεωγραφικό χώρο συντεταγμένων. Ο geocoder λαμβάνει ως είσοδο μια διεύθυνση, εντοπίζει το τμήμα του δρόμου στον οποίο ανήκει η διεύθυνση και παρεμβάλει την θέση της.

Σήμερα υπάρχουν αρκετές δωρεάν διαθέσιμες υπηρεσίες γεωκωδικοποίησης, όπως Google Maps, Bing Maps, OpenStreetMaps και άλλες.

Η αντίστροφη γεωκωδικοποίηση είναι η διαδικασία εμπλουτισμού των γεωγραφικών συντεταγμένων με μια περιγραφή της θέσης, πιο συγκεκριμένα μια ταχυδρομική διεύθυνση ή όνομα πόλης. [15]

### 2.6.3 Ultimate GPS της Adafruit

Κάνοντας μια καλή έρευνα αγοράς καταλήξαμε στην επιλογή του Ultimate GPS της Adafruit. Η χαμηλή τιμή του, σε συνδυασμό με την επάρκεια της τεχνολογίας που αυτό ενσωματώνει, κρίναμε ότι ήταν το καλύτερο μεταξύ των επιλογών μας για την εργασία αυτή.



Παρακάτω παρατίθενται ορισμένα χαρακτηριστικά-κλειδιά του εν λόγω module:

- -165 dBm ευαισθησία, 10 Hz ενημερώσεις, 66 κανάλια
- 5V σχεδιασμό τροφοδοσίας και μόνο 20mA τρέχουσα ροή ρεύματος
- Εύκολα προσβάσιμο σε Breadboard και δύο τρύπες τοποθέτησης
- Συμβατό με μπαταρία RTC
- Ενσωματωμένη καταγραφή δεδομένων
- Έξοδος PPS στο fix

- > 25 χιλιόμετρα υψόμετρο
- Εσωτερική κεραία ετικέτας + σύνδεσμος u.FL για εξωτερική ενεργή κεραία
- Ρύθμιση LED κατάστασης

Η πλακέτα αυτή είναι χτισμένη γύρω από το chipset MTK3339, ένα υψηλής ποιότητας μονάδα GPS που μπορεί να εντοπίσει μέχρι 22 δορυφόρους σε 66 κανάλια, διαθέτει έναν εξαιρετικό δέκτη υψηλής ευαισθησίας (-165 dB ανίχνευση) και μια ενσωματωμένη κεραία. Μπορεί να κάνει μέχρι και 10 ενημερώσεις τοποθεσίας το δευτερόλεπτο για υψηλή ταχύτητα και διαθέτει υψηλή ευαισθησία καταγραφής και παρακολούθησης. Η κατανάλωση ενέργειας είναι εξαιρετικά χαμηλή, μόνο 20 mA κατά τη διάρκεια της πλοήγησης, γεγονός πολύ θετικό για την κατασκευή μας. Επιπλέον η συσκευή αυτή διαθέτει ένα εξαιρετικά χαμηλό dropout 3.3V, έναν ρυθμιστή έτσι ώστε να μπορούμε να τροφοδοτήσετε με 3.3 – 5 VDC, είσοδο 5V επιπέδου ασφαλείας, ENABLE pin έτσι ώστε να μπορούμε να απενεργοποιήσουμε τη μονάδα χρησιμοποιώντας οποιονδήποτε ακροδέκτη ή διακόπτη του μικροελεγκτή, ένα αποτύπωμα για το προαιρετικό κελί κερμάτων CR1220 για να διατηρήσουμε το RTC σε λειτουργία και να επιτρέψουμε θερμές εκκινήσεις και ένα μικροσκοπικό έντονο κόκκινο LED. Η λυχνία LED αναβοσβήνει περίπου με συχνότητα 1Hz καθώς ψάχνει για δορυφόρους και αναβοσβήνει μια φορά στα 15 δευτερόλεπτα, όταν βρεθεί δορυφόρος για να εξοικονομήσει ενέργεια. Εάν θέλουμε να έχουμε συνεχώς μια λυχνία LED, παρέχουμε επίσης το σήμα FIX σε ένα pin και έτσι μπορούμε να βάλουμε μια εξωτερική λυχνία LED. Δύο χαρακτηριστικά που ξεχωρίζουν πραγματικά για την έκδοση MTK3339 είναι η εξωτερική λειτουργική κεραία και η ενσωματωμένη δυνατότητα καταγραφής δεδομένων. Η συγκεκριμένη πλακέτα διαθέτει μια πρότυπη κεραμική κεραία που της δίνει ευαισθησία -165 dB, αλλά όταν θέλουμε να έχουμε μεγαλύτερη κεραία, μπορείτε να προσαρμόσουμε οποιαδήποτε 3V ενεργή κεραία GPS μέσω της υποδοχής uFL. Το module αυτό ανιχνεύει αμέσως την κεραία και είναι έτοιμο για λειτουργία. Το άλλο θετικό χαρακτηριστικό της νέας μονάδας MTK3339 είναι η ενσωματωμένη ικανότητα καταγραφής δεδομένων. Δεδομένου ότι υπάρχει ένας μικροελεγκτής μέσα στη μονάδα, με κάποια κενή μνήμη FLASH, το πιο πρόσφατο υλικολογισμικό επιτρέπει πλέον την αποστολή εντολών. Το μόνο πράγμα είναι ότι πρέπει να έχουμε ένα μικροελεγκτή για να στείλει την εντολή "Έναρξη καταγραφής". Ωστόσο, μετά την αποστολή του μηνύματος, ο μικροελεγκτής μπορεί να πάει να «κοιμηθεί» και δεν χρειάζεται να «ξυπνήσει» για να μιλήσει στο GPS για να μειώσει την παροχή ισχύος. Η ώρα, η ημερομηνία, το μήκος, το γεωγραφικό πλάτος και το ύψος καταγράφονται το γρηγορότερο, κάθε 1 δευτερόλεπτο. Η εσωτερική FLASH μπορεί να αποθηκεύσει περίπου 16 ώρες δεδομένων και να προσθέσει αυτόματα δεδομένα, ώστε να μην χρειάζεται να ανησυχούμε για τυχαία απώλεια δεδομένων σε περίπτωση απώλειας ισχύος. [16-26]

#### 2.6.4 Σχετικές Εφαρμογές GPS

Παρακάτω θα γίνει μια ενδεικτική αναφορά σε μερικές εφαρμογές που χρησιμοποιούν το δέκτη GPS του κινητού και έχουν σκοπό την παρακολούθηση απομακρυσμένου χρήστη.

- *Real-Time GPS Tracker*, είναι μια εφαρμογή με την οποία βλέπουμε την ακριβή τοποθεσία της οικογένεια μας ή των φίλων μας σε χάρτες Google Maps, και είναι κάτι παραπάνω από ένα GPS Tracker αφού παρέχει ζωντανή παρακολούθηση. Με τη συγκεκριμένη εφαρμογή έχετε τη δυνατότητα να παρακολουθείτε την οικογένεια, τους φίλους, με την προϋπόθεση να έχει εγκαταστήσει και το άτομο που παρακολουθείται την εφαρμογή στο κινητό του. Τέλος η εφαρμογή διατίθεται δωρεάν. [27]
- *Family Locator –GPS Tracker* είναι άλλη μια εφαρμογή με την οποία βλέπουμε την ακριβή τοποθεσία της οικογένεια μας ή των φίλων μας. Επίσης με την συγκεκριμένη εφαρμογή υπάρχει η δυνατότητα δωρεάν επικοινωνίας μέσω μηνυμάτων μεταξύ των χρηστών της εφαρμογής, και επιπλέον υπάρχουν χρήσιμες δυνατότητες. Τέλος η εφαρμογή με κάποιες βασικές λειτουργίες διατίθεται δωρεάν ενώ απαιτεί κάποια συνδρομή για επιπλέον δυνατότητες. [28]
- *Sprint Family Locator* είναι άλλη μια εφαρμογή με την οποία βλέπουμε την ακριβή τοποθεσία της οικογένεια μας ή των φίλων μας. Και αυτή είναι μια εφαρμογή παρακολούθησης αγαπημένων προσώπων. Η συγκεκριμένη εφαρμογή είναι δωρεάν για 15 μέρες, και μετά έχει μηνιαία συνδρομή 5,99 δολάρια. [29]
- *Life360* είναι άλλη μια εφαρμογή με την οποία βλέπουμε την ακριβή τοποθεσία της οικογένεια μας ή των φίλων μας με τις ίδιες δυνατότητες με τις παραπάνω εφαρμογές. Το κόστος της εφαρμογής είναι δωρεάν για 30 μέρες και μετά έχει μηνιαία συνδρομή. [30]

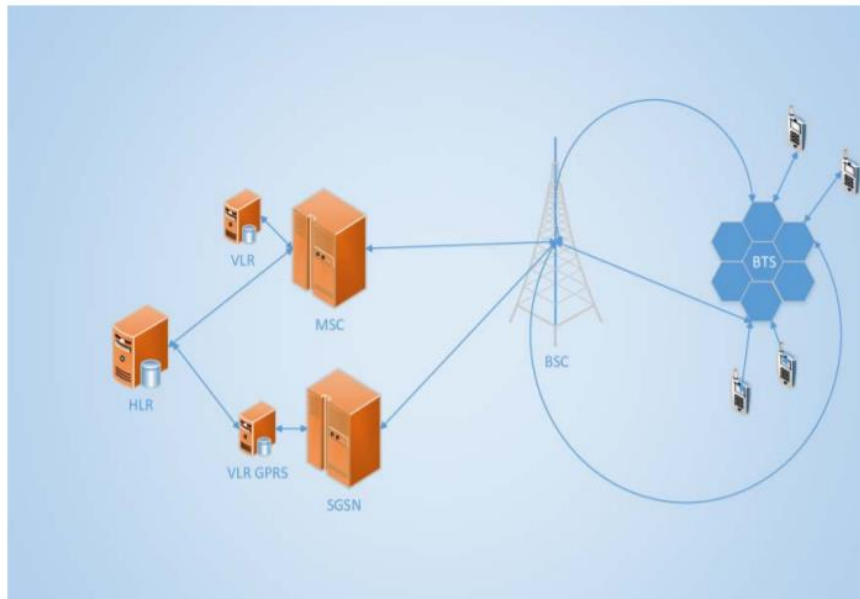
Η συσκευή μας είναι ικανή να χρησιμοποιηθεί ως βάση σε όλες αυτές τις εφαρμογές απόλυτα αποτελεσματικά.

#### 2.7 GSM

Το Global System for Mobile communications (Παγκόσμιο Σύστημα Κινητών Επικοινωνιών), είναι ένα κοινό Ευρωπαϊκό ψηφιακό σύστημα κινητής τηλεφωνίας. Το Ευρωπαϊκό Τηλεπικοινωνιακό Συμβούλιο (European Telecommunications Standards Institute) το 1982, άρχισε την μελέτη για την δημιουργία ενός κοινού Ευρωπαϊκού ψηφιακού συστήματος κινητής τηλεφωνίας δεύτερης γενιάς (2G). Αυτό το σύστημα ονομάστηκε αρχικά Group Special Mobile (GSM).

Το 1989 η ευθύνη του GSM ανατέθηκε στο Ευρωπαϊκό Τηλεπικοινωνιακό Ινστιτούτο Προτύπων (ETSI) και το 1990 ανακοινώθηκαν επίσημα για πρώτη φορά το πρότυπο και τα χαρακτηριστικά του GSM. Το 1991 άρχισε η εμπορική του διάθεση στην Ευρώπη, ενώ στην Ελλάδα το σύστημα χρησιμοποιήθηκε το 1993 από την WIND Hellas (πρώην TIM ή πρώην TELESTET). Το πρότυπο GSM δεν είναι μόνο

Ευρωπαϊκό πρότυπο, αφού υιοθετήθηκε από πολλές άλλες χώρες των άλλων Ηπείρων, εκμεταλλευόμενο διάφορες ζώνες συχνοτήτων. [73]



*Δομή Δικτύου GSM*

### 2.7.1 Επικοινωνία μέσω γραπτών μηνυμάτων και φωνητικής κλήσης

Η τεχνολογία επικοινωνίας μέσω γραπτών μηνυμάτων και φωνητικής κλήσης του κινητού τηλεφώνου με GSM GPS Tracker για εντοπισμό απομακρυσμένου χρήστη, είναι μια τεχνολογία που μας βοηθά στην επίλυση του προβλήματος αυτής της διπλωματικής. Η επιλογή αυτή δεν ήταν τυχαία αλλά προϊόν σκέψης, αναζήτησης της αγοράς και φυσικά του κόστους. Η τεχνολογία που χρησιμοποιήθηκε βασίζεται στη μεταφορά δεδομένων ασύρματα, πράγμα που είναι υποχρεωτικό καθώς ο στόχος μας είναι η παρακολούθηση απομακρυσμένου χρήστη.

Επομένως η επικοινωνία του κινητού τηλεφώνου με την GSM GPS Tracker συσκευή που είναι υπεύθυνη για τον εντοπισμό του απομακρυσμένου χρήστη πραγματοποιήθηκε με φωνητική κλήση και με αποστολή και λήψη γραπτών μηνυμάτων. Βεβαίως ο συγκεκριμένος τρόπος επικοινωνίας έχει ένα αρνητικό, ότι η αποστολή και η λήψη γραπτών μηνυμάτων χρεώνεται. Ευτυχώς στις μέρες μας υπάρχουν αρκετά πακέτα μηνυμάτων από όλες τις εταιρίες κινητής τηλεφωνίας επομένως το κόστος να είναι αρκετά μικρό.

### 2.7.2 Arduino GSM Shield 2

Όπως είπαμε και παραπάνω η κατάλληλότερη πλακέτα GSM είναι η Arduino GSM shield 2, καθώς παρέχει ιδανικό συνδιασμό τιμής, χαμηλής παροχής ισχύος και αυτονομίας. Η πλακέτα αυτή παρέχει και ενσωματωμένη κεραία, συνεπώς δεν απαιτείται αγορά εξωτερικής.



Το Arduino GSM Shield 2 συνδέει το Arduino με το Διαδίκτυο χρησιμοποιώντας το ασύρματο δίκτυο GPRS. Δεν απαιτεί περαιτέρω παροχή ισχύος, πέρα από τα 5V που του παρέχει το ίδιο το Arduino. Απλά συνδέουμε αυτό το module στην πλακέτα Arduino, εισάγουμε μια κάρτα SIM της κινητής τηλεφωνίας Cosmote που προσφέρει κάλυψη GPRS και ακολουθούμε μερικές απλές οδηγίες για να ξεκινήσουμε την λειτουργία του. Με την πλακέτα αυτή μπορούμε επίσης να κάνουμε/λάβουμε φωνητικές κλήσεις (θα χρειαστεί ένα εξωτερικό κύκλωμα ηχείων και μικροφώνου) και να στείλουμε/λάβουμε μηνύματα SMS. [92]



Η πλακέτα περιλαμβάνει μια σειρά LEDs κατάστασης:

On: δείχνει ότι το Shield παίρνει ενέργεια.

Status: ενεργοποιείται όταν τροφοδοτείται το μόντεμ και μεταφέρονται δεδομένα από / προς το δίκτυο GSM / GPRS.

Net: αναβοσβήνει όταν το μόντεμ επικοινωνεί με το ραδιοδίκτυο. [93]

Άλλα χαρακτηριστικά και πλεονεκτήματα του GSM shield 2 είναι:

- Συχνότητες μόντεμ: GSM 850MHz, GSM 900MHz, DCS 1800 MHz, PCS 1900 MHz
- Υποστηρίζει TCP / UDP και HTTP
- Ταχύτητα μεταφοράς Glinks προς τα κάτω και μεταγωγής ανερχόμενης ζεύξης 85,6 kbps
- Προσαρμογέας micro έως nano SIM
- LED κατάσταση
- Υποδοχή ήχου
- Πλήκτρα τροφοδοσίας & επαναφοράς [93]

Οι ψηφιακές ακίδες 2, 3 και 7 προορίζονται για επικοινωνία μεταξύ του Arduino και του μόντεμ. Η επικοινωνία μεταξύ μόντεμ και Arduino γίνεται από τη βιβλιοθήκη λογισμικού Serial στις ακίδες 2 και 3. Η ακίδα 7 χρησιμοποιείται για την επαναφορά του μόντεμ. [94]

### 2.7.3 Σχετικές Εφαρμογές με GPS και GSM

Εκτός βέβαια από τις εφαρμογές που γίνεται παρακολούθηση απομακρυσμένου χρήστη μεταξύ κινητών συσκευών, υπάρχουν στο εμπόριο μεγάλη ποικιλία GPS Tracker τα οποία δέχονται κάρτα SIM και είτε στέλνουν την απομακρυσμένη τοποθεσία μέσω SMS ή μέσω του διαδικτύου. Παρακάτω θα γίνει μια ενδεικτική αναφορά σε τέτοιες συσκευές που έχουν σκοπό την παρακολούθηση απομακρυσμένου χρήστη.

- *Συσκευή παρακολούθησης με κάρτα SIM (κοριός)* είναι μία αθόρυβη συσκευή παρακολούθησης, με ενσωματωμένη τεχνολογία για σύνδεση σε δίκτυο GSM. Διαθέτει σύστημα τεσσάρων μικρόφωνων και την δυνατότητα να δεχτεί οποιαδήποτε κάρτα SIM κινητής τηλεφωνίας 900/1800/1900MHZ. Το μόνο που χρειάζεται είναι μία ενεργή κάρτα SIM. Μπορείτε να καλέσετε την συσκευή η οποία ανοίγει την γραμμή αυτόματα και να ακούτε όλους τους ήχους μέσα στο πεδίο της συσκευής. Λειτουργικά δε διαφέρει από ένα κινητό τηλέφωνο, με τις ιδιαιτερότητες της αυτόματης αποδοχής κλήσης αλλά και της έλλειψης ηχείου, ώστε να μην παράγει κανέναν ήχο. [75]



- *GPS Tracker δορυφορικού εντοπισμού οχημάτων σε πραγματικό χρόνο με δυνατότητα ανανέωσης GPS σήματος απομακρυσμένη επανεκκίνηση και αμφίδρομη επικοινωνία.* Με την συγκεκριμένη μονάδα GPS που διαθέτει η συσκευή λαμβάνει συντεταγμένες μέσω του δορυφορικού δικτύου και με την βοήθεια της μονάδας GSM σας ενημερώνει για το στίγμα του οχήματός σας ή του αγαπημένου σας προσώπου που σας ενδιαφέρει σε γραπτό μήνυμα ώστε να ξέρετε την ακριβή θέση. [76]



Οι παραπάνω αναφορά σε GPS Trackers τα οποία δέχονται κάρτα SIM και είτε στέλνουν την απομακρυσμένη τοποθεσία μέσω SMS ή μέσω του διαδικτύου είναι μόνο ένα δείγμα από τις διαθέσιμες εφαρμογές που υπάρχουν στην αγορά.

Τέλος είναι διαθέσιμες μερικές Android εφαρμογές που παρέχουν απομακρυσμένη παρακολούθηση χρήστη με την βοήθεια συσκευών GPS Tracker με κάρτα SIM. Τρεις τέτοιες εφαρμογές είναι οι παρακάτω.

- *GPS Tracker Car TK SMS* είναι μια εφαρμογή που χειρίζεται ένα GPS Tracker με SMS και επιτρέπει την παρακολούθηση του οχήματος, η της μοτοσυκλέτας ή του σκάφους, Είναι συμβατή με αρκετά μοντέλα κατασκευαστών GPS Tracker με SMS, και παρέχει και επιπλέον δυνατότητες όπως συναγερμός κίνησης, συναγερμός ταχύτητας [78]
- *GPS Tracker by SMS -Pro* είναι άλλη μια εφαρμογή που χειρίζεται ένα GPS Tracker με SMS και επιτρέπει την παρακολούθηση του οχήματος, η της μοτοσυκλέτας ή του σκάφους. Η συγκεκριμένη εφαρμογή δεν παρέχει καμία άλλη λειτουργία παρά μόνο τον εντοπισμό της θέσης απομακρυσμένου χρήστη και την εμφάνιση της τοποθεσίας σε χάρτη. [79]
- *Pocket finder Android* είναι άλλη μια εφαρμογή που στοχεύει στην παρακολούθηση απομακρυσμένου χρήστη με τη βοήθεια συσκευής παρακολούθησης. Οι λειτουργίες που παρέχει η συγκεκριμένη εφαρμογή είναι μάλλον ίδιες με τις λειτουργίες που παρέχει και η δική μας εφαρμογή και λέω μάλλον διότι δεν έχουμε πρόσβαση στην εφαρμογή γιατί είναι επί πληρωμή.

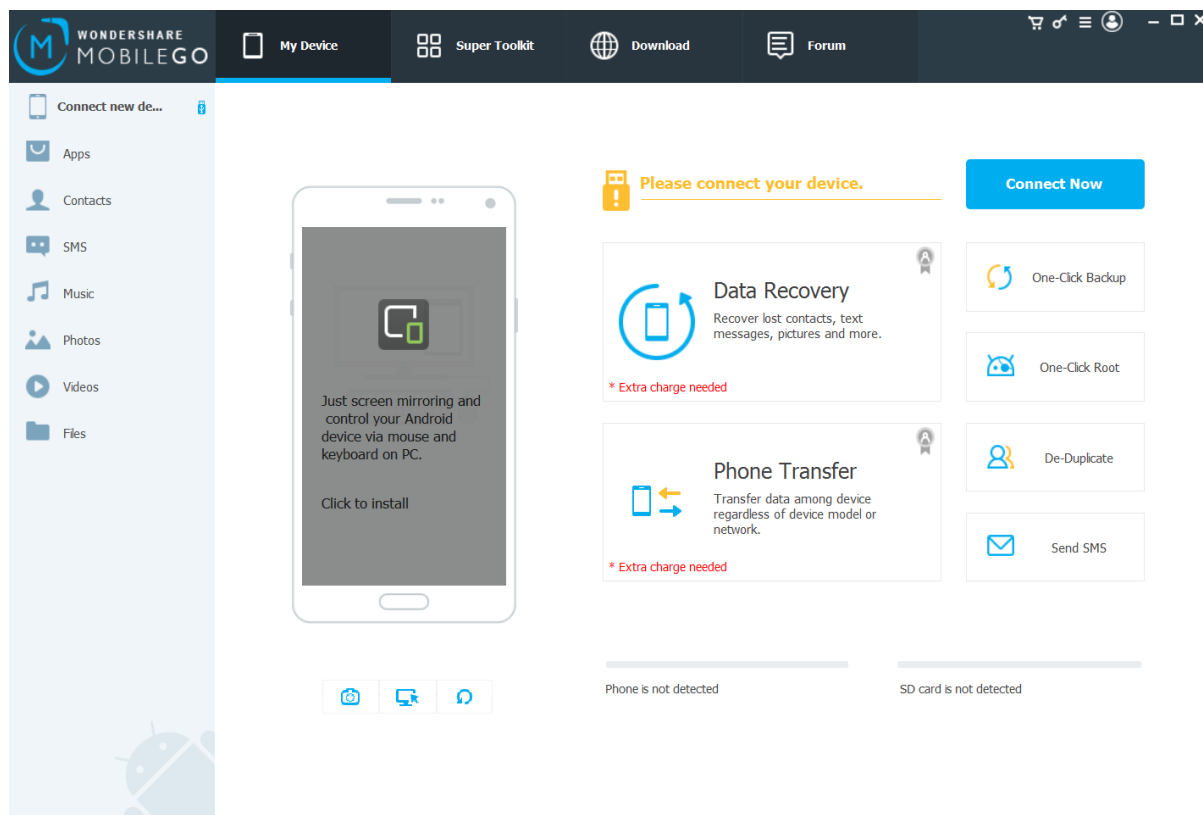
## 2.8 Επικοινωνία Συσκευής με Ηλεκτρονικό Υπολογιστή

Όπως έχουμε ήδη αναφέρει, θέλουμε τα δεδομένα από το GPS, εκτός από κινητό τηλέφωνο, να στέλνονται και σε Ηλεκτρονικό Υπολογιστή. Αναζητώντας, λοιπόν, τον αποτελεσματικότερο και παράλληλα οικονομικότερο τρόπο προκειμένου να επιτευχθεί η επικοινωνία αυτή, επιλέξαμε την μεταφορά SMS από το κινητό στον Ηλεκτρονικό Υπολογιστή με μια επιφάνεια εργασίας του λογισμικού του κινητού που

διαθέτουμε, είτε είναι Android είτε iOS. Με αυτόν τον τρόπο πετύχαμε επικοινωνία μεταξύ συσκευών απόλυτα οικονομική, καθώς δεν απαιτεί καμία έξτρα χρέωση μιας και δεν χρειάζεται καν να αγοράσουμε άλλη συσκευή. Επιπλέον, η διαδικασία σύνδεσης και υλοποίησης είναι ιδιαίτερα απλή και γρήγορη, καθώς το μόνο που χρειάζεται είναι να κατεβάσουμε ένα πολύ απλό και μικρό πρόγραμμα στον υπολογιστή που έχουμε μπροστά μας. Τέλος, όπως θα δούμε και παρακάτω, ο τρόπος αυτός επικοινωνίας μας επιτρέπει και την αποθήκευση των μηνυμάτων που λαμβάνουμε κατευθείαν στον υπολογιστή μας σε αρχεία txt. Περισσότερα στοιχεία θα αναλυθούν στο αμέσως επόμενο υποκεφάλαιο.

### 2.8.1 Πλατφόρμα MobileGo

Στην αρχή εγκαστήσαμε την σωστή έκδοση του MobileGo στον υπολογιστή μας και αφού το τρέξαμε, χρησιμοποιήσαμε ένα καλώδιο USB για να συνδέσουμε το τηλέφωνό μας με τον υπολογιστή. Μετά την σύνδεση, το τηλέφωνό μας (είτε Android είτε iOS) εμφανίστηκε στο κύριο παράθυρο. Αξίζει να σημειωθεί σε αυτό το σημείο ότι θα μπορούσαμε επίσης να συνδέσουμε το τηλέφωνο μας μέσω Wi-Fi, αντί να χρησιμοποιήσουμε καλώδιο. Σε αυτή την περίπτωση όμως θα έπρεπε να βεβαιωθούμε ότι έχουμε εγκαταστήσει το απαραίτητο αρχείο MobileGo.apk στο κινητό μας. Ενδεικτικά παρουσιάζουμε φωτογραφία της πλατφόρμας αυτής:



Παραπάνω φαίνεται η πλατφόρμα χωρίς να έχουμε συνδέσει ακόμα το κινητό μας. Βλέπουμε στην αριστερή στήλη τα διάφορα αρχεία του κινητού στα οποία θα έχουμε πρόσβαση μόλις το συνδέσουμε με τον υπολογιστή. Βλέπουμε, λοιπόν, καθαρά να αναγράφεται και η λέξη sms. Από εκεί θα μπορέσουμε στη συνέχεια να δούμε και να αποθηκεύσουμε τα sms των συντεταγμένων που έρχονται στο κινητό μας, στον υπολογιστή μας. Αυτό θα το εξετάσουμε στο αμέσως επόμενο κεφάλαιο. [80]

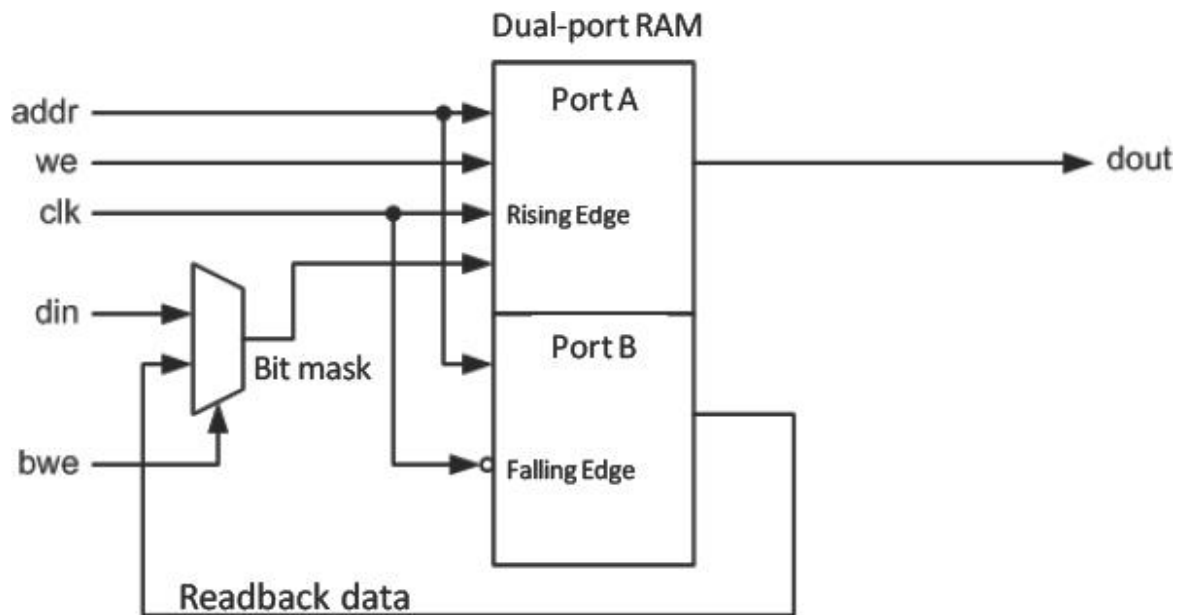
## 2.9 Dual-Port Memory

Dual-Port RAM είναι μνήμη τυχαίας προσπέλασης, στην οποία υπάρχουν 2 ports για ταυτόχρονη πρόσβαση από 2 χρήστες. Στο επίπεδο ολοκληρωμένων κυκλωμάτων, αυτό σημαίνει ότι κάθε κελί μνήμης μπορεί να γίνει accessed ταυτόχρονα από 2 ανεξάρτητα σύνολα από διευθύνσεις, δεδομένα, και γραμμές ελέγχου. Ο λόγος που αναζητήθηκαν πληροφορίες για αυτές τις μνήμες είναι ότι θα μπορούσαν να συνδεθούν σε κάθε συσκευή του δικτύου, και να μπορούν να εκτελούν κώδικα που τους έρχεται μέσω ασύρματης επικοινωνίας. Δηλαδή, να διαβάζουν και να εκτελούν κώδικα από τη μνήμη, ενώ ταυτόχρονα μπορεί να γράφεται στη μνήμη και άλλος κώδικας ο οποίος ήρθε από το ασύρματο. Αναφέρονται παρακάτω μερικές βασικές πληροφορίες για την πληρέστερη κατανόηση της λειτουργίας τους. Με τη χρήση τους, η αποδοτικότητα των προσβάσεων στη μνήμη πρακτικά διπλασιάζεται. Αφού δεν υπάρχουν περιορισμοί του πότε γίνονται οι προσπελάσεις, 2 επεξεργαστές που πχ. έχουν συνδεθεί στο dual-port RAM μπορούν να έχουν μη συγχρονισμένα ρολόγια.

Για να εξασφαλιστεί η εγκυρότητα των δεδομένων στη μνήμη, όταν λειτουργεί με 2 συνδεδεμένους επεξεργαστές, χρειάζεται όταν ο ένας χρησιμοποιεί κάποιο μέρος της μνήμης, τότε αν ο άλλος θέλει να χρησιμοποιήσει αυτό το μέρος, να περιμένει να τελειώσει ο πρώτος. Αυτό γίνεται με κάποια λογική κλειδώματος ή ατομικών εντολών. Ο πρώτος επεξεργαστής θέτει μία μεταβλητή LOCK σε τιμή 1, και χρησιμοποιεί το μέρος της μνήμης. Ο δεύτερος, όταν πάει να κλειδώσει το ίδιο μέρος μνήμης, θα δει ότι είναι ήδη LOCK=1, άρα καταλαβαίνει ότι υπάρχει κάποιος άλλος σε αυτό το μέρος, οπότε και περιμένει να ξεκλειδώσει.

Μία οριακή περίπτωση είναι αν και οι 2 διευθύνσεις που θέλουν οι επεξεργαστές να προσπελάσουν είναι ίδιες, και η προσπέλαση τυχαίνει να γίνει πρακτικά την ίδια χρονική στιγμή. Τότε υπάρχει η πιθανότητα να “χάσουν” και οι 2 στη διεκδίκηση του κλειδώματος, και να περιμένουν ο ένας τον άλλον, και να μη γίνεται τίποτα. Και τα 2 ports είναι σε κατάσταση busy, και το σύστημα δεν προχωράει. Η λύση για τις dual-port RAMs είναι να μην τίθενται ποτέ 2 masters, αλλά 1 master και 1 slave. Το interface που χρησιμοποιείται από τις dual-port RAMs έχει 2 σύνολα των εξής σημάτων (ένα σύνολο για το αριστερό port και ένα για το δεξί): address, data, control.

Στο σημείο αυτό αξίζει να αναφερθούμε στο ότι υπάρχουν δύο ειδών dual-port μνήμες, οι σύγχρονες και οι ασύγχρονες. Οι διαφορές που έχουν έγκειται στο γεγονός ότι οι μιν σύγχρονες έχουν πιο απλό interface από τις ασύγχρονες, ενώ αντίστοιχα οι ασύγχρονες υπερτερούν έναντι των σύγχρονων στο γεγονός ότι έχουν μεγαλύτερες δυνατότητες bandwidth.



### 2.9.1 Προτεινόμενες Dual-Port RAMs

Στο σημείο αυτό κάναμε μια έρευνα αγοράς για τις dual-port RAMs που κυκλοφορούν. Τα κρισιμότερα κριτήρια που αναλύσαμε είναι το μέγεθος της μνήμης, ο τρόπος λειτουργίας (σύγχρονη/ασύγχρονη), ο χρόνος που ένα πρόγραμμα ή συσκευή χρειάζεται για να εντοπίσει μία μονάδα πληροφορίας και να την κάνει διαθέσιμη για επεξεργασία, η παροχή τάσης και προφανώς η τιμή, καθώς όπως προείπαμε αναζητούμε οικονομικές συσκευές.

Τα αποτελέσματα που εντοπίσαμε είναι τα ακόλουθα:

Dual-Port RAM	Μέγεθος Μνήμης	Τεχνολογία	Χρόνος Πρόσβασης	Τιμή (\$)	Παροχή Τάσης (V)
IDT709089/79S/L high-speed 64/32Kx8 synchronous dual-port static RAM	256	Σύγχρονη	15ns	15	5
IDT7028L high speed 64Kx16 dual-port static ram	1MB	Ασύγχρονη	15ns	75	4.5

IDT709099L high-speed 128K x 8 synchronous pipelined dual-port static RAM	1MB	Σύγχρονη	12ns	30	5
IDT7130LA35PDG high speed 1Kx8 dual-port static sram	8KB	Ασύγχρονη	35ns	10	4.5
IDT709099L high-speed 32/16K x 16 synchronous dual-port static RAM	256KB	Σύγχρονη	15ns	57	5
IDT7130LA100PDG	8KB	Ασύγχρονη	100ns	8	4.5
IDT7132LA100PDG	16KB	Ασύγχρονη	100ns	10	4.5

[81-84]

## 2.10 Σχετικές Ενδιαφέρουσες Συσκευές

Στο σημείο αυτό κάναμε μια εκτενή έρευνα αναφοράς για να βρούμε microcontroller-based συσκευές με ιδιαίτερες και ενδιαφέρουσες περιπτώσεις στις οποίες υπάρχει δυνατότητα αποστολής του προγράμματος και των δεδομένων της εκάστοτε συσκευής από εξωτερικό σταθμό, με ασύρματη ή άλλη τηλεπικοινωνιακή ζεύξη.

### 2.10.1 Κυψελοειδές σύστημα GPS για παρακολούθηση οχημάτων

Κατά την αναζήτησή μας, εντοπίσαμε μία πολύ ενδιαφέρουσα διάταξη που έχει αρκετές ομοιότητες με την δική μας συσκευή, όσον αφορά το πρωτόκολλο επικοινωνίας. Πρόκειται για μια διάταξη συναγερμού / εντοπισμού οχήματος που ενσωματώνει ένα συγκρότημα λήψης / μετάδοσης το οποίο λαμβάνει και/ή μεταδίδει κινητά τηλεφωνικά σήματα μέσω ενός κυψελοειδούς τηλεφωνικού δικτύου. Επίσης, ενσωματώνει ένα σύστημα εντοπισμού GPS το οποίο λαμβάνει τρεις ή περισσότερες δορυφορικές μεταδόσεις. Ένας μικροελεγκτής ελέγχει και λαμβάνει δεδομένα τόσο από το συγκρότημα λήψης / μετάδοσης όσο και από την κάρτα δέκτη GPS. Αυτή η εφεύρεση περιλαμβάνει επίσης ένα συγκρότημα εντοπισμού που προσδιορίζει τη θέση του οχήματος και αναφέρει αυτή τη θέση φωνητικά σε οποιαδήποτε γλώσσα ή ψηφιακά όταν ζητηθεί. Όταν είναι απαραίτητο, όπως όταν ενεργοποιείται ένας αισθητήρας συναγερμού, ένα αυτόματο σύστημα κλήσης θα αρχίσει να καλεί αυτόματα έναν ή περισσότερους προ-προγραμματισμένους τηλεφωνικούς αριθμούς μέσω του συγκροτήματος λήψης / μετάδοσης. [86]

### 2.10.2 GPS tracker με χρήση της τεχνολογίας RFID

Μία ακόμα ιδιαίτερη περίπτωση διάταξης που εντοπίσαμε είναι ένα σύστημα παρακολούθησης οχήματος που επιτρέπει στον ιδιοκτήτη ή σε τρίτο να παρακολουθεί τη θέση του οχήματος. Όπως και στη δική μας τη διάταξη, έτσι κι εδώ χρησιμοποιείται GPS το οποίο είναι σε θέση να εντοπίζει την τοποθεσία ανά πάσα στιγμή. Τα δεδομένα που εντοπίζει το συγκεκριμένο GPS της διάταξης αυτής στέλνονται μέσω RFID στους χρήστες των οχημάτων προκειμένου να αποφευχθούν περιπτώσεις συγκρούσεων και κλοπών. Η τεχνολογία RFID είναι μια τεχνολογία που χρησιμοποιεί την επικοινωνία μέσω της χρήσης ραδιοκυμάτων, για τη μεταφορά δεδομένων μεταξύ ενός αναγνώστη και μιας ηλεκτρονικής ετικέτας που είναι προσαρτημένη σε ένα αντικείμενο για σκοπούς ταυτοποίησης και παρακολούθησης. Ουσιαστικά η λειτουργία των συστημάτων RFID βασίζεται στη δυναμική και αμψίδρομη επικοινωνία των ετικετών και των αναγνώστων. Όταν οι ετικέτες RFID βρεθούν στην εμβέλεια της κεραίας του αναγνώστη, η μονάδα ελέγχου επικοινωνεί με ραδιοκύματα με την κεραία των ετικετών RFID. Οι ετικέτες RFID ενεργοποιούνται με τη σειρά τους και επιστρέφουν τα αναζητούμενα δεδομένα στους αναγνώστες. Στη συνέχεια παρεμβαίνει ένα ενδιάμεσο λογισμικό, το οποίο κατανοεί τις πληροφορίες, οι οποίες αποστέλλονται από τη μονάδα ελέγχου του αναγνώστη. Ο αναγνώστης τις μεταφέρει στο εκάστοτε πληροφοριακό σύστημα. [87]

### 2.10.3 Αποστολή δεδομένων τοποθεσίας μέσω Wifi

Ακόμα μία πολύ ενδιαφέρουσα διάταξη που εντοπίσαμε και βασίζεται αυτή τη φορά σε παρακολούθηση ατόμων είναι μία συσκευή που θα αφορά καταγραφή ζωής και θα ανιχνεύει την τοποθεσία στο εσωτερικό κτιρίων. Το GPS είναι ένας τρόπος καταγραφής, αλλά δεν μπορεί να εντοπίσει κινήσεις σε εσωτερικούς χώρους και, ως εκ τούτου, δεν μπορεί να παρακολουθήσει τις συνήθειες δραστηριότητες των ανθρώπων. Σε αυτή όμως τη διάταξη, προτείνεται μια τεχνολογία ανίχνευσης θέσης με βάση την τεχνολογία WiFi για την καταγραφή της θέσης. Ανιχνεύει τη θέση μιας συσκευής από τα εισερχόμενα σήματα WiFi. Αυτό είναι απαραίτητο για την καταγραφή των ιστορικών τοποθεσίας επειδή οι περισσότερες καθημερινές δραστηριότητες βρίσκονται σε εσωτερικά περιβάλλοντα. Η υποκείμενη πλατφόρμα τοποθεσίας WiFi, που ονομάζεται PlaceEngine, παρέχει μια υπηρεσία εκτίμησης θέσης χρησιμοποιώντας μια βάση δεδομένων που αποτελείται από περισσότερα από μισό εκατομμύριο εκτιμώμενες θέσεις σημείου πρόσβασης. Η συσκευή αυτή, η LifeTag είναι μια συσκευή που μοιάζει με keychain που ανιχνεύει και καταγράφει αυτόματα ακόμα και πληροφορίες δακτυλικών αποτυπωμάτων WiFi γύρω από ένα χρήστη. Χρησιμοποιεί επίσης βασικά είδη αισθητήρων, όπως αισθητήρες κίνησης ή προσανατολισμού, για να βοηθήσουν στην καταγραφή της ζωής. [88]

### 2.10.4 Αποστολή δεδομένων μέσω Ethernet (LAN)

Ένα ακόμα ενδιαφέρον σενάριο που υποστηρίζεται από μία διάταξη που εντοπίσαμε κατά τη διάρκεια της έρευνας που κάναμε, το οποίο είναι αρκετά παρεμφερές, αλλά αλλάζει ο τρόπος με τον οποίο στέλνουμε τα δεδομένα είναι το ακόλουθο. Επί του παρόντος, ένα πολύ σημαντικό θέμα σε όλο τον κόσμο είναι η παρακολούθηση των



παραβιάσεων της κυκλοφορίας και των ατυχημάτων που προκύπτουν, ιδίως στον τομέα των σιδηροδρόμων. Θεωρείται ότι αυτό το ζήτημα θα είναι υπό έλεγχο εάν κάποιος θα μπορούσε να παρακολουθήσει σωστά τα επιμέρους τρένα. Αλλά αυτό δεν είναι δυνατό από τα ίδια τα άτομα και έτσι η παρακολούθηση των αμαξοστοιχιών χρησιμοποιώντας μια ειδική συσκευή ενσωματωμένη στο τρένο φαίνεται να είναι μια πιθανότητα. Αυτή η ενσωματωμένη συσκευή προειδοποιεί το μηχανοδηγό να οδηγεί με ασφάλεια και επίσης να ενημερώνει περιοδικά την κατάσταση της θέσης του σε ένα τηλεχειριστήριο. Η συσκευή, λοιπόν, αυτή αποτελεί ένα σύστημα παρακολούθησης αμαξοστοιχιών σε πραγματικό χρόνο χρησιμοποιώντας το παγκόσμιο σύστημα εντοπισμού θέσης (GPS) και την επικοινωνία πληροφοριών μέσω των ιδεών Ethernet. Το σύστημα επιτρέπει την επικοινωνία των πληροφοριών πραγματικού χρόνου σχετικά με τη θέση της αμαξοστοιχίας καθώς και τις υγειονομικές συνθήκες που βασίζονται σε λίγους αισθητήρες που είναι προσαρτημένοι. Ακόμη, το σύστημα παρέχει πλήρεις πληροφορίες σχετικά με τα διάφορα τρένα που τρέχουν, στο τηλεοπτικό τερματικό του ελεγκτή, χρησιμοποιώντας τις διαθέσιμες εγκαταστάσεις διαδικτύου και τις έννοιες χαρτογράφησης της Google. Αυτό το σύστημα χρησιμοποιεί ένα Arduino, όπως ακριβώς και η δική μας διάταξη. Επιπροσθέτως, αυτό το σύστημα παρέχει μια λύση για μια μεγάλη οργάνωση συγκοινωνιών που θα επιτρέπει την παρακολούθηση όλων των κινήσεων αμαξοστοιχιών σε ένα μέρος, χρησιμοποιώντας το διαδίκτυο ή τα διάφορα δίκτυα LAN. Οι θέσεις όλων των επιμέρους οχημάτων χαρτογραφούνται χρησιμοποιώντας τις διευθύνσεις IP τους, στον χάρτη Google, όπου κάθε σημείο στο χάρτη της Google παρουσιάζει τις τρέχουσες πληροφορίες σχετικά με τις αμαξοστοιχίες καθώς και την κατάσταση αλκοολούχων οδηγών, μέγιστη ταχύτητα, συντεταγμένες GPS κλπ. [89]

#### 2.10.5 Συνδυασμένο σύστημα εντοπισμού GPS και σύστημα επικοινωνιών που χρησιμοποιεί κοινόχρηστο κύκλωμα

Κατά την αναζήτηση μας αυτή εντοπίσαμε διάφορες ιδιαίτερες συσκευές που λειτουργούν ως GPS trackers. Μία πολύ ενδιαφέρουσα δουλειά που έχει εκπονηθεί είναι η παραπάνω δουλειά.

Πρόκειται για ένα συνδυασμένο σύστημα GPS και επικοινωνιών με κοινόχρηστο κύκλωμα. Το συνδυασμένο σύστημα περιλαμβάνει μια κεραία για τη λήψη δεδομένων αντιπροσωπευτικών σημάτων GPS, έναν μετατροπέα συχνότητας συνδεδεμένο στην κεραία, έναν συνθετήρα συχνοτήτων συζευγμένο στον μετατροπέα συχνότητας, έναν μετατροπέα αναλογικού προς ψηφιακό συνδεδεμένο στον μετατροπέα συχνότητας και έναν επεξεργαστή συνδεδεμένο με τον μετατροπέα συχνότητας. Ο επεξεργαστής επεξεργάζεται τα δεδομένα που είναι αντιπροσωπευτικά των σημάτων GPS για να καθορίσει μια ψευδοαντιμετώπιση με βάση τα δεδομένα που είναι αντιπροσωπευτικά των σημάτων GPS. Ο ενσωματωμένος δέκτης επικοινωνίας περιλαμβάνει ένα κοινόχρηστο στοιχείο το οποίο είναι τουλάχιστον ένα από την κεραία, τον μετατροπέα συχνότητας, τη συσκευή σύνθεσης συχνοτήτων και τον αναλογικό σε ψηφιακό μετατροπέα. Τυπικά, σε ορισμένες υλοποιήσεις ο επεξεργαστής αποδιαμορφώνει επίσης τα σήματα επικοινωνίας που λαμβάνονται

καθώς και ελέγχει τη διαμόρφωση των δεδομένων που πρόκειται να μεταδοθούν ως σήμα επικοινωνίας μέσω ενός συνδέσμου επικοινωνίας. [96]

#### 2.10.6 Παρακολούθηση οχήματος με GPS, αυτόματη μεταφόρτωση και απομακρυσμένη χειραγώγηση

Πρόκειται για μια πολύ ενδιαφέρουσα διάταξη η οποία περιλαμβάνει έναν δέκτη παγκόσμιου δορυφορικού συστήματος πλοήγησης, μία συσκευή αποθήκευσης δεδομένων για την αποθήκευση δεδομένων λήψης παγκόσμιου συστήματος πλοήγησης από τον παγκόσμιο δέκτη δορυφορικού συστήματος πλοήγησης, και έναν προσαρμογέα ασύρματου δικτύου. Ο προσαρμογέας δικτύου είναι ικανός να αποστέλλει, να λαμβάνει και να ερμηνεύει σήματα δεδομένων που μεταδίδονται ασύρματα σε ένα δίκτυο, όπως ένα οικιακό δίκτυο ή / και ασύρματο δίκτυο. Ένας τέτοιος προσαρμογέας ασύρματου δικτύου μπορεί να διαμορφωθεί για να αναζητήσει διαθέσιμα ασύρματα δίκτυα και να στείλει δεδομένα που είναι αποθηκευμένα στη συσκευή αποθήκευσης δεδομένων σε έναν απομακρυσμένο διακομιστή κατά τη σύνδεση με το ασύρματο δίκτυο. Τέτοια δίκτυα μπορεί να είναι είτε μη ασφαλισμένα δίκτυα ή / και δίκτυα προ-προγραμματισμένα στη συσκευή, όπως μέσω διαμόρφωσης πριν από την τοποθέτηση στο όχημα. [97]

#### 2.10.7 GPS με βοήθεια E-Blocks

Το GPS στέλνει μια συμβολοσειρά μηνύματος στον μικροελεγκτή κάθε δευτερόλεπτο, το οποίο περιέχει όλες τις λεπτομέρειες σχετικά με τη θέση, την ταχύτητα, την πορεία πάνω από το έδαφος κλπ. Χρησιμοποιώντας ταυτόχρονα το GPS και το Card E-blocks επιτρέπει την καταγραφή των διαδρομών συμπεριλαμβανομένης της ταχύτητας και του ύψους πάνω από τη στάθμη της θάλασσας. Οι συντεταγμένες του ταξιδιού μπορούν επίσης να σχεδιαστούν χρησιμοποιώντας λογισμικό όπως το Excel και σε συνδυασμό με μια εικόνα χάρτη για να δείξει την ακριβή διαδρομή του ταξιδιού. Το παράδειγμα αρχικοποιεί την κάρτα και στη συνέχεια δημιουργεί ένα αρχείο που ονομάζεται 'log.csv'. Εάν αυτό το αρχείο υπάρχει ήδη στην κάρτα τότε τα δεδομένα θα επισυναφθούν στο τέλος του αρχείου. Στη συνέχεια δημιουργείται μια συμβολοσειρά στον μικροελεγκτή, ο οποίος αποτελείται από τις τιμές που μεταδίδονται από το στοιχείο GPS. Μόλις δημιουργηθεί η συμβολοσειρά, γράφεται στο τέλος του αρχείου της κάρτας. Όλη αυτή η διαδικασία επαναλαμβάνεται κάθε δέκα δευτερόλεπτα. Εάν εντοπιστεί κάποιο σφάλμα, τότε το πρόγραμμα θα επαναφέρεται στην αρχή και θα προσπαθήσει ξανά. Φυσικά το κύκλωμα των E-block είναι αρκετά μεγάλο, οπότε μόλις έχετε το σύστημα σε λειτουργία και στη συνέχεια είναι δυνατόν να χρησιμοποιήσετε τα φύλλα δεδομένων του E-blocks για να συλλέξετε τα κυκλώματα και να δημιουργήσετε τη δική σας προσαρμοσμένη συμπτυκνωμένη σανίδα. [98]

## 2.11 FPGA

Το FPGA ή Field Programmable Gate Array ή συστοιχία επιτόπια προγραμματιζόμενων πυλών είναι τύπος προγραμματιζόμενου ολοκληρωμένου κυκλώματος γενικής χρήσης το οποίο διαθέτει πολύ μεγάλο αριθμό τυποποιημένων πυλών και άλλων ψηφιακών λειτουργιών. Σε ορισμένα από αυτά ενσωματώνονται και αναλογικές λειτουργίες. Κατά τον προγραμματισμό του FPGA, ο οποίος γίνεται πάντοτε ενώ αυτό είναι τοποθετημένο στο τυπωμένο κύκλωμα, ενεργοποιούνται οι επιθυμητές λειτουργίες και διασυνδέονται μεταξύ τους έτσι ώστε το FPGA να συμπεριφέρεται ως ολοκληρωμένο κύκλωμα με συγκεκριμένη λειτουργία.

Ο κώδικας με τον οποίο προγραμματίζεται το FPGA γράφεται σε γλώσσες περιγραφής υλικού (VHDL, AHDL, Verilog). Όμως τα ιδιαίτερα χαρακτηριστικά του FPGA είναι τα εξής:

Το FPGA χάνει τον προγραμματισμό του κάθε φορά που διακόπτεται η τάση τροφοδοσίας του. Επομένως απαιτεί εξωτερικό μικροεπεξεργαστή ή μνήμη με μόνιμη συγκράτηση δεδομένων (non-volatile memory) από τα οποία θα προγραμματίζεται, κάθε φορά που επανέρχεται η τάση τροφοδοσίας. Ο προγραμματισμός του FPGA μπορεί να αλλάζει κάθε φορά που τροποποιείται το λογισμικό του μικροεπεξεργαστή ή τα δεδομένα της μνήμης που το ελέγχει. Δεν υπάρχει όριο στο πόσες φορές μπορεί να επαναπρογραμματιστεί. Η κατανάλωση ισχύος είναι σημαντικά αυξημένη, σε σχέση με τα ASIC.

Έτσι το FPGA είναι ιδιαίτερα κατάλληλο εκεί που οι παράμετροι λειτουργίας πρέπει να αλλάζουν συχνά ή σε μικρές ποσότητες παραγωγής, ενώ το ASIC, λόγω μαζικής παραγωγής, είναι φτηνότερο εκεί που απαιτούνται μεγάλες ποσότητες και η επιθυμητή λειτουργία είναι αυστηρά προκαθορισμένη, χωρίς σφάλματα (το ASIC δεν επαναπρογραμματίζεται).

Βασική δομική μονάδα του FPGA είναι το λογικό μπλοκ, με τη χρήση του οποίου υλοποιούνται οι λογικές συναρτήσεις που εκφράζουν τη λειτουργία ενός ψηφιακού κυκλώματος. Ανάλογα με το μέγεθος του κυκλώματος πολλά λογικά μπλοκ συνδέονται για να υλοποιήσουν το πλήθος των απαραίτητων λογικών συναρτήσεων. [85]

### 2.11.1 Ασφάλεια στο hardware με τη βοήθεια FPGA

Σε αυτό το σημείο κάναμε μια αναφορά στην ασφάλεια που αφορά το Hardware προκειμένου να αποφύγουμε κλοπές δεδομένων από hackers. Παρακάτω παραθέτουμε μία μέθοδο που μπορούμε να πετύχουμε ασφάλεια στη συσκευή μας, με αποτέλεσμα την αποφυγή τέτοιων παρεμβολών.

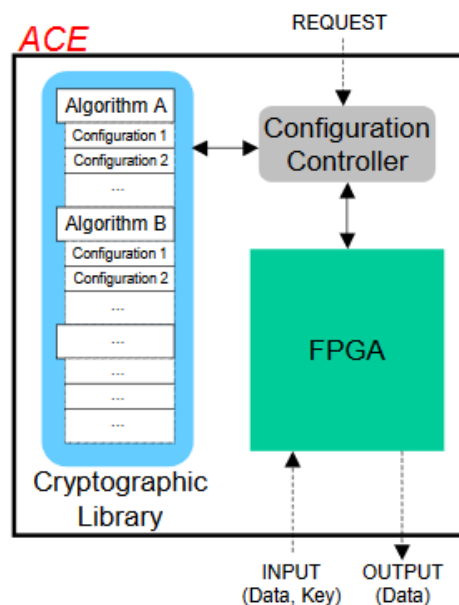
#### *Συσκευή αποκρυπτογράφησης βασισμένη σε FPGA*

Με βάση την τεχνολογία FPGA, προτείνουμε ένα Adaptive Cryptographic Engine (ACE). Το ACE αποτελείται από μια συσκευή FPGA, μια κρυπτογραφική βιβλιοθήκη και έναν ελεγκτή διαμόρφωσης. Το FPGA έχει ρυθμιστεί σε λειτουργία από τον ελεγκτή διαμόρφωσης. Ακολουθώς, γίνεται προσαρμογή στο κλειδί εισόδου και αρχίζει η κρυπτογράφηση / αποκρυπτογράφηση δεδομένων. Ο ελεγκτής

διαμόρφωσης καθορίζει τη διαμόρφωση που θα επιλεγεί με βάση την αιτούμενη ένωση ασφαλείας. Ο πυρήνας του ACE είναι η τεχνολογία FPGA. Οι υλοποιήσεις με βάση το FPGA μπορούν να παρέχουν τις υψηλές ταχύτητες επεξεργασίας που απαιτούνται από τα δίκτυα υψηλής ταχύτητας. Επιπλέον, οι FPGAs μπορούν να αναδιαμορφωθούν εκ νέου, παρέχοντας τον υψηλό βαθμό ευελιξίας που απαιτείται σε δυναμικά μεταβαλλόμενα περιβάλλοντα. Η κρυπτογραφική βιβλιοθήκη αποτελείται από διαμορφώσεις FPGA κρυπτογραφικών αλγορίθμων ιδιωτικού κλειδιού. Κάθε κρυπτογραφικός αλγόριθμος μπορεί να αντιστοιχεί σε διάφορες διαμορφώσεις. Αυτές οι παραλλαγές αλγορίθμων διαφέρουν μεταξύ τους όσον αφορά τον τρόπο κρυπτογραφικής λειτουργίας, το μήκος του κλειδιού, τον αριθμό των γύρων κλπ. Οι διαμορφώσεις αποθηκεύονται στη μνήμη και μπορούν να ενημερωθούν για να επεκταθεί η διάρκεια ζωής του ACE. Ο ελεγκτής διαμόρφωσης είναι απαραίτητος για τη διαμόρφωση ενός FPGA. Εκτός από τη διεπαφή μεταξύ της κρυπτογραφικής βιβλιοθήκης και του FPGA, επιλύει επίσης τα εξωτερικά αιτήματα (π.χ. παράμετροι ασφαλείας). Για τον σχεδιασμό της κρυπτογραφικής βιβλιοθήκης χρησιμοποιείται μια συγκεκριμένη προσέγγιση. Ο τομέας ορίζεται από τον αλγόριθμο και την αρχιτεκτονική FPGA στόχου. Κάθε παραλλαγή του αλγορίθμου αντιστοιχεί σε διαφορετικό υποτομέα. Με βάση συγκεκριμένο τομέα, παράγονται συγκεκριμένες διαμορφώσεις αλγορίθμου. Η κρυπτογραφική βιβλιοθήκη προέρχεται εκτός γραμμής ενώ η διαμόρφωση που θα φορτωθεί στο FPGA συντίθεται εν πτήση.

Βασική ιδέα είναι να δημιουργούμε διαδοχικά συνθέσεις κατά το χρόνο εκτέλεσης με τη συγχώνευση ενός σκελετού με μια διαμόρφωση υποτομέα. Ένας σκελετός είναι η τομή όλων των διαμορφώσεων σε έναν τομέα, δηλαδή μια παραμετροποιημένη διαμόρφωση που αντιστοιχεί στις στοιχειώδεις λειτουργίες ενός αλγορίθμου.

Έτσι, ένας υποτομέας αντιστοιχεί στις τροποποιήσεις του σκελετού που οδηγούν σε μια διαμόρφωση μιας παραλλαγής ενός αλγορίθμου. Ως αποτέλεσμα, οι διαμορφώσεις ενός τομέα μπορούν να εκπροσωπούνται αποτελεσματικά αποφεύγοντας ταυτόχρονα την εφεδρεία δεδομένων, δηλαδή τα δεδομένα διαμόρφωσης του σκελετού. Έτσι στην συσκευή μας μπορούμε να ενσωματώσουμε την συσκευή αυτή προκειμένου να εμποδίσουμε διάφορου είδους «χακαρίσματα» και κλοπές. [90]



## 2.12 AES Algorithm (Advanced Encryption Standard) – Ασφάλεια στο Hardware

Η σημασία της κρυπτογραφίας που εφαρμόζεται στην ασφάλεια στις ηλεκτρονικές συναλλαγές δεδομένων έχει αποκτήσει ουσιαστική σημασία τα τελευταία χρόνια. Καθημερινά πολλοί χρήστες δημιουργούν και ανταλλάσσουν μεγάλες ποσότητες πληροφοριών σε διάφορα πεδία μέσω Internet, τηλεφωνικών συνδιαλέξεων και συναλλαγών ηλεκτρονικού εμπορίου. Αυτά και άλλα παραδείγματα εφαρμογών αξίζουν μια άποψη ασφάλειας, όχι μόνο στη μεταφορά τέτοιων πληροφοριών αλλά και στην αποθήκευσή τους. Αυτό μπορεί να επιτευχθεί με διάφορες τεχνικές όπως ο κωδικός πρόσβασης, η κρυπτογραφία και η βιομετρία. Με αυτή την έννοια, οι τεχνικές της κρυπτογράφησης είναι ιδιαίτερα χρήσιμες. Στην κρυπτογραφία, το AES, επίσης γνωστό ως Rijndael, είναι ένας κρυπτογραφημένος κώδικας, ο οποίος καθορίζει έναν αλγόριθμο κρυπτογράφησης ικανό να προστατεύει τα ευαίσθητα δεδομένα. Ο αλγόριθμος AES είναι ένας συμμετρικός κωδικός μπλοκ που μπορεί να κρυπτογραφήσει και να αποκρυπτογραφήσει τις πληροφορίες. Ο αλγόριθμος AES χρησιμοποιεί κρυπτογραφικά κλειδιά 128, 192 και 256 bits για κρυπτογράφηση και αποκρυπτογράφηση δεδομένων σε μπλοκ των 128 bits. Η υλοποίηση υλικού αυτού του αλγορίθμου μπορεί να προσφέρει είτε υψηλή απόδοση είτε χαμηλό κόστος για συγκεκριμένες εφαρμογές. Στα κυριότερα κανάλια επικοινωνίας ή στους υπερφορτωμένους διακομιστές δεν είναι δυνατή η απώλεια της ταχύτητας επεξεργασίας, η οποία μειώνει την αποδοτικότητα του συνολικού συστήματος κατά την εκτέλεση αλγορίθμων κρυπτογραφίας στο λογισμικό. Από την άλλη πλευρά, ένα χαμηλό κόστος και ένα μικρό σχέδιο μπορούν να χρησιμοποιηθούν σε εφαρμογές έξυπνων καρτών, οι οποίες επιτρέπουν την ασφαλή λειτουργία ενός ευρέος φάσματος εξοπλισμού.

Ο αλγόριθμος AES μπορεί να υλοποιηθεί αποτελεσματικά από υλικό και λογισμικό. Οι εφαρμογές λογισμικού κοστίζουν τους μικρότερους πόρους, αλλά προσφέρουν μια περιορισμένη φυσική ασφάλεια και την πιο αργή διαδικασία. Εκτός αυτού, αυξάνονται οι απαιτήσεις για υψηλής ταχύτητας, υψηλού όγκου ασφαλείς επικοινωνίες σε συνδυασμό με τη φυσική ασφάλεια και την υλοποίηση υλικού της κρυπτογράφησης. Έτσι, αξιοποιώντας τον αλγόριθμο αυτό, μπορούμε να προσθέσουμε ασφάλεια στο Hardware της συσκευής που υλοποιούμε, προκειμένου να αποφευχθούν πιθανές κλοπές δεδομένων από hackers.[91]

## ΚΕΦΑΛΑΙΟ 3 - ΠΡΟΣΕΓΓΙΣΗ

### 3.1 Ελεύθερο Λογισμικό

Σαν πρώτη προσέγγιση για την εκπόνηση της εργασίας, αναζητήσαμε ενσωματωμένο σύστημα με μικροελεγκτή που χρησιμοποιεί ελεύθερο λογισμικό. Η επιλογή μας στηρίζεται στο γεγονός ότι η χρήση ελεύθερου λογισμικού μας παρέχει πολλά πλεονεκτήματα. Ας αναλύσουμε, όμως, αρχικά την έννοια του ελεύθερου λογισμικού και στη συνέχεια θα ασχοληθούμε και με τα πλεονεκτήματά του.

Το Ελεύθερο Λογισμικό/ Λογισμικό Ανοικτού Κώδικα (ΕΛ/ΛΑΚ) είναι το λογισμικό που ο καθένας μπορεί ελεύθερα να χρησιμοποιεί, να αντιγράφει, να διανέμει και να τροποποιεί ανάλογα με τις ανάγκες του. Είναι ένα εναλλακτικό μοντέλο ανάπτυξης και χρήσης λογισμικού που βασίζεται στην ελεύθερη διάθεση του πηγαίου κώδικα, το οποίο παρέχει τη δυνατότητα αλλαγών ή βελτιώσεων ώστε να καλύπτονται οι ανάγκες αυτού που το χρησιμοποιεί.

Τα οφέλη της χρήσης ελεύθερου λογισμικού είναι τα ακόλουθα:

- Απόλυτα νόμιμο λογισμικό το οποίο διατίθεται χωρίς κόστος και έχει πολύ μικρότερο κόστος συντήρησης
- Λογισμικό που ενσωματώνει άμεσα τις πιο σύγχρονες τεχνολογικές εξελίξεις
- Συχνές ενημερώσεις με νέες δυνατότητες
- Μεγάλη και φιλική κοινότητα ανάπτυξης και υποστήριξης του λογισμικού
- Πληθώρα πρόσθετων προγραμμάτων με εύκολη και δωρεάν εγκατάσταση
- Σταθερότητα και ασφάλεια
- Σημαντικά ασφαλέστερο και αξιόπιστο σε σχέση με ιδιόκτητο λογισμικό που κατεβάζουμε από το διαδίκτυο.
- Δυνατότητα να εξερευνήσουμε και να μάθουμε τον τρόπο λειτουργίας του λογισμικού προσαρμόζοντάς το στις ανάγκες μας

Το κόστος άδειας χρήσης των εφαρμογών ελεύθερου λογισμικού είναι τις περισσότερες φορές μηδενικό. Δεν αγοράζονται άδειες χρήσεις και μπορούμε να έχουμε απεριόριστο αριθμό εγκαταστάσεων. Η χρήση ανοικτού κώδικα δεν περιορίζει τον οργανισμό ή τον απλό χρήστη σε μια σχέση εξάρτησης από εταιρίες και επειδή η διανομή, η διόρθωση σφαλμάτων και η ανάπτυξη του λογισμικού ΕΛ/ΛΑΚ μπορεί να γίνει από κάθε τεχνικά καταρτισμένη ομάδα, δημιουργείται ένα περιβάλλον έντονου ανταγωνισμού ο οποίος οδηγεί σε χαμηλές τιμές και υψηλές υπηρεσίες υποστήριξης.

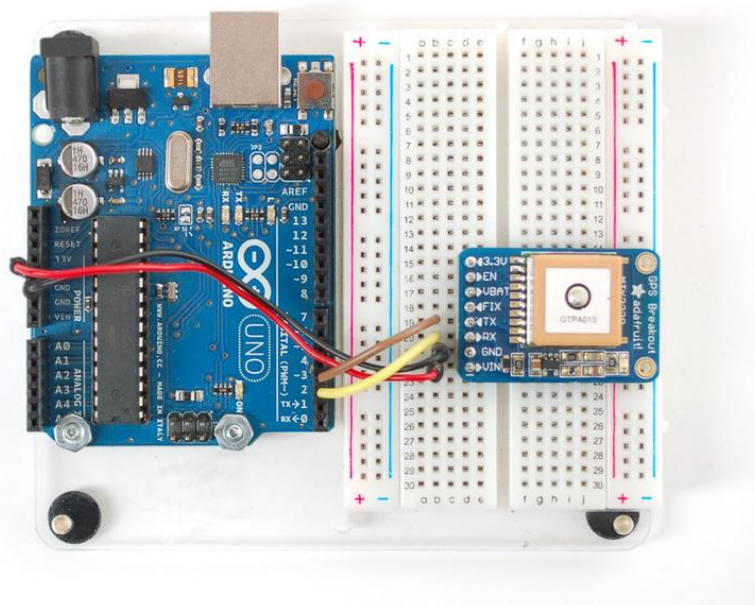
Το ελεύθερο λογισμικό αποτελείται από μια μεγάλη κοινότητα χρηστών και προγραμματιστών, οι οποίοι συνεργάζονται για τη συνεχή βελτίωση του λογισμικού, παρέχοντας γνώσεις και εργασία. Σήμερα λειτουργεί ένα παγκόσμιο ανοικτό δίκτυο προγραμματιστών, οι οποίοι παράλληλα αναπτύσσουν και διορθώνουν τον κώδικα των προγραμμάτων, κυκλοφορώντας ταχύτατα νέες βελτιωμένες εκδόσεις λογισμικού. Με αυτό τον τρόπο συμβάλλουν καθημερινά στην δημιουργία νέων κοινών αγαθών.

Η λογική της ανάπτυξης του λογισμικού του ανοιχτού κώδικα είναι τέτοια έτσι ώστε να είναι δοκιμασμένο από πολλούς και αποφεύγονται αρνητικές εκπλήξεις και σφάλματα. Ο κώδικας μελετάται από πλήθος ανθρώπων, άρα τα όποια κενά ασφαλείας εντοπίζονται και διορθώνονται με μεγάλη ταχύτητα. Η υποστήριξη σε περίπτωση εμφάνισης προβλημάτων μπορεί να προέλθει άμεσα.

Το διαδίκτυο αποτελεί το βασικό τρόπο πρόσβασης στο διαθέσιμο ελεύθερο λογισμικό. Η σχέση του διαδικτύου και ΕΛ/ΛΑΚ, υπήρξε εξ αρχής στενή και αμφίδρομη. Η εξάπλωση του ΕΛ/ΛΑΚ έχει στηριχθεί στην ευρεία χρήση του διαδικτύου και η διαδικασία ανάπτυξης και λειτουργίας του Διαδικτύου βασίζεται, κατά κύριο λόγο, σε ΕΛ/ΛΑΚ. Τέλος και το διαδίκτυο νέας γενιάς αναπτύσσεται με ανοιχτό λογισμικό. Το λογισμικό ανοιχτού κώδικα, με συνεχείς βελτιώσεις και αυξημένη πλέον φιλικότητα προς το χρήστη, κερδίζει διαρκώς νέους φίλους παγκοσμίως. Στην εκπαίδευση, στη δημόσια διοίκηση και στις επιχειρήσεις, οι ενδιαφερόμενοι ενημερώνονται και αποκτούν ιδιαίτερα ελκυστικά εργαλεία, αξιόπιστα, σταθερά στη λειτουργία, και απαλλαγμένα από τα σημαντικά κόστη απόκτησης και συνεχούς αναβάθμισης που απαιτούν τα κλειστά λογισμικά. Έτσι πλέον όλο και πιο πολλοί πόροι διατίθενται στην τεχνική υποστήριξη με σημαντικά οφέλη για την τοπική και εθνική οικονομία. Με βάση όλα τα παραπάνω οφέλη αποφασίσαμε να επιλέξουμε για την εργασία το Arduino που όπως έχουμε πει είναι προϊόν ελεύθερου λογισμικού και συνεπώς είναι οικονομικό και πολύ λειτουργικό. Το Arduino έχει αναλυθεί παραπάνω με τις ιδιότητές του.

### 3.2 Αρχική Σύνδεση του GPS με το Arduino

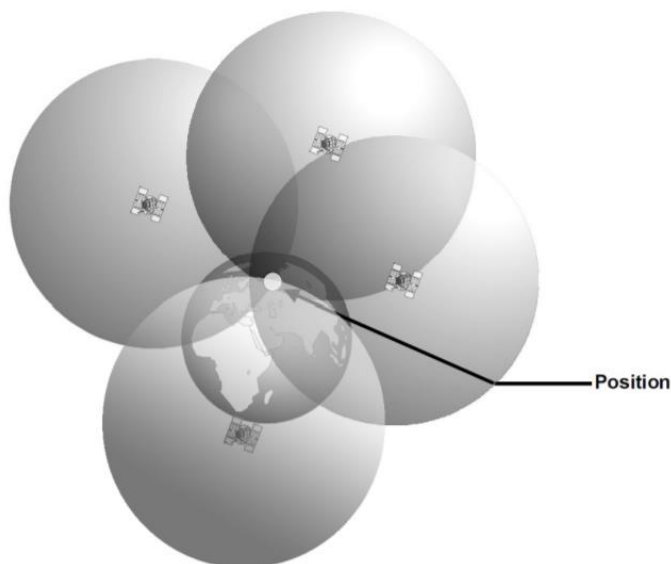
Σε πρώτη φάση εκτελέσαμε σύνδεση του GPS μας με το Arduino Uno με τη βοήθεια καλωδίων. Θέλαμε να ελέγξουμε αν το GPS μας λειτουργεί καλά, ώστε στη συνέχεια να το ενσωματώσουμε στην τελική συσκευή μας. Συνδέοντας, λοιπόν, αρχικά το GPS το βλέπουμε να ψάχνει να βρει δορυφόρους. Οι συνδέσεις που κάναμε μεταξύ Arduino και GPS είναι οι ακόλουθες:



Γράψαμε και τον αντίστοιχο κώδικα, ο οποίος βρίσκεται στο παράρτημα Α στο τέλος. Στη συνέχεια συνδέσαμε το καλώδιο USB και ανοίξαμε τη σειριακή οθόνη από το IDE του Arduino και επιλέξαμε 115200 baud στο μενού. Το αποτέλεσμα που προέκυψε ήταν το GPS να μας εμφανίζει κάθε φορά τις συντεταγμένες του γεωγραφικού μήκους και πλάτους του σημείου που βρισκόμαστε. Επιπλέον, μας εμφάνιζε και τον αριθμό των δορυφόρων από τους οποίους λάμβανε σήμα. Ένα παράδειγμα εκτέλεσης του κώδικα φαίνεται στην παρακάτω εικόνα:

```
Fix: 1 quality: 1
Location: 4043.5717N, 7400.2783W
Speed (knots): 0.39
Angle: 81.93
Altitude: 60.40
Satellites: 7
$PGTOP,11,3*6F
$GPGGA,225358.000,4043.5719,N,07400.2785,W,1,7,1.05,60.4,M,-34.2,M,,*5E
$GPRMC,225358.000,A,4043.5719,N,07400.2785,W,0.34,81.93,081112...A*4A
```

Όπως μπορούμε να διαπιστώσουμε το GPS λειτουργεί σωστά και μας εκτυπώνει στην οθόνη τις συντεταγμένες, την ταχύτητα που έχουμε, την γωνία, το υψόμετρο καθώς και τους δορυφόρους που έπιασε. Να σημειωθεί σε αυτό το σημείο ότι από τις δοκιμές που κάναμε με το GPS, έπιανε από 7 μέχρι και 15 δορυφόρους ανάλογα με τις συνθήκες που επικρατούσαν. Ο αριθμός αυτός είναι πραγματικά πάρα πολύ καλός, καθώς για να αποκτήσουμε στίγμα τοποθεσίας, αρκούν 4 δορυφόροι (ένας για την συντεταγμένη x, ένας για την y, ένας για την z και ένας που τους επαληθεύει).





### 3.3 Αρχική Σύνδεση του GSM με το Arduino

Αφού τσεκάρουμε ότι το GPS μας δουλεύει καλά, σε αυτό το σημείο έπρεπε να ελέγξουμε την λειτουργία της πλακέτας Arduino GSM shield 2. Συνεπώς αυτό που κάναμε ήταν να συνδέσουμε την πλακέτα Arduino GSM shield 2 με το Arduino Uno, να εισάγουμε την κάρτα SIM της Cosmote και να τρέξουμε τον κατάλληλο κώδικα. Να σημειωθεί σε αυτό το σημείο ότι από την κάρτα SIM αφαιρέσαμε το PIN πριν την εισαγωγή στη συσκευή μας, προκειμένου να αποφευχθούν τυχόν προβλήματα ασφαλείας. Η σύνδεση που εκτελέσαμε είναι η παρακάτω:



Εν συνεχεία γράψαμε τον αντίστοιχο κώδικα που φαίνεται στο παράρτημα Α στο τέλος, με τον οποίο θα καταλάβουμε αν το GSM λειτουργεί και μπορεί να συνδεθεί με το δίκτυο και να στείλει SMS. Μόλις τον υλοποιήσαμε, συνδέσαμε το καλώδιο USB και ανοίξαμε τη σειριακή οθόνη από το IDE του Arduino και επιλέξαμε 9200 baud στο μενού. Όπως μπορούμε να δούμε και στον κώδικα, θέλαμε να στείλουμε την λέξη “test” στον αριθμό +306947050972. Πράγματι, με το που εκτελέσαμε τον κώδικα μετά από λίγα δευτερόλεπτα, η λέξη test ήρθε με μήνυμα στο κινητό τηλέφωνο που είχαμε βάλει. Στη συνέχεια κάναμε λίγες ακόμα δοκιμές αυξάνοντας την απόσταση μεταξύ του κινητού τηλεφώνου που στέλναμε τα μηνύματα και του Arduino GSM shield. Και πάλι λαμβάναμε SMS σε κάθε περίπτωση. Συνεπώς, και η πλακέτα GSM λειτουργούσε σωστά και συνεπώς ήμασταν έτοιμοι να προχωρήσουμε στην τελική σύνδεση που αποτελεί και την ολική υλοποίηση της συγκεκριμένης διπλωματικής.

### 3.4 Τελική Σύνδεση

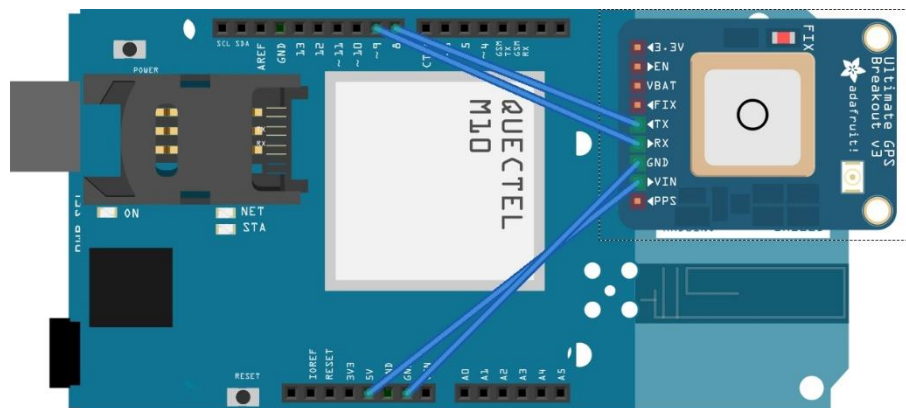
Στο σημείο αυτό είχαμε τσεκάρει την λειτουργία και του GPS, το οποίο ήταν σε θέση να λαμβάνει στίγμα τοποθεσίας κάθε στιγμή και με μεγάλη ακρίβεια καθώς και την λειτουργία του GSM, το οποίο μπορούσε να στέλνει μηνύματα σε κινητά τηλέφωνα,

πετυχαίνοντας μια πολύ αποτελεσματική επικοινωνία, και έπρεπε πλέον να κάνουμε την τελική υλοποίηση. Να στέλνουμε δηλαδή με γραπτό μήνυμα σε κινητό τηλέφωνο, τα σήματα τοποθεσίας που λαμβάνει το GPS.κάθε στιγμή.

Στο σημείο αυτό προέκυψε ένα πολύ σημαντικό πρόβλημα, κατά τη συγχώνευση των δύο κωδίκων (του GPS και του GSM). Το πρόβλημα ήταν ότι ο κώδικας του GPS χρησιμοποιεί την βιβλιοθήκη SoftwareSerial.h και ο κώδικας του GSM χρησιμοποιεί την GSM.h, οι οποίες δύο βιβλιοθήκες χρησιμοποιούν τα ίδια pin interrupts (pin 2 και pin 3). Συνεπώς, λόγω της σύγκρουσης των δύο συναρτήσεων, το πρόγραμμα δεν μπορούσε να τρέξει και έπρεπε να ανατρέξουμε σε αναζήτηση λύσης. Μετά από έρευνα που έγινε καταλήξαμε στην χρήση της συνάρτησης AltSoftSerial.h αντί της SoftwareSerial.h που χρησιμοποιεί το GPS μας. Η συνάρτηση αυτή ουσιαστικά έχει παρόμοια λειτουργία με την SoftwareSerial, με τη διαφορά ότι σαν pin interrupts χρησιμοποιεί τα pin 7 και 8. Έτσι έχουμε την GSM.h που χρησιμοποιεί ως pin interrupts τα pin 2 και 3 και την AltSoftSerial.h που χρησιμοποιεί ως pin interrupts τα pin 7 και 8. Με αυτόν τον τρόπο λύσαμε το πρόβλημα αυτό. [95]

Επιπλέον, χρησιμοποιήσαμε στον Α' τελικό κώδικα και την βιβλιοθήκη TinyGPS++.h που είναι μια νέα βιβλιοθήκη Arduino για την ανάλυση ροών δεδομένων NMEA που παρέχονται από μονάδες GPS. Παρέχει, δηλαδή, συμπαγείς και εύχρηστες μεθόδους για την εξαγωγή θέσης, ημερομηνίας, χρόνου, υψόμετρου, ταχύτητας και πορείας από τις συσκευές GPS του καταναλωτή.

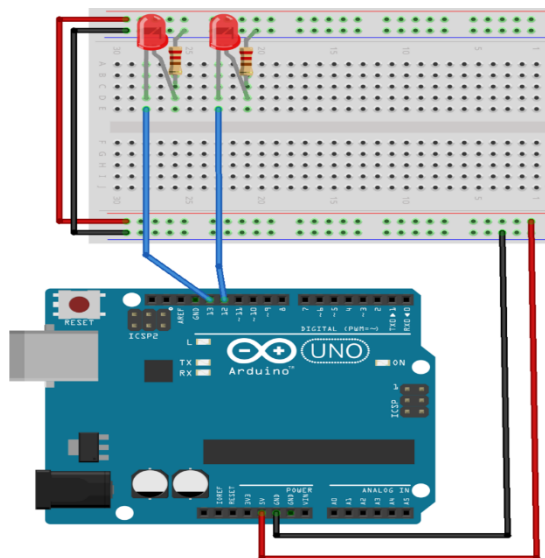
Η σύνδεση που υλοποιήσαμε φαίνεται στο παρακάτω σχήμα:



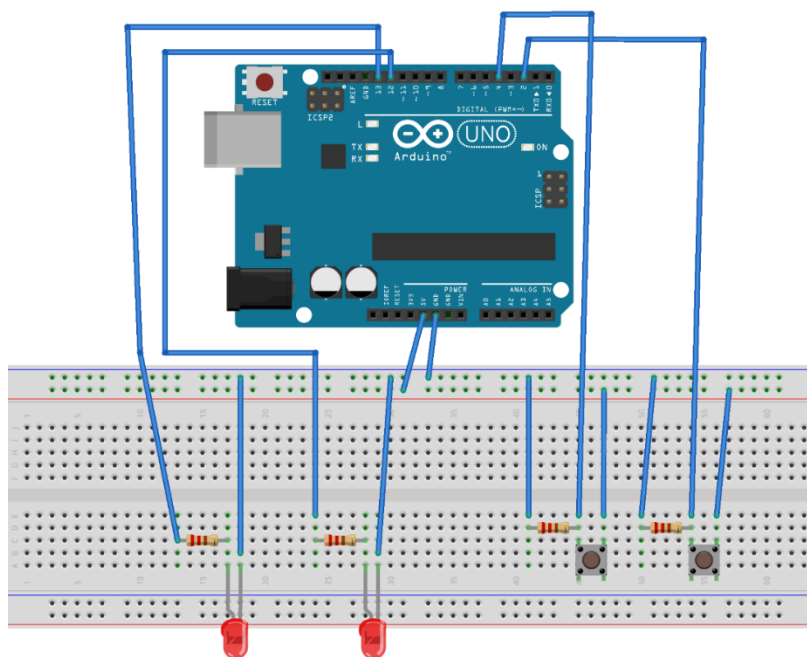
Όπως μπορούμε να δούμε, λόγω χρήσης της AltSoftSerial.h, συνδέσαμε τα RX και TX στα pins 9 και 8 αντίστοιχα. Εν συνεχεία ασχοληθήκαμε με τον κώδικα και προσπαθήσαμε να βγάλουμε το επιθυμητό αποτέλεσμα. Να σημειωθεί σε αυτό το σημείο ότι υλοποιήσαμε πέντε διαφορετικούς τελικούς κώδικες οι οποίοι παρατίθενται στο παράρτημα Β στο τέλος και ο καθένας επιτελεί μία ξεχωριστή και ιδιαίτερα χρήσιμη λειτουργία με βάση, όμως πάντα το παραπάνω μοτίβο.

### 3.4.1 Τελικές Προσθήκες στην Σύνδεση

Για τις ανάγκες του πρώτου και βασικού τελικού κώδικα που γράψαμε, θεωρήσαμε πως θα ήταν καλό να συνδέσουμε την διάταξή μας με δύο led διαφορετικού χρώματος, ο σκοπός των οποίων εξηγείται αναλυτικά στο επόμενο κεφάλαιο. Ο βασικός λόγος ένταξης των led έχει να κάνει με την καλύτερη εικόνα της συσκευής μας. Ο τρόπος με τον οποίο συνδέσαμε τα led με το Arduino είναι ο ακόλουθος:



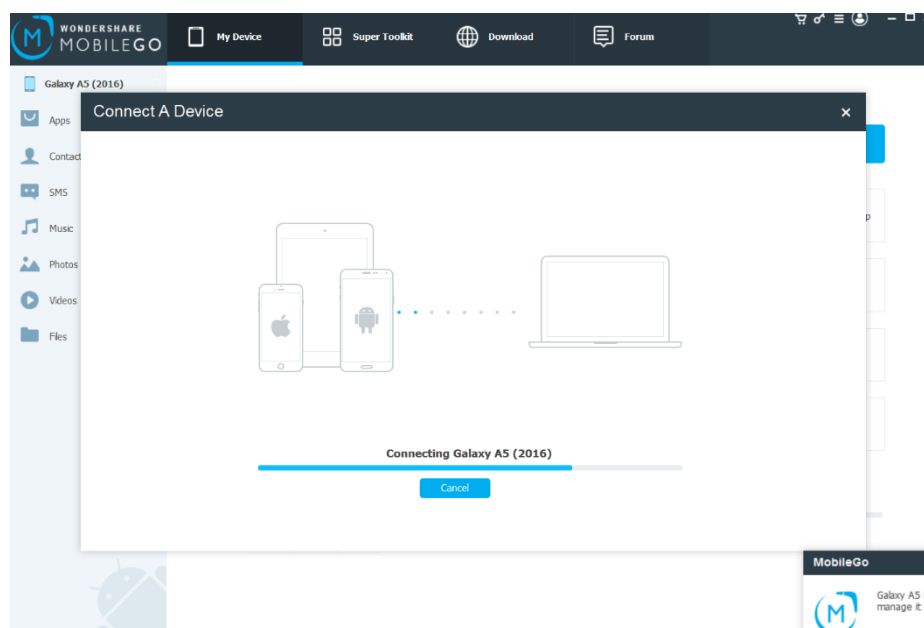
Έπειτα, πάλι για τις ανάγκες του πρώτου κώδικα χρειάστηκε να βάλουμε επιπλέον δύο push-buttons που θα αντιστοιχούν και θα ρυθμίζουν τη λειτουργία δύο ακόμα led. Ο λόγος ύπαρξης αυτών είναι καθαρά λειτουργικός και έχει να κάνει με αποθήκευση δεδομένων, στοιχεία που θα εξηγηθούν αναλυτικότερα στο επόμενο κεφάλαιο. Η σύνδεση των δύο push-buttons με τα δύο νέα led και το Arduino είναι η ακόλουθη:



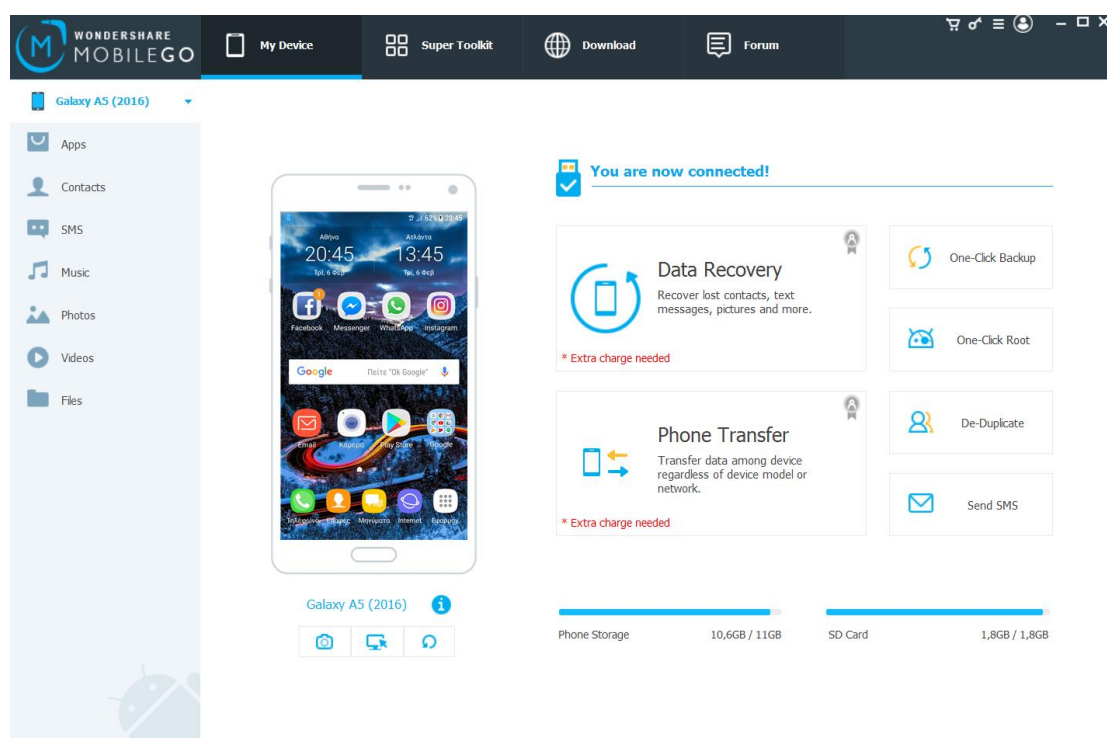
### 3.5 Σύνδεση κινητού τηλεφώνου με Η/Υ με αποτέλεσμα την επικοινωνία της συσκευής με τον Η/Υ

Έχοντας δείξει στο προηγούμενο κεφάλαιο τον τρόπο επικοινωνίας με τη βοήθεια του προγράμματος MobileGo, πάμε στο σημείο αυτό να δούμε τι συμβαίνει στο πρόγραμμα αυτό μόλις συνδέσουμε το κινητό που δέχεται τα μηνύματα, από τη συσκευή μας, στον υπολογιστή. Το αποτέλεσμα φαίνεται παρακάτω.

Αρχικά μόλις συνδέσαμε το κινητό μας εμφανίστηκε η εικόνα αυτή:

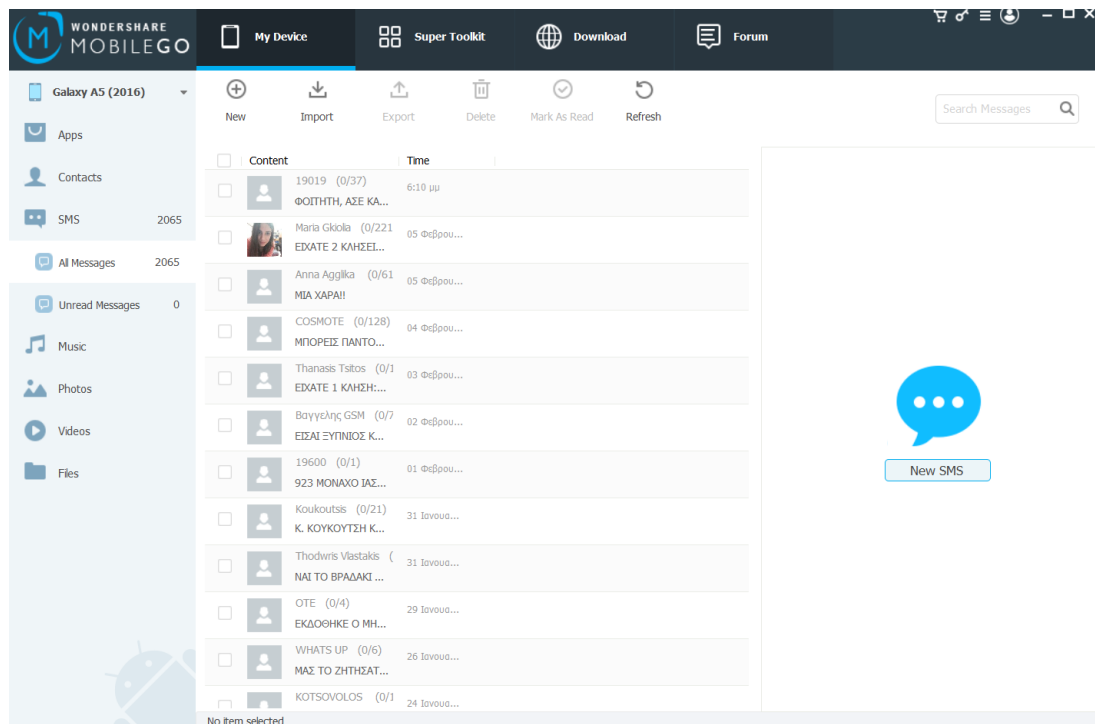


Και στη συνέχεια μόλις αναγνώρισε την συσκευή και το λειτουργικό της εμφανίστηκε η τελική εικόνα:



Όπως βλέπουμε εμφανίζεται η αρχική οθόνη του κινητού που έχουμε συνδέσει, καθώς και στοιχεία όπως η μνήμη του κινητού καθώς και η εξωτερική μνήμη SD που έχουμε βάλει. Αυτή τη στιγμή, λοιπόν, το κινητό μας έχει συνδεθεί στον ηλεκτρονικό υπολογιστή και μπορούμε να έχουμε πρόσβαση σε καθένα από τα στοιχεία της αριστερής στήλης, άρα και στα sms.

Πατώντας, λοιπόν, το κουμπί sms της αριστερής στήλης προέκυψε το ακόλουθο αποτέλεσμα:



Όπως, λοιπόν, μπορούμε να δούμε κάνοντας κλικ στο sms, μας εμφανίζονται όλα τα sms που έχει λάβει το κινητό μας. Συνεπώς, εφόσον λαμβάνουμε με sms τις συντεταγμένες που στέλνει η συσκευή μας στο κινητό τηλέφωνο, σημαίνει ότι μπορούμε να τις βλέπουμε τώρα και στον ηλεκτρονικό υπολογιστή μέσω του παραπάνω προγράμματος. Επομένως πετύχαμε την επικοινωνία της συσκευής μας με τον Η/Υ.

## ΚΕΦΑΛΑΙΟ 4 – ΥΛΟΠΟΙΗΣΗ

Ο πρώτος κώδικας που υλοποιήσαμε εκτελεί δύο διαφορετικές λειτουργίες που παρουσιάζονται στο κεφάλαιο αυτό.

### 4.1 Λήψη Στίγματος Τοποθεσίας - Με Αποστολή Μηνύματος

Ο στόχος της λειτουργίας αυτής είναι η συσκευή μας να μας στέλνει με γραπτό μήνυμα τις συντεταγμένες που λαμβάνει το GPS, μόνο αν του στείλουμε εμείς πρώτα γραπτό SMS με την σωστή λέξη, που στην περίπτωση μας είναι η λέξη location.

#### Αρχικός Αλγόριθμος

- **Input:** message1 //το SMS που στέλνουμε εμείς στην κάρτα της συσκευής
- **Output:** message2, led1, led2 //οι συντεταγμένες του GPS που θα μας επιστρέψει η συσκευή και τα αντίστοιχα led
- Αναμονή μέχρι να πιάσει στίγμα το GPS
- Αναμονή αποστολής κάποιου μηνύματος (message1)
- Αν message1 = location, τότε στείλε μήνυμα με τις συντεταγμένες και άναψε το led1, δηλαδή message2=TRUE && led1=HIGH
- Αν message1 != location, τότε μην στείλεις τίποτα και άναψε το led2, δηλαδή message2=FALSE && led2=HIGH
- Καθυστέρηση
- Επανάληψη

Ουσιαστικά αυτό που γίνεται είναι το εξής:

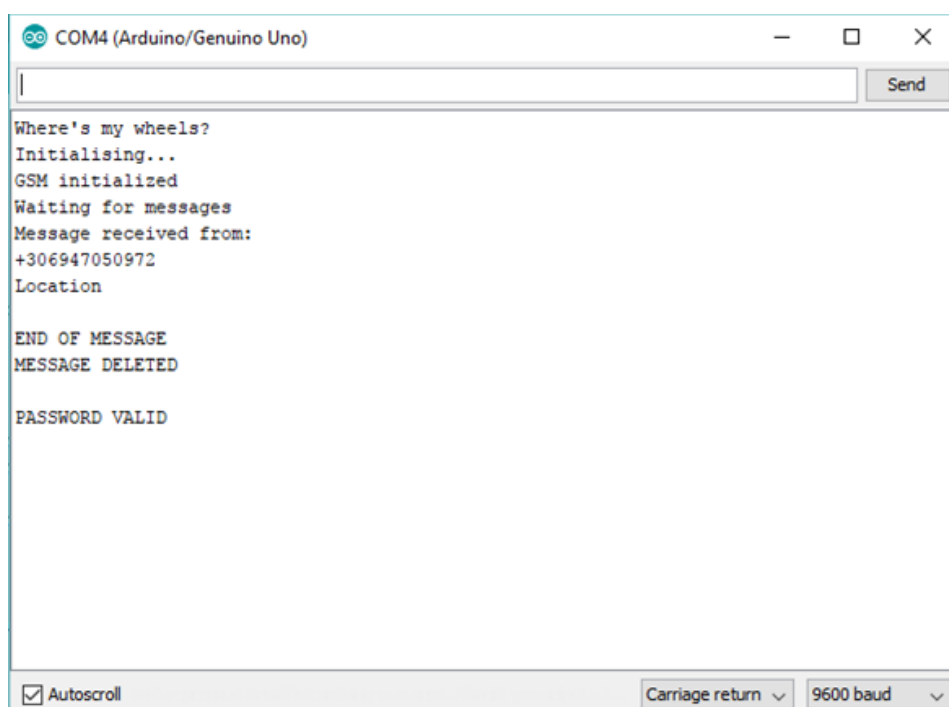
Συνδέουμε την συσκευή μας στον υπολογιστή, τρέχουμε τον κώδικα και περιμένουμε λίγα δευτερόλεπτα μέχρι το GPS να λάβει στίγμα τοποθεσίας από τους δορυφόρους. Αυτό το καταλαβαίνουμε όταν το κόκκινο led του GPS σταματήσει να αναβοσβήνει (όσο το GPS ψάχνει να βρει σήμα το κόκκινο led αναβοσβήνει). Τότε σημαίνει ότι έχουμε σήμα. Παράλληλα επειδή ο κώδικας τρέχει, το πρόγραμμα έχει φτάσει σε ένα σημείο και περιμένει να λάβει σήμα από τον χρήστη για να του στείλει τα δεδομένα τοποθεσίας. Όπως μπορούμε να δούμε και στον κώδικα, υπάρχει μία εντολή στην οποία έχουμε ορίσει τη μεταβλητή yourPassword στην οποία έχουμε αποθηκεύσει τη λέξη "Location". Πιο κάτω υπάρχει μια εντολή, η password=yourPassword, η οποία περιμένει να δει άμα θα στείλουμε την σωστή λέξη (δηλαδή Location) στην κάρτα SIM που χρησιμοποιεί η συσκευή μας.

Περιμένει, λοιπόν, το πρόγραμμα να του στείλει ο χρήστης SMS με τη λέξη Location, προκειμένου η συσκευή να του στείλει πίσω με γραπτό μήνυμα την τοποθεσία του. Αν ο χρήστης στείλει την λέξη Location (στον αριθμό της κάρτας SIM που έχουμε εισάγει στη συσκευή μας), τότε εμφανίζεται το μήνυμα VALID PASSWORD στην οθόνη του υπολογιστή και παράλληλα ανάβει το μπλε led και σε πολύ λίγα δευτερόλεπτα, ο χρήστης λαμβάνει στο κινητό του το στίγμα τοποθεσίας της συσκευής. Από την άλλη εάν ο χρήστης στείλει με SMS μια διαφορετική λέξη από τη λέξη Location, τότε εμφανίζεται το μήνυμα INVALID PASSWORD στην οθόνη του υπολογιστή και παράλληλα ανάβει το κίτρινο led.

Η βασική αρχή, δηλαδή, του κώδικα είναι ότι λαμβάνει όλα τα κείμενα που αποστέλλονται σε αυτόν και αν περιέχουν τον κωδικό πρόσβασης ("Location") τότε το Arduino θα στείλει ένα γραπτό SMS στον αριθμό του εκάστοτε χρήστη που του έστειλε, με τις τρέχουσες συντεταγμένες GPS. Οι συντεταγμένες GPS θα ενημερώνονται αυτόματα κάθε δευτερόλεπτο και συνεπώς θα έχουμε πάντα μια ενημερωμένη και ακριβή τοποθεσία όποτε στέλνει.

#### 4.1.1 Φωτογραφίες Αποτελεσμάτων από Πρώτη Λειτουργία Α' Κώδικα

Τρέξαμε, λοιπόν, τον κώδικα και στείλαμε από το κινητό μας την λέξη "Location" που είναι το σωστό password στην συσκευή μας. Αυτό που εμφανίστηκε στην σειρική έξοδο του Arduino IDE είναι το εξής:



```

COM4 (Arduino/Genuino Uno)
Where's my wheels?
Initialising...
GSM initialized
Waiting for messages
Message received from:
+306947050972
Location

END OF MESSAGE
MESSAGE DELETED

PASSWORD VALID
  
```

Όπως μπορούμε να δούμε το πρόγραμμα ξεκινάει να τρέχει και εμφανίζονται στην σειριακή έξοδο οι σειρές:

*Where's my wheels?*

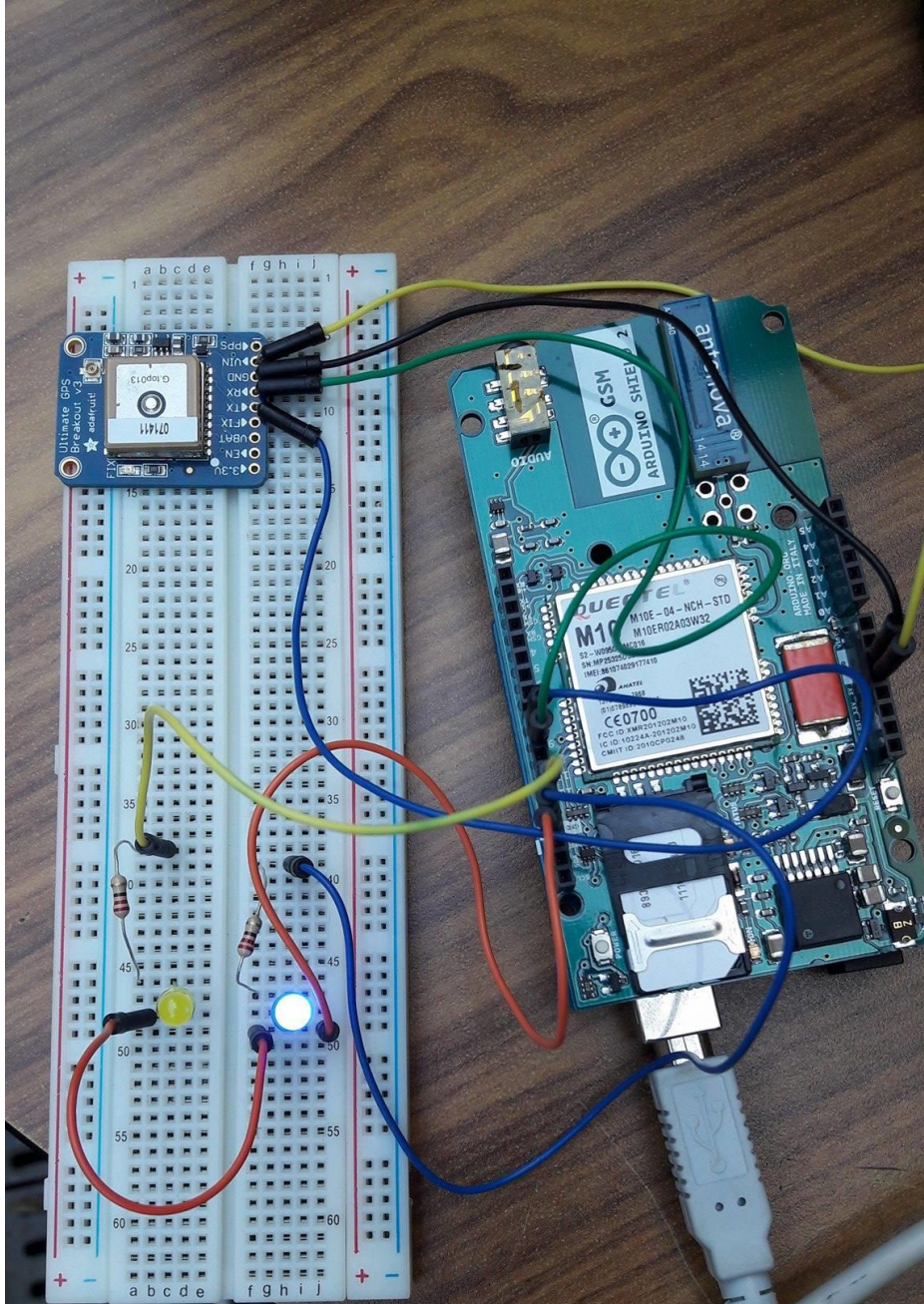
*Initialized...*

*GSM initialized*

*Waiting for messages*

Σε αυτή τη φάση η συσκευή μας περιμένει να στείλει κάποιος αριθμός ένα μήνυμα σε αυτήν, και αν είναι η λέξη Location, τότε η συσκευή μας να επιστρέψει πίσω στον αριθμό, που της έστειλε το μήνυμα, τις συντεταγμένες του GPS.

Στείλαμε, λοιπόν, από ένα κινητό τηλέφωνο με αριθμό +306947050972 την λέξη “Location” με γραπτό SMS στον αριθμό της κάρτας SIM που έχει εισαχθεί στη συσκευή μας. Βλέπουμε στην οθόνη ότι εμφανίζεται το μήνυμα PASSWORD VALID, διότι στείλαμε το σωστό. Σε αυτό το σημείο στην συσκευή μας άναψε το μπλε led, καθώς της στάλθηκε το σωστό password.

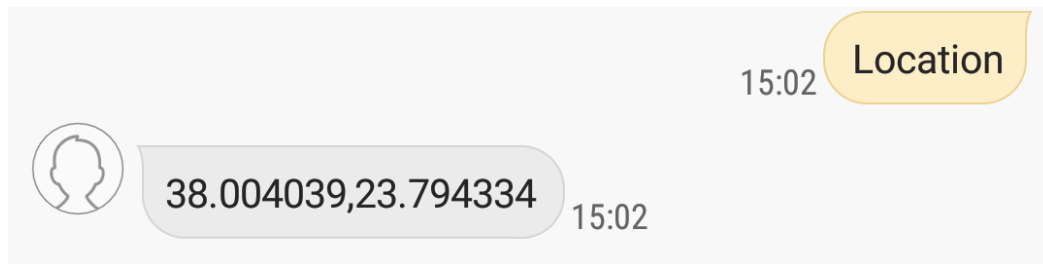


*Password Valid – Μπλε Led Αναμμένο*

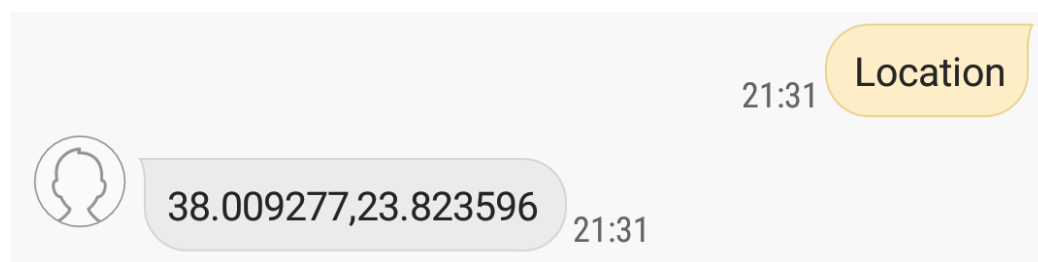
Μετά από ελάχιστα δευτερόλεπτα λάβαμε μήνυμα στον αριθμό του κινητού τηλεφώνου από το οποίο στείλαμε την λέξη “Location” με τις συντεταγμένες του GPS.

Η οθόνη του κινητού μας με τις συντεταγμένες φαίνεται παρακάτω:





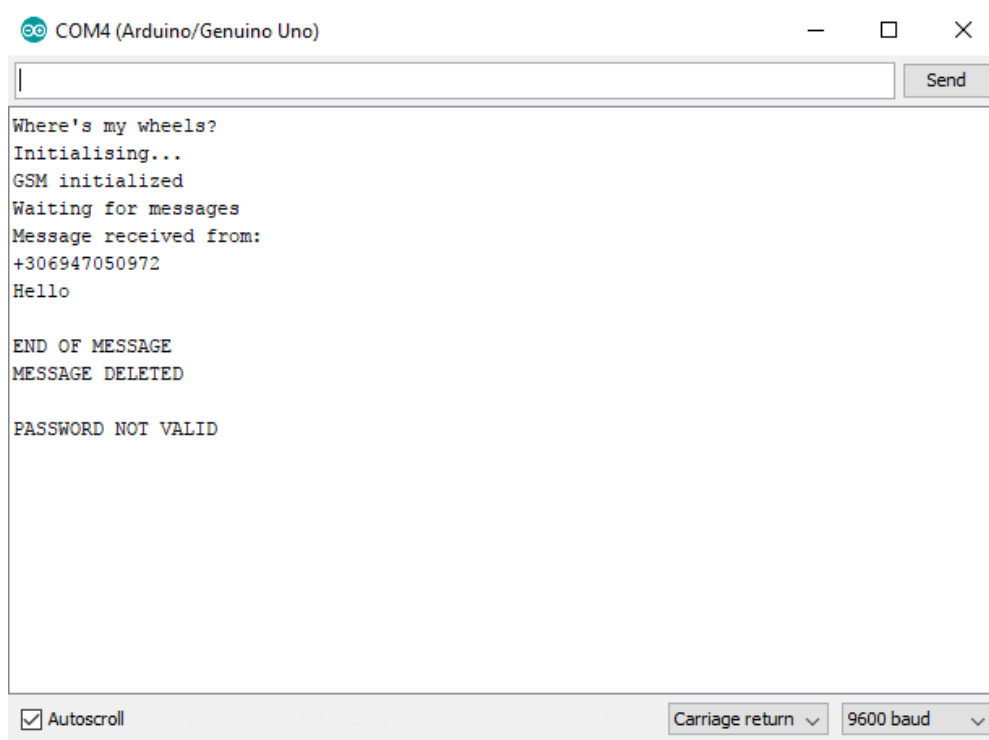
*1<sup>η</sup> δοκιμή στην τοποθεσία Χολαργός*



*2<sup>η</sup> δοκιμή στην τοποθεσία Αγία Παρασκευή*

Στη συνέχεια βάλαμε τις συντεταγμένες αυτές στο Google Maps για να δούμε αν ανταπεξέρχονται στην πραγματικότητα. Και στις δύο δοκιμές που κάναμε το Google Maps μας επιβεβαίωσε τις συντεταγμένες που βρισκόμασταν. Μάλιστα η συσκευή μας είχε και πάρα πολύ μεγάλη ακρίβεια στο στίγμα τοποθεσίας που έστειλε.

Σε επόμενη φάση δοκιμάσαμε να στείλουμε ως μήνυμα μια διαφορετική λέξη από το "Location" που είναι για να δούμε αν θα λειτουργήσει σωστά με βάση τις προδιαγραφές που θέσαμε. Αυτό που εμφανίστηκε στην σειριακή έξοδο του Arduino IDE είναι το εξής:



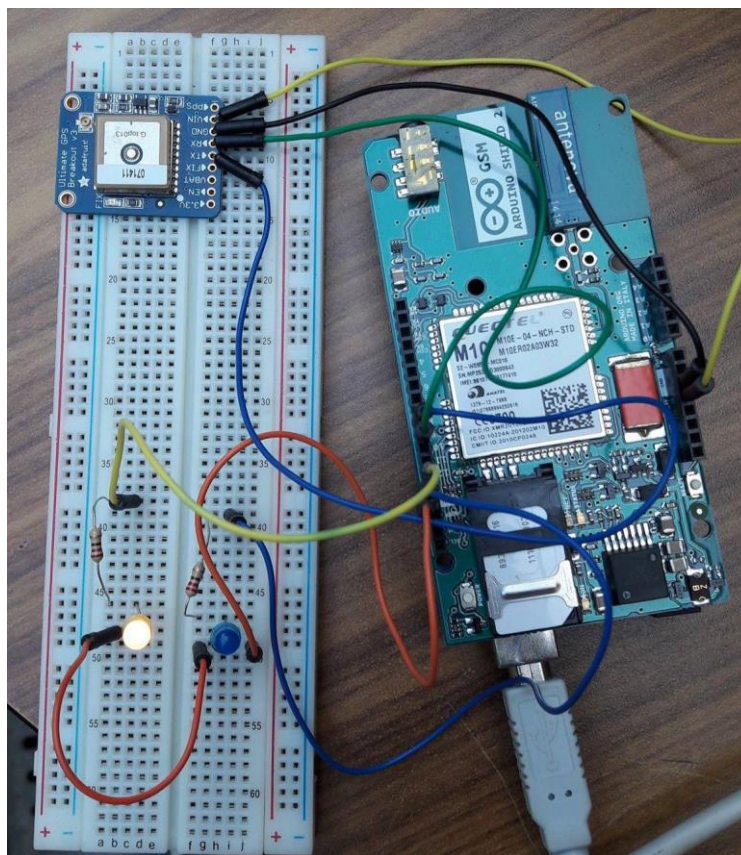
```

COM4 (Arduino/Genuino Uno)
Where's my wheels?
Initialising...
GSM initialized
Waiting for messages
Message received from:
+306947050972
Hello

END OF MESSAGE
MESSAGE DELETED

PASSWORD NOT VALID
  
```

Βλέπουμε ότι εμφανίστηκε το μήνυμα PASSWORD NOT VALID. Επιπλέον στην διάταξη μας άναψε το κίτρινο led, όπως ακριβώς το είχαμε ρυθμίσει:



*Password Not Valid – Κίτρινο Led Αναμμένο*

Προφανώς, όπως ακριβώς το είχαμε προγραμματίσει, η συσκευή μας δεν έστειλε πίσω στον αριθμό, που της έστειλε το μήνυμα, τις συντεταγμένες του GPS, καθώς δεν είχε σταλεί η σωστή λέξη.

#### 4.2 Λήψη Ομάδας Στιγμάτων Τοποθεσίας Με Χρήση Κουμπιών

Στο σημείο αυτό κάναμε μία επέκταση στον ήδη υπάρχοντα κώδικα προσθέτοντας μια επιπλέον λειτουργία με χρήση 2 κουμπιών, ένα που αντιστοιχεί σε ένα πράσινο led και ένα που αντιστοιχεί σε ένα πορτοκαλί led. Κάθε φορά που πατάμε κάποιο εκ των δύο κουμπιών ανάβει το αντίστοιχο led και μας επιστρέφονται οι συντεταγμένες του GPS μαζί με το γράμμα g ή y ανάλογα με το κουμπί που πατήσαμε. Επιπλέον, με τη χρήση των κουμπιών επιτελέσαμε και μία λειτουργία αποθήκευσης συντεταγμένων και αποστολής τους ανά ομάδες. Δηλαδή, με διαδοχικά πατήματα των κουμπιών αποθηκεύονται, αλλά δεν στέλνονται, οι συντεταγμένες του πιάνει το GPS κάθε στιγμή και όποια στιγμή θέλουμε, τότε μας στέλνει το GPS όλες τις συντεταγμένες που αποθηκεύτηκαν κατά τη διαδρομή.

#### Συνολικός Αλγόριθμος Λειτουργίας του Α' Τελικού Κώδικα

Αρχικά παραθέτουμε τον συνολικό αλγόριθμο του Α' Κώδικα, που είναι υπεύθυνος τόσο για την πρώτη όσο και για τη δεύτερη λειτουργία που αναφέραμε:

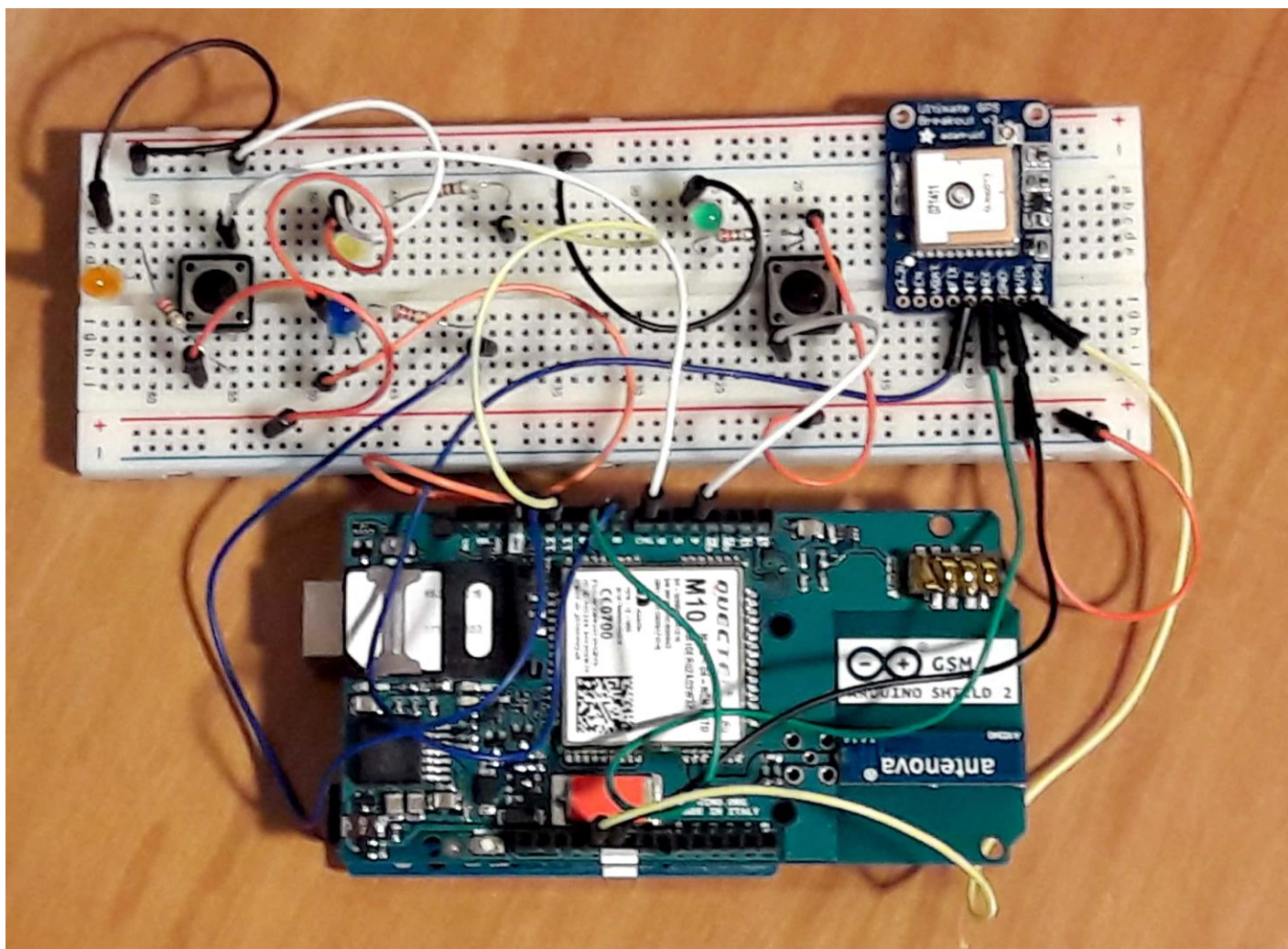
- **Input:** green button, orange button, message1
- **Output:** message2, led1, led2, led3, led4
- Αναμονή μέχρι να πιάσει στίγμα το GPS
- Αναμονή αποστολής κάποιου μηνύματος (message1 or press green button or press orange button)
- Αν message1 = location, τότε στείλε μήνυμα με τις συντεταγμένες και άναψε το led1, δηλαδή message2=TRUE && led1=HIGH
- Αν message1 != location, τότε μην στείλεις τίποτα και άναψε το led2, δηλαδή message2=FALSE && led2=TRUE
- Αν πατηθεί το κουμπί που αντιστοιχεί στο πράσινο led τότε γράψε τις συντεταγμένες του GPS εκείνη τη στιγμή μαζί με τον χαρακτήρα g στον buffer και άναψε το πράσινο led3
- Αν πατηθεί το κουμπί που αντιστοιχεί στο πορτοκαλί led τότε γράψε τις συντεταγμένες του GPS εκείνη τη στιγμή μαζί με τον χαρακτήρα y στον buffer και άναψε το πορτοκαλί led4
- Αν buffer > κάποιο όριο που θέτουμε εμείς, τότε στείλε μήνυμα με όλες τις συντεταγμένες που έχεις αποθηκεύσει, δηλαδή message2=TRUE. Και στη συνέχεια ετοιμάσου να δημιουργήσεις νέο μήνυμα, οπότε άδειασε τον buffer σου.
- Καθυστέρηση
- Επανάληψη

Στο σημείο αυτό να διευκρινίσουμε το όριο που θέσαμε στον παραπάνω αλγόριθμο. Κάθε SMS αποτελείται από 140 χαρακτήρες. Υπολογίσαμε τον αριθμό των χαρακτήρων που αποθηκεύονται κάθε φορά στον buffer από τις συντεταγμένες και

τον βρήκαμε ίσο με 20 ( 9 από κάθε συντεταγμένη, 1 κόμμα και 1 γράμμα g ή y ανάλογα με το κουμπί). Συνεπώς επιλέξαμε το όριο αποθήκευσης να μην ξεπερνάει το ένα SMS. Άρα, το όριο που επιλέξαμε ήταν το 120, άρα κάθε  $120/20 = 6$  πατήματα κουμπιού μας αποστέλεται ένα γραπτό SMS με τις 6 συντεταγμένες που έχουν αποθηκευτεί στον buffer έως τώρα. Φυσικά θα μπορούσαμε να θέσουμε ότι όριο θέλουμε εμείς.

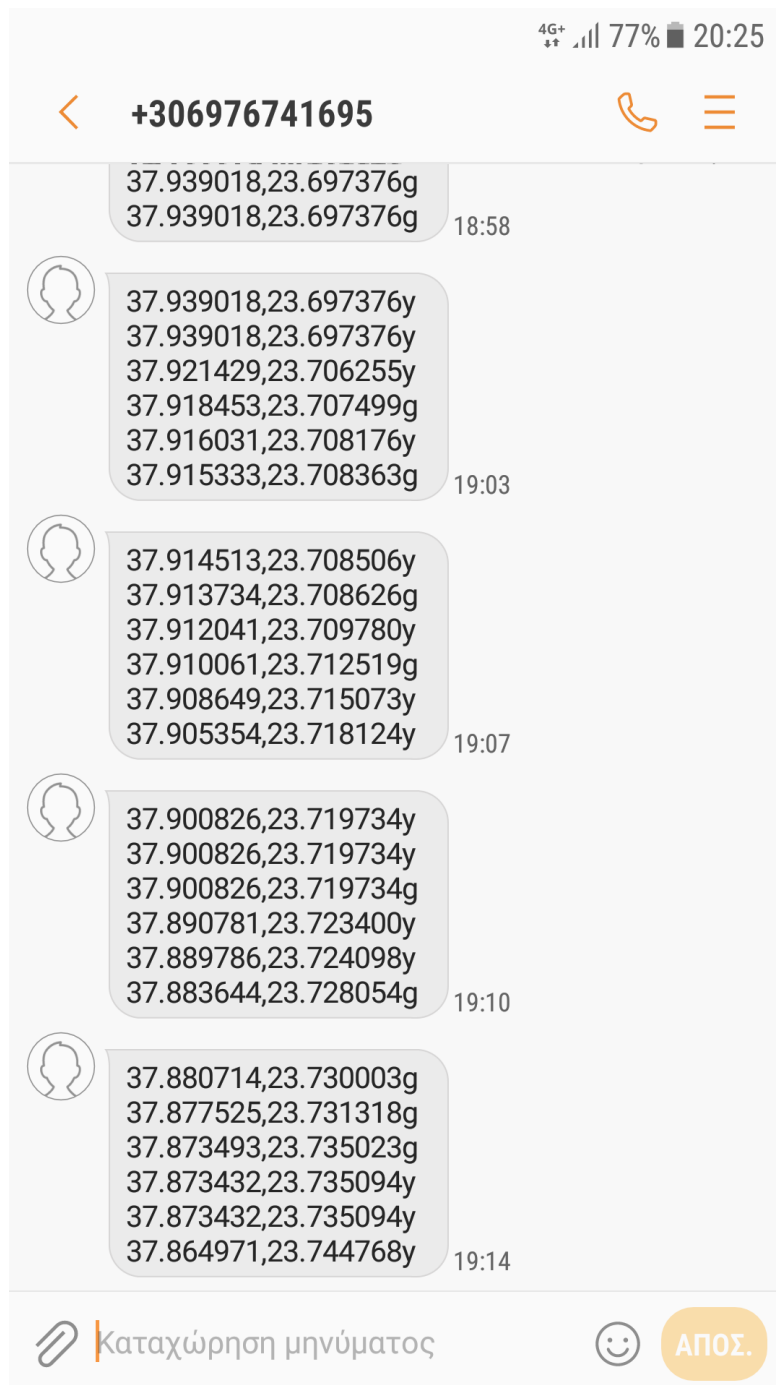
#### 4.2.1 Φωτογραφίες Αποτελεσμάτων από Δεύτερη Λειτουργία Α' Κώδικα

Στο υποκεφάλαιο αυτό παρουσιάζουμε φωτογραφία από την τελική συνολική διάταξη μας, με την προσθήκη των κουμπιών και των αντίστοιχων led:



Παράλληλα, παραθέσαμε και φωτογραφία από τα μηνύματα που μας στάλθηκαν από την λειτουργία με τα κουμπιά. Να σημειωθεί ότι η λήψη των αποτελεσμάτων αυτών λήφθηκαν κατά τη διαδρομή Π. Φάληρο – Βούλα όπου θέσαμε τη συσκευή μας σε λειτουργία και πατούσαμε καθένα εκ των δύο κουμπιών κατά τη διαδρομή. Στη συνέχεια τα αποτελέσματα αυτά ελέγχθηκαν μέσω του Google Maps και πράγματι

επαληθεύτηκαν με μεγάλη ακρίβεια. Τα αποτελέσματα που λάβαμε κατά τη διαδρομή φαίνονται στην παρακάτω φωτογραφία.



*Συντεταγμένες κατά τη διαδρομή Π. Φάληρο - Βούλα*

Βλέπουμε ότι η συσκευή μας επιστρέφει ομάδα συντεταγμένων και όχι μία μία μεμονωμένες. Αυτή η λειτουργία είναι πολύ χρήσιμη, καθώς με αυτόν τον τρόπο μπορούμε να ελέγξουμε όχι απλώς τις συντεταγμένες που βρίσκεται η συσκευή μας αυτή τη στιγμή, αλλά και το κατά πόσο πέρασε από συγκεκριμένες τοποθεσίες.



















Παρακάτω παραθέτουμε φωτογραφίες με τις συντεταγμένες που μας επέστρεψε η συσκευή μας, που επαληθεύουν με τη βοήθεια του Google Maps την θέση που βρισκόμασταν και συνεπώς την ορθή λειτουργία της συσκευής μας.



Χάρτης χωρίς τίτλο

Αποθήκευση όλων των αλλαγών στο Drive

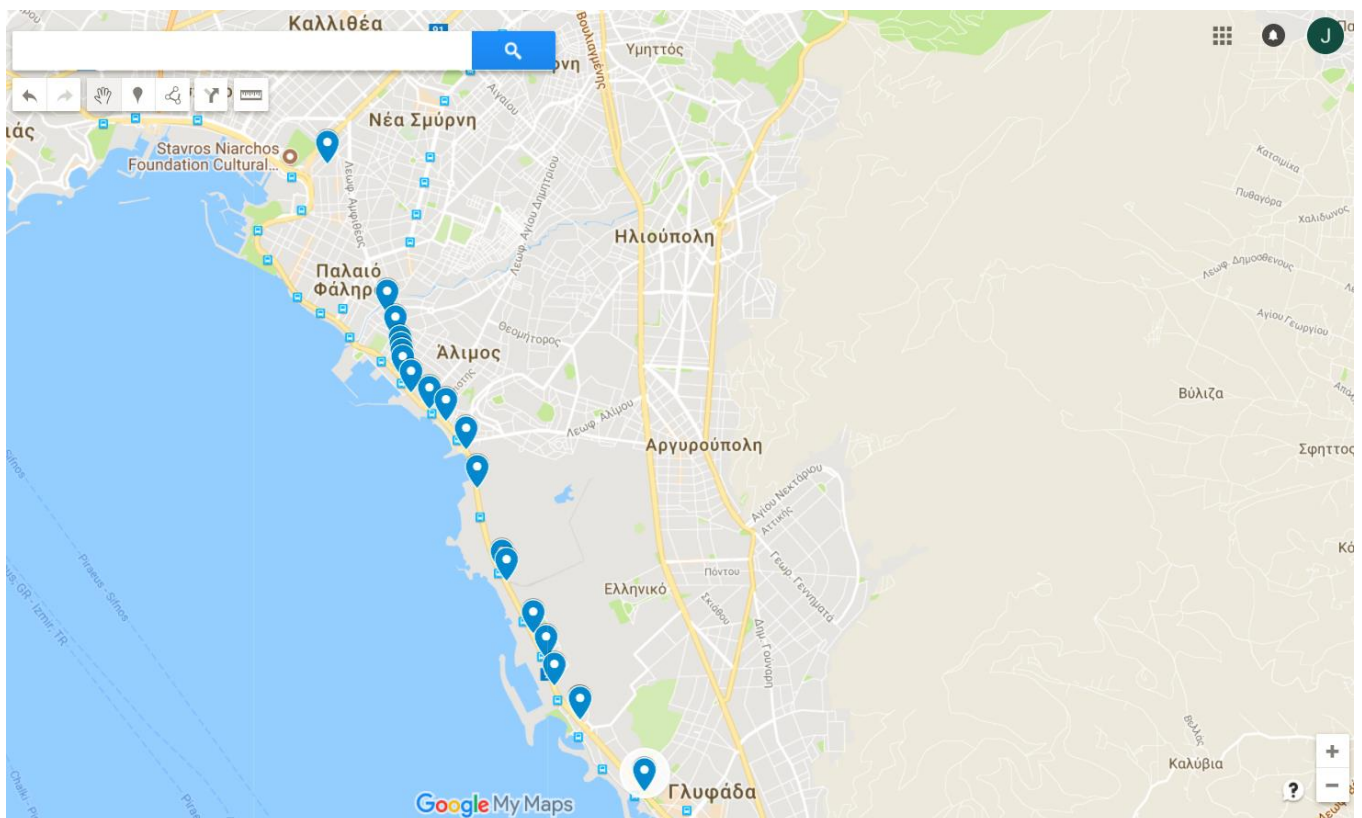
Προσθήκη επιπέδου
  Κοινοποίηση

Προεπισκόπηση

-  37.939018,23.697376
-  37.921429,23.706255
-  37.918453,23.707499
-  37.916031,23.708176
-  37.915333,23.708363
-  37.914513,23.708506
-  37.913734,23.708626
-  37.912041,23.709780
-  37.910061,23.712519
-  37.908649,23.715073
-  37.905354,23.718124
-  37.900826,23.719734
-  37.890781,23.723400
-  37.889786,23.724098
-  37.883644,23.728054
-  37.880714,23.730003
-  37.877525,23.731318
-  37.873493,23.735023

-  37.873432,23.735094
-  **37.864971,23.744768**

Βασικός χάρτης



### 4.3 Συνεχόμενη Λήψη Ομάδας Στιγμάτων Τοποθεσίας

Στο σημείο αυτό, αφού έχουμε εκτελέσει τις παραπάνω λειτουργίες που αναφέρθηκαν, υλοποιήσαμε έναν κατάλληλο κώδικα ο οποίος τρέχει αέναα και κάνει αποθήκευση συντεταγμένων, αυτή τη φορά χωρίς πάτημα κάποιου κουμπιού, αλλά ανά κάθε επανάληψη του προγράμματος. Στη συνέχεια, όταν έχει αποθηκευτεί ικανός για την κάθε περίπτωση αριθμός συντεταγμένων, αποστέλονται με γραπτό SMS όλες οι συντεταγμένες που είχαν αποθηκευτεί σαν ομάδα. Πιο συγκεκριμένα, ο αλγόριθμος που υλοποιήσαμε είναι ο εξής:

#### Συνολικός Αλγόριθμος Λειτουργίας του Β' Τελικού Κώδικα

- **Input:** -
- **Output:** message
- Αναμονή μέχρι να πιάσει στίγμα το GPS
- Γράψε τις συντεταγμένες του GPS στον buffer
- Αν buffer > κάποιο όριο που θέτουμε εμείς, τότε στείλε μήνυμα με όλες τις συντεταγμένες που έχεις αποθηκεύσει, δηλαδή message=TRUE. Και στη συνέχεια ετοιμάσου να δημιουργήσεις νέο μήνυμα, δηλαδή άδειασε τον buffer
- Καθυστέρησε
- Επανάληψη

Όπως, είπαμε και παραπάνω το όριο μπορούμε να το καθορίσουμε εμείς. Στο συγκεκριμένο κώδικα βάλουμε το όριο 250. Αυτό σημαίνει, δεδομένου ότι κάθε μήνυμα αποτελείται από 140 χαρακτήρες και ότι κάθε φορά οι συντεταγμένες αποτελούνται από 19 χαρακτήρες (9 για κάθε συντεταγμένη και 1 χαρακτήρας το κόμμα που τις διαχωρίζει), μόλις έρθει η κατάλληλη στιγμή, θα μας σταλεί μήνυμα με  $250/19 = 13.2$ , δηλαδή 13 συντεταγμένες, που θα έρθουν σε δύο ξεχωριστά μηνύματα, το ένα με 7 συντεταγμένες και το άλλο με 6, διότι  $140/19=7.4$ . Αντίστοιχα με πριν, έτσι κι εδώ μπορούμε να ορίσουμε εμείς το όριο χαρακτήρων που θέλουμε, άρα κατά συνέπεια τον αριθμό των συντεταγμένων που θέλουμε. Το αποτέλεσμα που λάβαμε φαίνεται στην επόμενη φωτογραφία.

```
38.020702,23.815199
38.020649,23.814938
38.020638,23.814871
38.020638,23.814871
38.020638,23.814871
38.022911,23.815065
38.022812,23.815279
```

```
38.022797,23.815311
38.022778,23.815357
38.022758,23.815407
38.022756,23.814809
38.022644,23.814823
38.022729,23.815062
```

Όπως βλέπουμε, με όριο 250 κάναμε ένα testing και βλέπουμε ότι η συσκευή μας, μας επέστρεψε δύο ομάδες συντεταγμένων αποτελούμενες από 7 και 6 συντεταγμένες αντίστοιχα. Στη συνέχεια επαληθεύσαμε τις συντεταγμένες αυτές με τη βοήθεια του Google Maps και ήταν σωστές.

#### 4.4 Συνεχής Αποστολή Στιγμάτων Τοποθεσίας με Ταυτόχρονη Αποστολή του Χρόνου

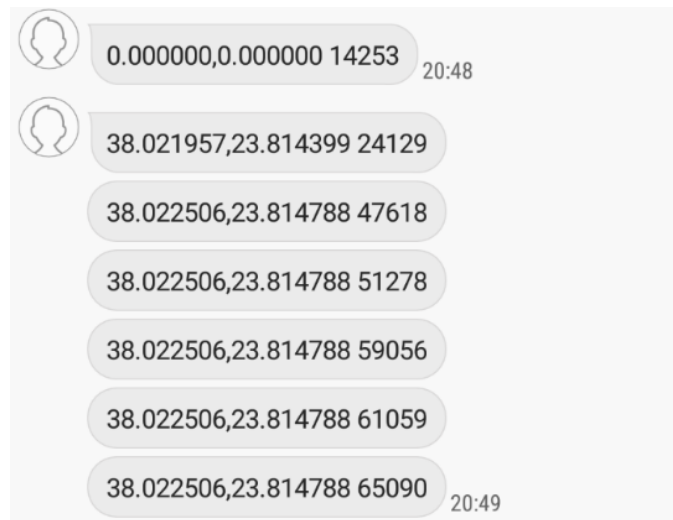
Στο σημείο αυτό βάλουμε στο παιχνίδι το χρόνο, με τη βοήθεια του ρολογιού που διαθέτει το Arduino. Αυτό που θέλαμε να υλοποιήσουμε είναι ουσιαστικά η συσκευή μας να μας στέλνει κάθε φορά στο κινητό μας, μία μία κάθε συντεταγμένη που πιάνει και παράλληλα δίπλα από τις συντεταγμένες να στέλνει και τη ροή του χρόνου σε ms, ώστε να δούμε πόσο γρήγορα μπορεί να στέλνει συντεταγμένες η συσκευή μας. Για το λόγο αυτό χρησιμοποιήσαμε μία πολύ χρήσιμη συνάρτηση του Arduino, την millis(). Η συνάρτηση αυτή επιστρέφει τον αριθμό των χιλιοστών του δευτερολέπτου (ms) από τότε που άρχισε να τρέχει το τρέχον πρόγραμμα Arduino. Αυτός ο αριθμός θα υπερχειλίσει (επιστρέψει στο μηδέν), μετά από περίπου 50 ημέρες. Αντί για τη συνάρτηση αυτή θα μπορούσαμε να χρησιμοποιήσουμε την micros(), η οποία επιτελεί ακριβώς την ίδια λειτουργία με την millis(), με την μόνη διαφορά ότι επιστρέφει το χρόνο σε μs, γεγονός που δεν μας εξυπηρετεί ιδιαίτερα, λόγω πολλών ψηφίων που θα επιστρέφονταν με το μήνυμα των συντεταγμένων. Ας δούμε τώρα τον αλγόριθμο που υλοποιήσαμε για την λειτουργία αυτή.

#### Συνολικός Αλγόριθμος Λειτουργίας του Γ' Τελικού Κώδικα

- **Input:** -
- **Output:** message
- Αναμονή να πιάσει στίγμα το GPS
- Ρωτάμε το Arduino το χρόνο μέχρι τώρα και τον αποθηκεύουμε στην μεταβλητή y
- Γράψε τις συντεταγμένες του GPS που μόλις έπιασες και αποθήκευσέ τις
- Πρόσθεσε στις συντεταγμένες ένα κενό και στην συνέχεια τη μεταβλητή y του χρόνου
- Στείλε μήνυμα με τις συντεταγμένες και το χρόνο, δηλαδή message=TRUE. Στη συνέχεια ετοιμάσου για νέο μήνυμα, άρα άδειασε την μνήμη.
- Επανάληψη

Στο σημείο αυτό βγήκαμε και πάλι έξω στο δρόμο με το αυτοκίνητο για να ελέγξουμε την ορθή λειτουργία του κώδικα που γράψαμε παραπάνω με το χρόνο. Το αποτέλεσμα που λάβαμε είναι το ακόλουθο:





Όπως μπορούμε να δούμε μας επιστρέφονται οι συντεταγμένες που βρισκόμαστε κάθε φορά και δίπλα ακριβώς είναι ο χρόνος του προγράμματος μέχρι εκείνη τη στιγμή που γίνεται η αποστολή σε ms. Προφανώς, όπως κάθε φορά επαληθεύσαμε τις συντεταγμένες της συσκευής μας με το Google Maps και ήταν πολύ σωστές. Επιπλέον, παρατηρούμε ότι αρχικά που το GPS δεν πρόλαβε να πιάσει σήμα, το πρόγραμμα έκανε 14,253 s για να στείλει το πρώτο μήνυμα. Στην συνέχεια βλέπουμε ότι το επόμενο μήνυμα ήρθε στο 24,129 s, το τρίτο στο 47,618 s, το τέταρτο στο 51,278 s κ.ο.κ. Άρα αυτό που μπορούμε να υπολογίσουμε με ευκολία είναι ο χρόνος αποστολής μεταξύ δύο διαδοχικών μηνυμάτων συντεταγμένων, με βάση πάντα το ρολόι του Arduino. Δηλαδή ο χρόνος ανάμεσα σε δύο τυπικά sms συντεταγμένων που αποστέλλονται προς το κινητό μας ή οποιαδήποτε άλλη συσκευή. Έτσι στο παράδειγμά μας βλέπουμε ότι το πρώτο με το δεύτερο sms συντεταγμένων έχουν χρονική διαφορά ίση με  $24,129 - 14,253 = 9,876s$ , το δεύτερο με το τρίτο έχουν  $47,618 - 24,129 = 23,489s$ , το τρίτο με το τέταρτο  $51,278 - 47,618 = 3,66s$  κ.ο.κ. Αυτό που παρατηρούμε, λοιπόν, είναι ότι ο χρόνος μεταξύ δύο διαδοχικών sms συντεταγμένων δεν είναι σταθερός, πράγμα που θα φτιάξουμε στο επόμενο υποκεφάλαιο. Προφανώς αυτή η αστάθεια οφείλεται στο σήμα κινητής τηλεφωνίας, σε τεχνικές αποκλίσεις της συσκευής μας καθώς και στις καιρικές συνθήκες που επικρατούν, που επηρεάζουν το στίγμα του GPS.

#### 4.5 Χρονικά Ελεγχόμενη και Συνεχής Αποστολή Στιγμάτων Τοποθεσίας

Όπως είδαμε στο προηγούμενο υποκεφάλαιο, ο χρόνος ανάμεσα σε δύο τυπικά sms συντεταγμένων, με βάση το ρολόι του Arduino δεν ήταν σταθερός. Αυτό οφείλεται όπως προείπαμε στο σήμα της τηλεφωνίας, αλλά και σε πιθανές τεχνικές αποκλίσεις της συσκευής μας. Στο υποκεφάλαιο αυτό θα προσπαθήσουμε να λύσουμε το πρόβλημα αυτό και να κάνουμε την συσκευή μας να αποστέλει μηνύματα συντεταγμένων με όσο γίνεται πιο σταθερό ρυθμό. Αυτό, λοιπόν, πρακτικά που κληθήκαμε να κάνουμε είναι μια δειγματοληψία των στιγμάτων τοποθεσίας,

προκειμένου να στείλουμε συγκεκριμένο αριθμό από αυτά με αποτέλεσμα να πετύχουμε αποστολή μηνυμάτων με σταθερό ρυθμό.

### Συνολικός Αλγόριθμος Λειτουργίας του Δ' Τελικού Κώδικα

- **Input:** -
- **Output:** message
- Αναμονή να πιάσει στίγμα το GPS
- Ρωτάμε το Arduino το χρόνο μέχρι τώρα και τον αποθηκεύουμε στην μεταβλητή t
- Εάν ο χρόνος  $t \geq x+2000$  τότε ξαναρώτα το Arduino για το χρόνο και αποθήκευσε το αποτέλεσμα αυτή τη φορά στην μεταβλητή x
  - Στη συνέχεια, γράψε τις συντεταγμένες του GPS που μόλις έπιασες και αποθήκευσέ τις
  - Πρόσθεσε στις συντεταγμένες ένα κενό και στην συνέχεια τη μεταβλητή x του χρόνου
  - Στείλε μήνυμα με τις συντεταγμένες και το χρόνο, δηλαδή message=TRUE. Στη συνέχεια ετοιμάσου για νέο μήνυμα, άρα άδειασε την μνήμη.
- Αλλιώς εάν  $t < x+2000$  τότε απλά:
- Επανάληψη

Σε αυτό το σημείο πρέπει να διευκρινίσουμε πολύ καλά την εντολή  $if t \geq x+2000$ . Αυτή είναι η συνθήκη δειγματοληψίας που εκτελέσαμε προκειμένου να πετύχουμε αποστολή μηνυμάτων συντεταγμένων με σταθερό ρυθμό. Αρχικά, η συνθήκη είχε τη μορφή  $if t \geq x+N$ , όπου N ένας τυχαίος αριθμός ms. Αυτό που κάναμε αρχικά ήταν να μεταφερθούμε με την συσκευή μας σε ένα μέρος που έχει πολύ καλή κάλυψη δικτύου τηλεφωνίας προκειμένου να αποφύγουμε τέτοιες παρεμβολές. Τότε, τρέξαμε τον Τελικό Κώδικα Γ', δηλαδή τον κώδικα του προηγούμενου υποκεφαλαίου, προκειμένου να βγάλουμε έναν μέσο όρο για τον χρόνο μεταξύ δύο διαδοχικών sms συντεταγμένων. Μετά από αρκετές εκτελέσεις του κώδικα προέκυψε ότι κατά μέσο όρο ο χρόνος που μεσολαβεί μεταξύ δύο διαδοχικών sms ήταν ίσος με 2,7 sec. Κατά τις διαδοχικές εκτελέσεις παρατηρήσαμε διαδοχική αποστολή sms που κυμαινόταν από 1,8 sec μέχρι και 5,3 sec υπό καλή κάλυψη δικτύου όπως είπαμε.

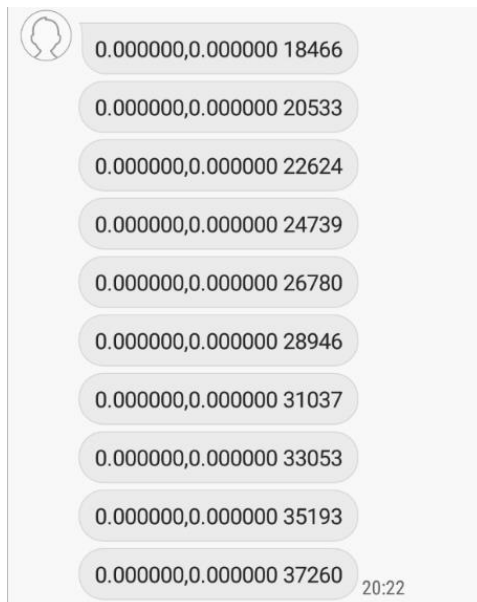
Σε αυτή τη φάση, λοιπόν, θέλησαμε να βρούμε την κατάλληλη συχνότητα δειγματοληψίας προκειμένου να πετύχουμε σταθερό ρυθμό αποστολής. Και παράλληλα, επειδή βασικός μας στόχος είναι και η υψηλή ταχύτητα αποστολής sms συντεταγμένων, έπρεπε ουσιαστικά να βρούμε την ελάχιστη συχνότητα δειγματοληψίας, καθώς έτσι θα πετυχαίναμε τον ταχύτερο σταθερό ρυθμό. Έτσι αφού λάβαμε τα αποτελέσματα αυτά, συνεχίσαμε με τον Τελικό Κώδικα Δ' που γράψαμε παραπάνω. Χρησιμοποιώντας την συνθήκη  $if t \geq x+N$ , κάναμε δοκιμές για την τιμή του N, προκειμένου να βρούμε την επιθυμητή συχνότητα. Ξεκινώντας από τον μέσο όρο που βρήκαμε, θέσαμε όπου N το 2700. Πράγματι αυτό που πετύχαμε είναι χρονικά σταθερή αποστολή διαδοχικών sms συντεταγμένων και ίση με 2,7sec. Στη συνέχεια μειώναμε κάθε φορά κατά 0.1sec αυτό το νούμερο μέχρι να βρούμε την ελάχιστη συχνότητα δειγματοληψίας. Όταν φτάσαμε στο σημείο να δοκιμάσουμε

N=1900, παρατηρήσαμε, ότι η συσκευή μας δεν ανταποκρινόταν καλά στο να κρατά σταθερό ρυθμό αποστολής, καθώς στο διάστημα αυτό δεν προλάβαινε να λάβει στίγμα και να στείλει τα στίγματα της τοποθεσίας που λάμβανε, με αποτέλεσμα να μην γίνεται η δειγματοληψία.

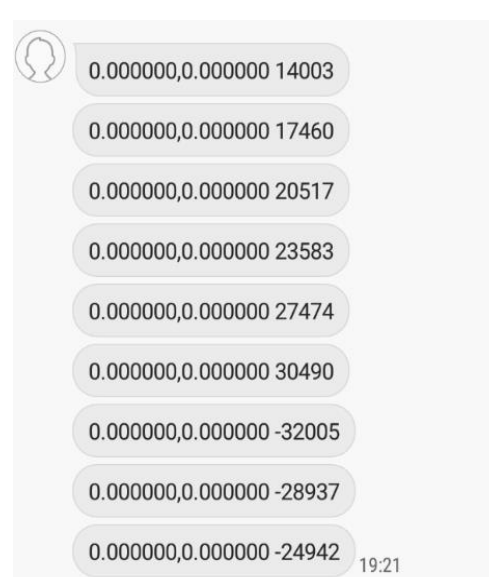
Έτσι καταλήξαμε στο ότι η ελάχιστη συχνότητα δειγματοληψίας για την οποία πετυχαίναμε σταθερό και βέλτιστο ρυθμό αποστολής διαδοχικών sms συντεταγμένων είναι τα 2 sec, δηλαδή  $f = 1/20 = 0.05$  Hz και N=2000. Για το λόγο αυτό στον παραπάνω κώδικα φαίνεται στην συνθήκη δειγματοληψίας η τιμή 2000.

Παραθέτουμε φωτογραφίες από τα παραπάνω αποτελέσματα. Να σημειωθεί ότι η μέτρηση έγινε σε χώρο όπου το GPS δεν μπορούσε να βρει σήμα, γι' αυτό και οι συντεταγμένες είναι μηδέν.

Για N=2000



Για N=1900

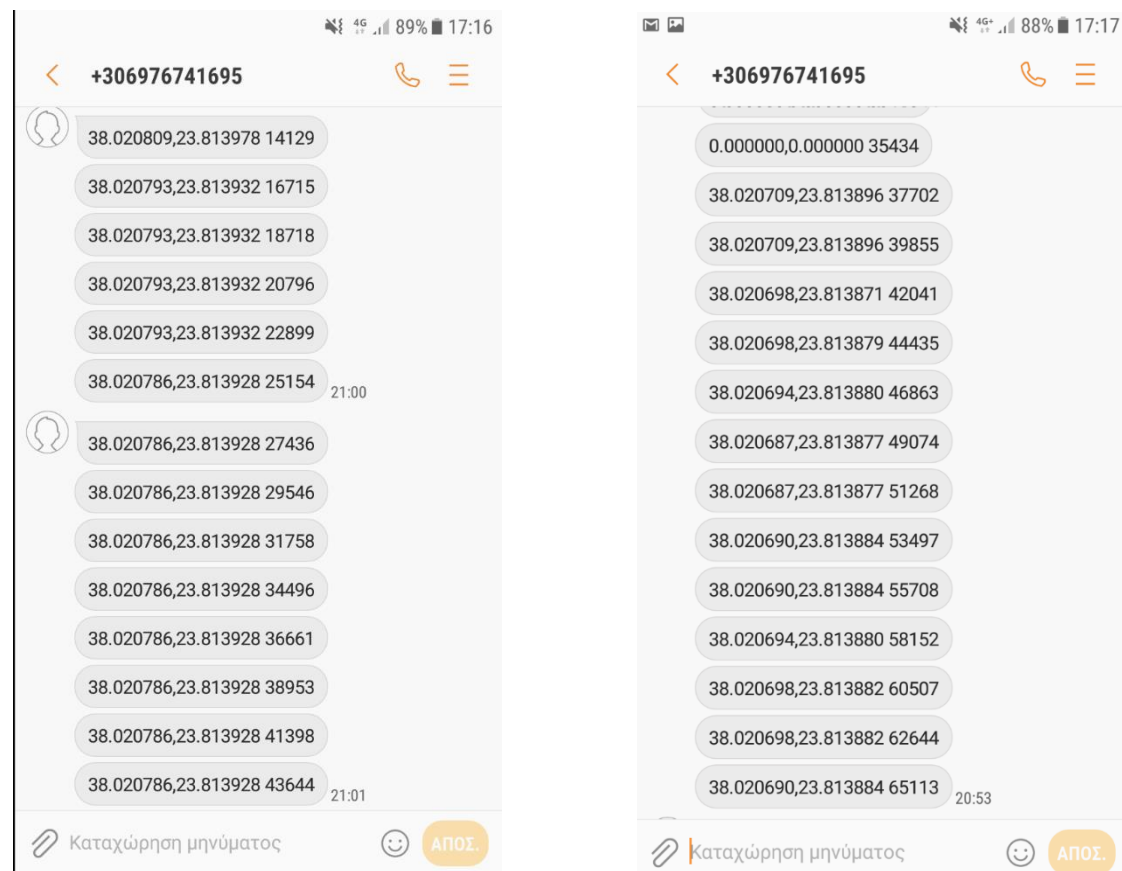


Βλέπουμε ότι για N=2000, δηλαδή 2sec, έχουμε σταθερό ρυθμό αποστολής δεδομένων που είναι κάθε φορά περίπου ίσος με 2 sec. Αντίθετα, όταν κατεβάσαμε λίγο ακόμα το N σε 1900, δηλαδή 1,9sec, παρατηρούμε ότι τα αποτελέσματα που προκύπτουν ούτε σταθερό ρυθμό έχουν, ούτε είναι λογικά, καθώς προκύπτουν αρνητικοί χρόνοι.

Άρα για N=2000 έχουμε τον ταχύτερο και σταθερό ρυθμό αποστολής.

Στην συνέχεια βγήκαμε στο δρόμο με το αυτοκίνητο και εκτελέσαμε τον κώδικα που υλοποιήσαμε για να ελέγξουμε την λειτουργία του σε κανονικές συνθήκες, που προφανώς περιμένουμε να υπάρχουν αποκλίσεις, καθώς το δίκτυο τηλεφωνίας δεν είναι εξίσου καλό σε όλα τα σημεία.

Ακολουθώντας, λοιπόν, μια διαδρομή στην περιοχή του Χαλανδρίου λάβαμε τις ακόλουθες μετρήσεις.



Όπως μπορούμε πολύ εύκολα να παρατηρήσουμε, από τις παραπάνω μετρήσεις που κάναμε, έχουμε πετύχει αποστολή διαδοχικών sms συντεταγμένων με σταθερό ρυθμό, περίπου ίσο με 2 sec. Προφανώς υπάρχουν και πολύ μικρές αποκλίσεις λόγω της κάλυψης του δικτύου. Εν συνεχεία ελέγξαμε τις συντεταγμένες που μας επέστρεψε το GPS με το Google Maps και το αποτέλεσμα ήταν σωστό.

Άρα, με τον τρόπο αυτό η συσκευή μας είναι σε θέση όχι μόνο απλά να μας στείλει διαδοχικά sms συντεταγμένων, αλλά να το κάνει αυτό με σταθερό και βέλτιστο ρυθμό.

#### 4.6 Χρονικά Ελεγχόμενη και Συνεχής Αποστολή Ομάδας Στιγμάτων Τοποθεσίας

Στο σημείο αυτό ουσιαστικά θέλαμε να επεκτείνουμε το θέμα του προηγούμενου υποκεφαλαίου και να πετύχουμε αυτή τη φορά διαδοχική αποστολή sms ομάδας συντεταγμένων με σταθερό ρυθμό. Προφανώς αξιοποιούμε το πόρισμα του προηγούμενου υποκεφαλαίου για τη συνθήκη δειγματοληψίας, δηλαδή  $N=2000$ , καθώς βρίκαμε πως για την τιμή αυτή έχουμε την μέγιστη ταχύτητα αποστολής μηνυμάτων. Όμως, όπως είναι λογικό ο χρόνος αποστολής μεταξύ δύο διαδοχικών ομάδων συντεταγμένων δεν αναμένεται να είναι 2 sec, καθώς τώρα μιλάμε για ομάδα

συντεταγμένων που σημαίνει πρώτα αποθήκευση μερικών και στη συνέχεια αποστολή αυτών. Ας δούμε όμως καλύτερα τον αλγόριθμο:

#### Συνολικός Αλγόριθμος Λειτουργίας του Ε' Τελικού Κώδικα

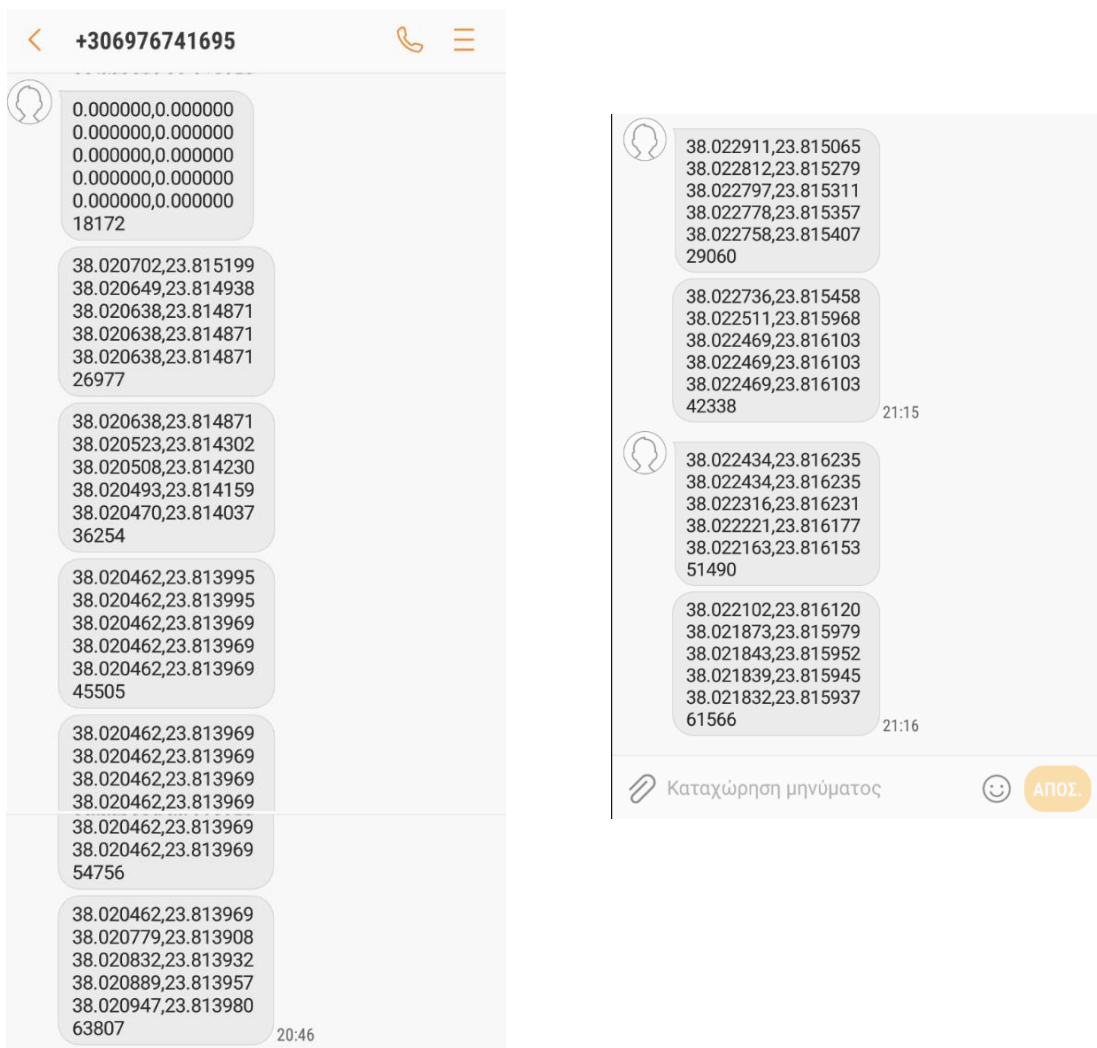
- **Input:** -
- **Output:** message
- Αναμονή να πιάσει στίγμα το GPS
- Ρωτάμε το Arduino το χρόνο μέχρι τώρα και τον αποθηκεύουμε στην μεταβλητή t
- Αποθήκευσε τις συντεταγμένες που έπιασε το GPS στη μεταβλητή txtmsg
- Εάν το μήκος του txtmsg είναι μεγαλύτερο από 80 και παράλληλα  $t \geq x+2000$ , τότε ρωτά ξανά για την ώρα του Arduino και αποθήκευσε την αυτή τη φορά στην μεταβλητή x.
  - Πρόσθεσε στις συντεταγμένες από κάτω τη μεταβλητή x του χρόνου
  - Στείλε μήνυμα με τις συντεταγμένες και το χρόνο που έχουν αποθηκευτεί στον buffer, δηλαδή message=TRUE. Στη συνέχεια ετοιμάσου για νέο μήνυμα, άρα άδειασε τον buffer της μνήμης
- Εάν το μήκος του txtmsg < 80 ή ο χρόνος  $t < x+2000$  τότε:
  - Καθυστέρησε 1sec
  - Επανάληψη

Σε αυτό το σημείο πρέπει να εξηγήσουμε λίγο τον παραπάνω αλγόριθμο. Βλέπουμε ότι έχει πολλά κοινά με τον προηγούμενο αλγόριθμο, καθώς επιτελεί ουσιαστικά την ίδια λειτουργία, με την διαφορά ότι τώρα έχουμε αποστολή sms ομάδας συντεταγμένων. Όπως έχουμε πει, κάθε συντεταγμένη που αποθηκεύεται αποτελείται από 19 χαρακτήρες (9 η κάθε συντεταγμένη x και y και 1 το κόμμα που τις διαχωρίζει). Εμείς εδώ επιλέξαμε να στείλουμε ομάδα συντεταγμένων προκειμένου να πραγματοποιηθεί πρακτική δοκιμή για την εύρεση της μέγιστης ταχύτητας αποστολής ομάδας μηνυμάτων από τον Arduino και σωστής αποδοχής των μηνυμάτων αυτών από το κινητό ή τον οποιοδήποτε άλλο δέκτη. Εκτελέσαμε δοκιμαστική αποστολή μηνυμάτων σε ομάδες με πλήθος διαδοχικά αυξανόμενο δοκιμάζοντας 5, 10 και 20 συντεταγμένες αντίστοιχα.

Στον παραπάνω αλγόριθμο εξηγείται η δοκιμή για ομάδα των 5 συντεταγμένων. Συνεπώς,  $(5 \text{ συντεταγμένες}) * (19 \text{ χαρακτήρες}) = 95 \text{ χαρακτήρες}$ . Για το λόγο αυτό εμείς έχουμε βάλει όριο μήκους txtmsg τους 80 χαρακτήρες. Έτσι κάθε φορά που θα γράφεται η 5<sup>η</sup> συντεταγμένη και αν προφανώς ισχύει η κλασική συνθήκη δειγματοληψίας  $t \geq x+2000$ , με την οποία πετυχαίνουμε τον ταχύτερο σταθερό ρυθμό αποστολής, τότε ο αλγόριθμός μας θα στείλει μήνυμα με την ομάδα των 5 αυτών συντεταγμένων. Αντίστοιχα για τις 10 συντεταγμένες θέσαμε όριο αντί για 80 το 180, διότι  $(10 \text{ συντεταγμένες}) * (19 \text{ χαρακτήρες}) = 190$ . Προφανώς και σε αυτή την περίπτωση η συνθήκη δειγματοληψίας είναι και πάλι η ίδια, δηλαδή  $N=2000$ . Ομοίως, όταν δοκιμάσαμε αποστολή 20 συντεταγμένων καθορίσαμε το όριο στα 370, καθώς  $(20 \text{ συντεταγμένες}) * (19 \text{ χαρακτήρες}) = 380$ .

Στο σημείο αυτό αξίζει να διευκρινίσουμε το ότι έχουμε θέσει και μία καθυστέρηση ενός δευτερολέπτου, διότι πρέπει να δίνουμε χρόνο στο GPS να ανανεώσει το σήμα του, αν αλλάξουμε θέση. Το GPS που χρησιμοποιούμε, ενημερώνεται κάθε 1 sec. Για το λόγο αυτό κάθε φορά που αποθηκεύει τις συντεταγμένες, το βάζουμε να καθυστερεί κατά 1 sec προκειμένου άμα κινούμαστε να καταφέρει να ενημερωθεί με την νέα τοποθεσία και να μην στείλει ξανά την παλιά.

Δοκιμάζοντας, λοιπόν, τον παραπάνω κώδικα, θελήσαμε να δούμε και στην πράξη, το πόσο γρήγορα μπορεί η συσκευή μας να στείλει sms ομάδας συντεταγμένων, αρχικά αποτελούμενη από 5 συντεταγμένες. Κάνοντας δοκιμές με το αυτοκίνητο βγάλαμε τα εξής αποτελέσματα που φαίνονται παρακάτω:



Όπως μπορούμε να δούμε από τα παραπάνω αποτελέσματα, η συσκευή μας είναι σε θέση να στέλνει sms ομάδας συντεταγμένων των πέντε σε χρόνο 9 sec. Άρα μπορούμε να βγάλουμε το εξής πόρισμα: Αφού η μέγιστη ταχύτητα αποστολής είναι 2 sec, σημαίνει ότι χρειάζονται  $9-2=7$  sec για να συλλέξει και να αποθηκεύσει η συσκευή μας τις πέντε συντεταγμένες, συμπεριλαμβανομένης της καθυστέρησης 1

sec κάθε φορά. Προφανώς οι παραπάνω συντεταγμένες ελέγχθηκαν και επαληθεύτηκαν με τη βοήθεια του Google Maps.

Στη συνέχεια επιλέξαμε να γίνεται αποστολή ομάδας συντεταγμένων αποτελούμενη από 10 συντεταγμένες. Μετά από δοκιμή που έγινε στον δρόμο, είδαμε ότι ο μέσος χρόνος αποστολής ομάδας συντεταγμένων των δέκα ήταν ίσος με 19 sec. Το αποτέλεσμα αυτό ήταν πολύ λογικό καθώς όπως είπαμε χρειάζονται περίπου 7 sec για να συλλεχθούν και να αποθηκευτούν οι 5 συντεταγμένες, άρα για τις 10 θα χρειαζόταν ένας μέσος χρόνος ίσος με  $7*2=14$  sec. Επιπλέον έχουμε 2 αποστολές μηνυμάτων, καθώς οι 10 συντεταγμένες αντιστοιχούν σε  $10*19=190$  χαρακτήρες  $> 140 = 1$  sms. Συνεπώς η καθυστέρηση από τις 2 αποστολές είναι  $2sec*2=4sec$ . Άρα  $14sec + 4sec = 18$  sec, περιμέναμε να προκύψει, και το αποτέλεσμα της δοκιμής που κάναμε ήταν, όπως μπορούμε να δούμε και εδώ ίσο με 19sec.

```
38.020702,23.815199
38.020649,23.814938
38.020638,23.814871
38.020638,23.814871
38.020638,23.814871
38.022911,23.815065
38.022812,23.815279
38.022797,23.815311
38.022778,23.815357
38.022758,23.815407
38.022756,23.814809
38.022644,23.814823
38.022729,23.815062
23427
```

```
38.020702,23.813854
38.020958,23.813990
38.020954,23.813992
38.020954,23.813992
38.020950,23.813992
38.021858,23.814278
38.021858,23.814278
38.021858,23.814278
38.021671,23.814402
38.021675,23.814505
42552
```

```
38.020462,23.813969
38.020779,23.813908
38.020832,23.813932
38.020889,23.813957
38.020947,23.813980
38.022434,23.816235
38.022434,23.816235
38.022316,23.816231
38.022221,23.816177
38.022163,23.816153
61610
```

Άρα με αυτόν τον τρόπο επαληθεύσαμε τη λειτουργία της συσκευής μας, καθώς τα δύο νούμερα βρίσκονται πολύ κοντά μεταξύ τους και η μικρή απόκλιση του 1 δευτερολέπτου μπορεί να οφείλεται στην κάλυψη του τηλεφωνικού δικτύου ή σε πιθανές αποκλίσεις της συσκευής μας.

Εν συνεχεία κάναμε ακόμα μία δοκιμή για αποστολή ομάδας συντεταγμένων αποτελούμενη από 20 συντεταγμένες. Θεωρητικά, αυτό που περιμέναμε να προκύψει ήταν το εξής:

Αφού οι 5 συντεταγμένες χρειάζονται 7 sec για να ληφθούν και να αποθηκευτούν, τότε οι 20 θα χρειάζονται κατά προσέγγιση 28 sec. Παράλληλα, έχουμε συνολικά  $20 \cdot 19$  χαρακτήρες = 380 χαρακτήρες  $> 280(2 \cdot 140) = 3$  sms θα σταλούν. Άρα 3 αποστολές μηνυμάτων, κάθε μία από τις οποίες απαιτεί 2sec για να γίνει, άρα σύνολο 6sec. Άρα συνολικά,  $28\text{sec} + 6\text{sec} = 34\text{sec}$ .

Δοκιμάσαμε, λοιπόν, την συσκευή μας στην πράξη και παρατηρήσαμε ότι ο συνολικός μέσος χρόνος αποστολής sms ομάδας συντεταγμένων των είκοσι ήταν τα 37sec, αριθμός πολύ κοντινός στο αποτέλεσμα που περιμέναμε να προκύψει.

#### 4.7 Αποστολή Δεδομένων και Αποθήκευσή τους σε Η/Υ

Όπως είδαμε και στο προηγούμενο κεφάλαιο καταφέραμε να πετύχουμε να βλέπουμε τα sms που έρχονται στο κινητό μας, στον ηλεκτρονικό υπολογιστή μέσω του προγράμματος MobileGo. Στο σημείο αυτό δοκιμάσαμε να δούμε αν τα sms των συντεταγμένων που στέλνει η συσκευή μας και λαμβάνει το κινητό τηλέφωνο μπορεί να τα λάβει σωστά και ο υπολογιστής. Και αμέσως μετά θα ασχοληθούμε και με την αποθήκευση των μηνυμάτων αυτών σε μορφή txt στον Η/Υ.

Αρχικά, λοιπόν, ανοίξαμε το πρόγραμμα MobileGo στον υπολογιστή μας και συνδέσαμε το κινητό μας με καλώδιο USB. Κάνοντας κλικ στα sms ήμασταν σε θέση να δούμε όλα τα μηνύματα του κινητού τηλεφώνου. Κατά συνέπεια ήμασταν σε θέση να δούμε και τα sms των συντεταγμένων που είχαν σταλεί από τη συσκευή μας. Το αποτέλεσμα αυτό φαίνεται στην παρακάτω εικόνα:

The screenshot displays the MobileGo application interface. On the left, a sidebar shows navigation options like 'Galaxy A5 (2016)', 'Apps', 'Contacts', 'SMS' (2066), 'All Messages' (2066), 'Unread Messages' (0), 'Music', 'Photos', 'Videos', and 'Files'. The main area shows a list of messages with columns for 'Content' and 'Time'. One message is selected, showing its content: '+306976741695 (0/1) 38.022102,23.81103'. To the right, a detailed view of this message is shown, including the sender's phone number, the date and time (22/1/2018 9:15 μμ), and the message body containing several lines of coordinates (e.g., 38.022736, 23.815458).



Βλέπουμε, λοιπόν, ότι επιλέξαμε (βάζοντας τικ στο κουτάκι) τα μηνύματα που έρχονται από τον αριθμό της sim που βρίσκεται στην συσκευή μας. Το αποτέλεσμα βρίσκεται στην δεξιά στήλη, που όπως μπορούμε να δούμε είναι μηνύματα συντεταγμένων που μας έστειλε η συσκευή μας.

Ενδεικτικά παρουσιάζουμε ακόμα δύο φωτογραφίες που προέκυψαν κάνοντας scroll up στην δεξιά στήλη για να δούμε και άλλα μηνύματα συντεταγμένων που έστειλε η συσκευή μας.

The screenshot shows the Wondershare MobileGo interface for a Galaxy A5 (2016). The left sidebar lists various data types: Apps, Contacts, SMS (2066), All Messages (2066), Unread Messages (0), Music, Photos, Videos, and Files. The main area displays a list of messages with columns for Content and Time. One message is selected, showing its content and coordinates.

Content	Time
WHATS UP (0/6) ΜΑΣ ΤΟ ΖΗΤΗΣΑΤ...	26 Ιανουα...
KOTSOVOLOS (0/1) ΕΧΕΙΣ 3 ΜΕΡΕΣ Γ...	24 Ιανουα...
+306976741695 (( 38.022102,23.81...	22 Ιανουα...
+306981204453 (( ΨΑΧΝΩ ΝΑ ΠΑΡΚ...	22 Ιανουα...
+306948070970 (( ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	20 Ιανουα...
Giannis Politopoulos ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	19 Ιανουα...
WHATSUP (0/2) ΤΩΡΑ ΠΟΡΤΑΖΟΥ...	19 Ιανουα...
+306976857912 (( ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	14 Ιανουα...
Δημήτρης Rock (0/ ΚΑΛΗΣΠΕΡΑ ΔΗΜ...	13 Ιανουα...
Cosmote (0/43) ΕΧΕΤΕ 1 ΝΕΟ ΜΗ...	12 Ιανουα...
Mrampas (0/38) ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	10 Ιανουα...
CALLS (0/10)	09 Ιανουα...

The selected message content is: **To: +306976741695**. The message body contains several lines of coordinates: 38.022102, 23.813928, 22/1/2018 9:00 μμ, 38.020786, 23.813928 25154, 22/1/2018 9:01 μμ, 38.020786, 23.813928 27436, 22/1/2018 9:01 μμ, 38.020786, 23.813928 29546, 22/1/2018 9:01 μμ, 38.020786, 23.813928 31758, 22/1/2018 9:01 μμ, 38.020786, 23.813928 34496, 22/1/2018 9:01 μμ.

The screenshot shows the Wondershare MobileGo interface for a Galaxy A5 (2016). The left sidebar lists various data types: Apps, Contacts, SMS (2066), All Messages (2066), Unread Messages (0), Music, Photos, Videos, and Files. The main area displays a list of messages with columns for Content and Time. One message is selected, showing its content and coordinates.

Content	Time
WHATS UP (0/6) ΜΑΣ ΤΟ ΖΗΤΗΣΑΤ...	26 Ιανουα...
KOTSOVOLOS (0/1) ΕΧΕΙΣ 3 ΜΕΡΕΣ Γ...	24 Ιανουα...
+306976741695 (( 38.022102,23.81...	22 Ιανουα...
+306981204453 (( ΨΑΧΝΩ ΝΑ ΠΑΡΚ...	22 Ιανουα...
+306948070970 (( ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	20 Ιανουα...
Giannis Politopoulos ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	19 Ιανουα...
WHATSUP (0/2) ΤΩΡΑ ΠΟΡΤΑΖΟΥ...	19 Ιανουα...
+306976857912 (( ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	14 Ιανουα...
Δημήτρης Rock (0/ ΚΑΛΗΣΠΕΡΑ ΔΗΜ...	13 Ιανουα...
Cosmote (0/43) ΕΧΕΤΕ 1 ΝΕΟ ΜΗ...	12 Ιανουα...
Mrampas (0/38) ΕΔΧΑΤΕ 1 ΚΛΗΣΗ...	10 Ιανουα...
CALLS (0/10)	09 Ιανουα...

The selected message content is: **To: +306976741695**. The message body contains several lines of coordinates: 6/10/2017 7:14 μμ, 37.880714, 23.730003g, 37.877525, 23.731318g, 37.873483, 23.735023g, 37.873432, 23.735094g, 37.873432, 23.735094g, 37.864971, 23.744768g, 6/10/2017 7:16 μμ, 37.863338, 23.746613g, 37.863338, 23.746613g, 37.863338, 23.746613g, 37.863338, 23.746613g, 37.863338, 23.746613g, 37.863338, 23.746613g, 6/10/2017 7:20 μμ, 37.863338, 23.746613g, 37.863338, 23.746613g, 37.838902, 23.762529g, 37.838875, 23.762558g, 37.838875, 23.762558g, 37.838875, 23.762558g, 6/10/2017 7:25 μμ.

Όπως, μπορούμε να δούμε καθαρά, όποιον κώδικα από αυτούς που έχουμε και να εκτελέσουμε, είμαστε σε θέση ανά πάσα στιγμή να λάβουμε τα στίγματα τοποθεσίας, που στέλνει η συσκευή μας, και στον ηλεκτρονικό υπολογιστή.

Στο σημείο αυτό αξίζει να σημειωθεί το εξής: όπως είδαμε στην πρώτη λειτουργία του πρώτου κώδικα που εκτελέσαμε, για να μας στείλει η συσκευή μας sms με τις συντεταγμένες που έχει λάβει, έπρεπε εμείς να στείλουμε μήνυμα με το κινητό μας την λέξη “Location” στον αριθμό της κάρτας sim που έχει μέσα η συσκευή μας. Αυτό, όπως μπορούμε να δούμε καθαρά δεν γίνεται μόνο από το κινητό μας, αλλά μπορεί να γίνει και από τον υπολογιστή. Δηλαδή, αντί να στείλουμε την λέξη location από το κινητό μας στην συσκευή και αυτή να μας επιστρέψει πίσω στο κινητό τις συντεταγμένες, μπορούμε να στείλουμε την λέξη αυτή από τον Η/Υ και να μας επιστρέψει η συσκευή τις συντεταγμένες στον Η/Υ. Και αυτό γιατί, όπως βλέπουμε στην δεξιά στήλη κάτω κάτω στις παραπάνω φωτογραφίες, υπάρχει ένα κενό πλαίσιο με την λέξη send για να στείλουμε μήνυμα. Δηλαδή το πρόγραμμα αυτό (MobileGo) επιτρέπει την αμφίδρομη επικοινωνία, καθώς εκτός από το να λάβουμε sms μπορούμε κάλλιστα και να στείλουμε.

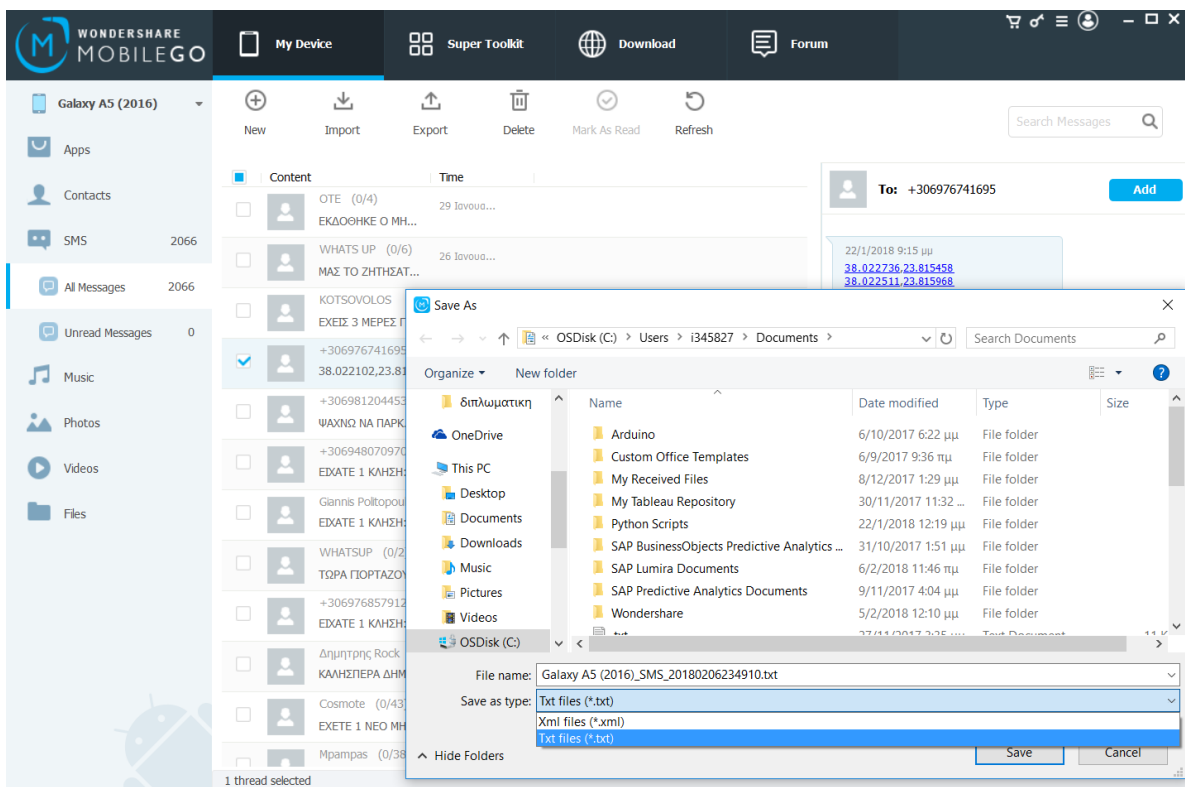
Στο σημείο αυτό, λοιπόν, έχουμε πετύχει την αποστολή δεδομένων της συσκευής μας στον Ηλεκτρονικό Υπολογιστή. Τώρα αυτό που μας έμεινε είναι η αποθήκευση των sms αυτών στον υπολογιστή σε μορφή txt. Αυτό επιτυγχάνεται πολύ εύκολα με το πρόγραμμα MobileGo, καθώς το μόνο που έχουμε να κάνουμε είναι το εξής:

Επιλέγουμε το μήνυμα με τον αριθμό της κάρτας sim που έχει η συσκευή μας, κάνοντας τικ στο κουτάκι που βρίσκεται στα αριστερά, όπως ακριβώς και πριν. Προφανώς στην δεξιά στήλη εμφανίζονται όλα τα μηνύματα συντεταγμένεων που έχει στείλει η συσκευή μας (όλη η συνομιλία δηλαδή). Στη συνέχεια πατάμε το κουμπί “Export” που βρίσκεται στην πάνω γραμμή (όπως φαίνεται παρακάτω):

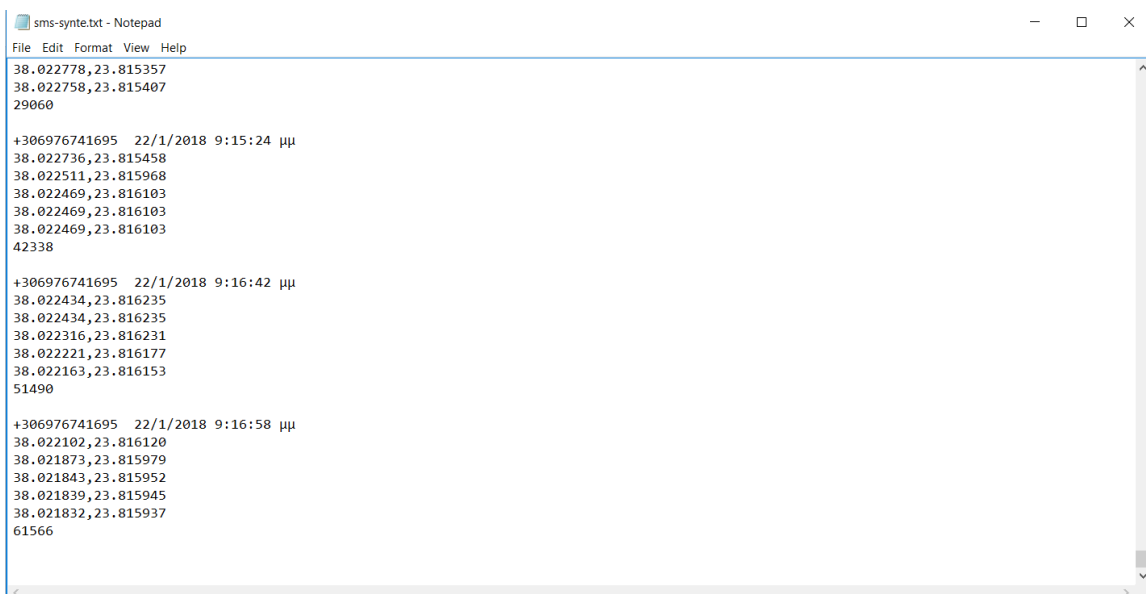
The screenshot shows the MobileGo application interface. On the left, there is a sidebar with navigation options: Galaxy A5 (2016), Apps, Contacts, SMS (2066), All Messages (2066), Unread Messages (0), Music, Photos, Videos, and Files. The main area displays a list of messages with columns for Content and Time. One message is selected, and the 'Export' button is highlighted. The right panel shows a detailed view of the selected message, including the recipient's phone number (+306976741695) and the message content, which consists of a list of phone numbers.

Content	Time
ΟΤΕ (0/4) ΕΚΔΟΗΚΕ Ο ΜΗ...	29 Ιανου...
WHATS UP (0/6) ΜΑΣ ΤΟ ΖΗΤΗΣΑΤ...	26 Ιανου...
ΚΟΤΣΟΒΟΛΟΣ (0/1) ΕΧΕΙΣ 3 ΜΕΡΕΣ ΓΙ...	24 Ιανου...
+306976741695 (1) 38.022102,23.81...	22 Ιανου...
+306981204453 (1) ΨΑΧΝΩ ΝΑ ΠΑΡΚ...	22 Ιανου...
+306948070970 (1) ΕΙΧΑΤΕ 1 ΚΗΣΗΝ...	20 Ιανου...
Giannis Politopoulos ΕΙΧΑΤΕ 1 ΚΗΣΗΝ...	19 Ιανου...
WHATSUP (0/2) ΤΩΡΑ ΠΟΡΤΑΖΟΥ...	19 Ιανου...
+306976857912 (1) ΕΙΧΑΤΕ 1 ΚΗΣΗΝ...	14 Ιανου...
Δημητρης Rock (0) ΚΑΛΗΣΠΕΡΑ ΔΗΜ...	13 Ιανου...
Cosmote (0/43) ΕΧΕΤΕ 1 ΝΕΟ ΜΗ...	12 Ιανου...
Mrampas (0/38) ...	10 Ιανου...

Και στη συνέχεια σώζουμε (κάνουμε export) όλα τα sms συντεταγμένων σε αρχείο είτε xml είτε txt. Προφανώς επιλέγουμε να τα σώσουμε σε μορφή txt, καθώς είναι πολύ πιο ευανάγνωστη για τον χρήστη. Ας δούμε και την αντίστοιχη φωτογραφία:



Έτσι, πηγαίνουμε στον υπολογιστή μας και βρίσκουμε το αρχείο txt με τα sms συντεταγμένων που μόλις σώσαμε, και το επιλέγουμε για να δούμε τη μορφή των στιγμάτων τοποθεσίας. Το αποτέλεσμα που προέκυψε προφανώς ήταν ακριβώς αυτό που θέλαμε. Είχαμε λάβει και αποθηκεύσει δηλαδή όλα τα sms συντεταγμένων που μας έστειλε η συσκευή μας, στον Η/Υ. Παρακάτω παραθέτουμε μερικές εικόνες ενδεικτικά:



```
sms-synte.txt - Notepad
File Edit Format View Help
38.020809,23.813978 14129

+306976741695 22/1/2018 9:00:51 μμ
38.020793,23.813932 16715

+306976741695 22/1/2018 9:00:53 μμ
38.020793,23.813932 18718

+306976741695 22/1/2018 9:00:55 μμ
38.020793,23.813932 20796

+306976741695 22/1/2018 9:00:57 μμ
38.020793,23.813932 22899

+306976741695 22/1/2018 9:00:59 μμ
38.020786,23.813928 25154

+306976741695 22/1/2018 9:01:01 μμ
38.020786,23.813928 27436

+306976741695 22/1/2018 9:01:04 μμ
38.020786,23.813928 29546

+306976741695 22/1/2018 9:01:07 μμ
38.020786,23.813928 31758

+306976741695 22/1/2018 9:01:09 μμ
38.020786,23.813928 34496
```

```
sms-synte.txt - Notepad
File Edit Format View Help
37.900826,23.719734y
37.900826,23.719734y
37.900826,23.719734g
37.890781,23.723400y
37.889786,23.724098y
37.883644,23.728054g

+306976741695 6/10/2017 7:14:12 μμ
37.880714,23.730003g
37.877525,23.731318g
37.873493,23.735023g
37.873432,23.735094y
37.873432,23.735094y
37.864971,23.744768y

+306976741695 6/10/2017 7:16:58 μμ
37.863338,23.746613g
37.863338,23.746613y
37.863338,23.746613y
37.863338,23.746613y
37.863338,23.746613g
37.863338,23.746613g

+306976741695 6/10/2017 7:20:11 μμ
37.863338,23.746613y
37.863338,23.746613g
37.838902,23.762529g
37.838875,23.762558y
37.838875,23.762558g
```

Όπως μπορούμε να δούμε, αποθηκεύσαμε όλες τις συντεταγμένες που μας έστειλε η συσκευή μας, στον υπολογιστή. Επιπλέον, όπως εξηγήσαμε αναλυτικά, ο τρόπος επικοινωνίας αυτός, με τη βοήθεια δηλαδή του προγράμματος MobileGo, συνδυάζει ταχύτητα, απλότητα και μηδενικά επιπλέον έξοδα.

#### 4.8 Πακέτο Μηνυμάτων

Όπως είπαμε στην αρχή, ένας από τους πρώτους στόχους μας ήταν η συσκευή αυτή να είναι όσο το δυνατόν πιο οικονομική γίνεται. Για το λόγο αυτό φροντίσαμε ώστε να επιλέξουμε τα κατάλληλα υλικά που προσφέρουν τον ιδανικό συνδυασμό υψηλής αποτελεσματικότητας και χαμηλής τιμής. Με βάση μια ενδελεχή έρευνα αγοράς που εκτελέσαμε καταλήξαμε πραγματικά στον ιδανικό αυτό συνδυασμό με τις συσκευές που επιλέξαμε.

Στο σημείο αυτό, όμως, για να μπορέσει η συσκευή μας να γίνει ακόμα πιο προσητή στην αγορά, προκειμένου να μπορεί να αγοραστεί από τον καθένα και όχι μόνο από το συγκοινωνιακό δίκτυο, κάναμε μια προσπάθεια να μειώσουμε ακόμα περισσότερο την τιμή της συσκευής. Αυτή, λοιπόν, την φορά δεν ασχοληθήκαμε με τις τιμές των υλικών, καθώς έγινε εκτεταμένη ανάλυση για αυτές, αλλά με την τιμή του γραπτού SMS που προσφέρουν οι εταιρίες τηλεπικοινωνιών της Ελλάδος. Παρατηρήσαμε ότι η τιμή του SMS τόσο για την εταιρία Cosmote όσο και για την εταιρία Vodafone κοστίζει 0,25 ευρώ ανεξαρτήτως δικτύου, τιμή αρκετά υψηλή που δυσκολεύει την άμεση παραγωγή τέτοιων συσκευών. Επομένως έπρεπε να βρούμε ένα τρόπο να μειώσουμε την τιμή του SMS, ώστε να περιορίσουμε το κόστος αυτό κατά πάρα πολύ.

Συνεπώς, ήρθαμε σε επικοινωνία με τους τεχνικούς των δύο παραπάνω εταιριών κινητής τηλεφωνίας προκειμένου να δούμε τι μπορεί να γίνει με τις τιμές των SMS. Στο σημείο αυτό αξίζει να αναφερθεί ότι ήρθαμε σε επαφή μόνο με τις συγκεκριμένες δύο εταιρίες, καθώς είναι αυτές που έχουν την μεγαλύτερη κάλυψη σήματος σε όλη την Ελλάδα και όπως είπαμε εκτός από την τιμή μας ενδιαφέρει ιδιαίτερα και η αποτελεσματικότητα της συσκευής.

Πρώτα ήρθαμε σε επαφή με την εταιρία Vodafone, και προφασιζόμενοι ότι έχουμε μια start-up εταιρία που θέλει να βγάλει τη συγκεκριμένη συσκευή στην αγορά, ζητήσαμε αν έχει να μας δώσει κάποιο καλό πακέτο με το οποίο θα είχαμε πολύ μεγάλο αριθμό SMS με πολύ χαμηλό κόστος. Το μόνο που μπορούσε να μας δώσει είναι ένα πακέτο μηνυμάτων που με 3 ευρώ έχεις 100 μηνύματα, το οποίο μεταφράζεται σε 0,03 ευρώ το SMS, που είναι μία καλή προσφορά, αλλά θα μπορούσαμε σίγουρα να βρούμε και καλύτερη.

Για το λόγο αυτό καλέσαμε αμέσως μετά την Cosmote με την ίδια ακριβώς πρόταση και πρόταση προκειμένου να δούμε τι ακριβώς μπορεί να μας προσφέρει. Εδώ μετά από αρκετές φορές επικοινωνίας καταλήξαμε στην καλύτερη επιλογή που ήταν ένα πάγιο αξίας 5 ευρώ το οποίο μας έδινε 1500 SMS, προσφορά που μειώνει το κόστος σε πολύ μεγάλο βαθμό. Άρα, χρησιμοποιώντας κάρτες Cosmote για την συσκευή μας, λόγω του μειωμένου κόστους και της υψηλής αποτελεσματικότητας που προσφέρει, έχουμε μια αποτελεσματική και πολύ οικονομική συσκευή.

## ΚΕΦΑΛΑΙΟ 5 – ΑΠΟΤΕΛΕΣΜΑΤΑ

Στο κεφάλαιο αυτό θα αναφερθούμε στα αποτελέσματα που προέκυψαν από διάφορες δοκιμές που κάναμε προκειμένου να διαπιστώσουμε την λειτουργικότητα και την αποτελεσματικότητα της συσκευής που φτιάξαμε.

### 5.1 Αποτελέσματα από Α' και Β' Κώδικα

Στο σημείο αυτό αφού έχουμε υλοποιήσει και τις δύο λειτουργίες του συγκεκριμένου κώδικα, έπρεπε να ελέγξουμε την σωστή λειτουργία της συσκευής μας. Κάνοντας, όπως είπαμε και πριν μία βόλτα με το αυτοκίνητο μας στην Αθήνα, πήραμε στίγματα από πολλές και διαφορετικές περιοχές είτε πατώντας κάποιο εκ των δύο κουμπιών είτε στέλνοντας με μήνυμα τη λέξη “location”. Κάθε φορά, που είτε πατούσαμε κάποιο κουμπί είτε στέλναμε το “location”, παράλληλα ανοίγαμε τους χάρτες Google και αποθηκεύαμε τις αντίστοιχες συντεταγμένες. Αυτό αποσκοπούσε στον έλεγχο του κατά πόσο οι συντεταγμένες που μας έδινε η συσκευή μας ταυτίζονταν σε μεγάλο βαθμό με αυτές που μας έδινε το Google Maps.

Το αποτέλεσμα ήταν πραγματικά εντυπωσιακό, καθώς επαληθεύσαμε όλες τις συντεταγμένες που λάβαμε κατά τη διαδρομή και διαπιστώσαμε ότι η συσκευή μας είχε πολύ μεγάλη ακρίβεια, καθώς τα αποτελέσματα που μας έδινε ήταν ίδια με αυτά του Google Maps. Μόνο ελάχιστες φορές είχαμε ορισμένες πολύ μικρές αποκλίσεις από την πραγματικότητα γεγονός που φαινόταν στο τελευταίο δεκαδικό ψηφίο των συντεταγμένων. Επιπλέον, αξίζει να σημειωθεί ότι κατά την διαδρομή στην Αθήνα σταματήσαμε επίτηδες και σε σημεία με πολλά δέντρα από πάνω ή ψηλά κτήρια για να διαπιστώσουμε τι συμβαίνει με την ακρίβεια της συσκευής μας όταν οι συνθήκες δεν είναι και οι ιδανικές. Το αποτέλεσμα και εδώ ήταν πολύ καλό, καθώς και πάλι οι συντεταγμένες που μας έδινε η συσκευή μας, μετά από επαλήθευση τους στο Google Maps ήταν πολύ ακριβείς.

Τέλος, κατά τη διαδρομή, θέσαμε σε λειτουργία και τον Β' Κώδικα που τρέχει αέναα και αποθηκεύει συντεταγμένες και στη συνέχεια τις στέλνει σε μήνυμα από μόνο του σε ομάδες. Και σε αυτή την περίπτωση οι συντεταγμένες που μας στέλνονταν σε μήνυμα από την συσκευή μας ήταν απόλυτα ακριβείς, γεγονός που διαπιστώσαμε μετά από επαλήθευση από τους χάρτες του Google. Επομένως, η συσκευή μας εκτός από το να στέλνει τα στίγματα τοποθεσίας, είναι σε θέση να και να αποθηκεύει ομάδα συντεταγμένων, ώστε να στέλνει τα στίγματα σε πακέτα. Με αυτόν τον τρόπο έχουμε στα χέρια μας μια συσκευή, η οποία όχι μόνο θα μας στέλνει την τοποθεσία της, αλλά θα μας στέλνει ακριβώς και τη συγκεκριμένη διαδρομή που ακολούθησε για να φτάσει σε μία τοποθεσία. Είμαστε σε θέση δηλαδή να γνωρίζουμε αν το όχημα που φέρει πάνω τη συσκευή μας, πέρασε ή όχι από συγκεκριμένα σημεία, γεγονός απόλυτα χρήσιμο.

Τέλος, διαπιστώσαμε ότι η συσκευή μας δουλεύει πολύ αποτελεσματικά και μάλιστα με πολύ μεγάλη ακρίβεια τόσο στις ιδανικές όσο και στις πιο δυσμενείς συνθήκες ορατότητας και δικτύου τηλεφωνίας. Επιπλέον, λόγω και της σωστής λειτουργίας του

δεύτερου κώδικα, διαπιστώσαμε ότι η συσκευή μας είναι και απόλυτα αυτόνομη και ευέλικτη, καθώς είναι σε θέση να μας στέλνει την τοποθεσία ανά πάσα στιγμή χωρίς την δική μας παρέμβαση.

### 5.2 Αποτελέσματα Γ', Δ' και Ε' Κώδικα

Στο σημείο αυτό, αφού σιγουρέψαμε τα παραπάνω αποτελέσματα, βάλαμε στο παιχνίδι τον χρόνο με τη βοήθεια του ρολογιού του Arduino. Αυτό που καταφέραμε να υλοποιήσουμε είναι η κατασκευή μια ιδιαίτερα σθεναρής συσκευής, η οποία είναι σε θέση να μας στέλνει στίγμα τοποθεσίας με μεγάλη ταχύτητα και σταθερό ρυθμό. Αποδείξαμε δηλαδή ότι η συσκευή μας εκτός από το ότι μπορεί να είναι αυτόνομη, είναι και σε θέση να μας επιστρέφει την τοποθεσία με τον ρυθμό που θα επιλέξουμε εμείς είτε θέλουμε με γρήγορη ταχύτητα είτε με πιο αργή. Επιπλέον, μπορεί να μας επιστρέφει και στίγματα τοποθεσίας που στέλνονται ανά ομάδες, πάλι με τον ρυθμό που θέλουμε εμείς να έχουμε. Συνεπώς, εξήγαμε το συμπέρασμα ότι πρόκειται για μια ιδιαίτερα ευέλικτη συσκευή που είναι σε θέση να εξυπηρετήσει τις ανάγκες του ανθρώπου και να προσαρμοστεί εύκολα σε αυτές. Τέλος, η απόδειξη ότι μπορεί να στέλνει συνεχόμενα και αέναα την τοποθεσία με χρόνο αποστολής ίσο με 2 sec, κάνει την συσκευή μας ιδιαίτερα αποτελεσματική, καθώς έτσι έχουμε ανά πάσα στιγμή πλήρη έλεγχο της θέσης που βρίσκεται ο κινούμενος στόχος. Συνεπώς σε περίπτωση μη αναμενόμενων γεγονότων, με τη συσκευή αυτή μπορούμε να διαπιστώσουμε πάρα πολύ γρήγορα το πρόβλημα που έχει συμβεί, με αποτέλεσμα να δράσουμε άμεσα για την επίλυσή του.

### 5.3 Οικονομική Συσκευή – Χαμηλή Παροχή Ισχύος

Παράλληλα, η συσκευή που υλοποιήσαμε, εκτός από τα υπόλοιπα προτερήματα που αναφέραμε, είναι και ιδιαιτέρως οικονομική. Τόσο από άποψη υλικών όσο και από άποψη χρέωσης γραπτού SMS, μετά από εκτενή έρευνα καταλήξαμε σε έναν μοναδικό συνδυασμό. Ειδικά μετά την ενεργοποίηση του πακέτου της Cosmote με τα μηνύματα, έχουμε στα χέρια μας μια συσκευή που είναι σε θέση να μας δίνει στίγμα τοποθεσίας με πολύ μεγάλη ακρίβεια ανά πάσα στιγμή, και όλη αυτή η διάταξη να είναι ιδιαίτερα οικονομική, ώστε να μπορεί να αγοραστεί και από τον μέσο άνθρωπο, προκειμένου να του εξυπηρετήσει τις ανάγκες του. Παράλληλα, λόγω της χαμηλής του τιμής, το σύστημά μας είναι σε θέση να αγοραστεί από τα κοινωνικά δίκτυα, που απαιτούν μεγάλες ποσότητες αγοράς για τα πολλά και διάφορα μέσα μεταφοράς που υπάρχουν.

Τέλος, αξίζει να σημειωθεί ότι η συνολική διάταξη που υλοποιήσαμε απαιτεί πολύ χαμηλή παροχή ισχύος. Αυτό το επιτύχαμε, καθώς σε κάθε υλικό που χρησιμοποιήσαμε, πριν το αγοράσουμε, ελέγξαμε τις προδιαγραφές του όσον αφορά την τάση τροφοδοσίας και το ρεύμα που απαιτείται. Έτσι, συνδέοντας απλά την συσκευή μας με μια μπαταρία 5V, αυτή είναι σε θέση να λειτουργήσει απόλυτα αποτελεσματικά, αφού της φορτωθούν πρώτα τα αντίστοιχα προγράμματα. Έτσι με αυτόν τον τρόπο εξοικονομούμε πάλι χρήματα, καθώς υψηλή παροχή ισχύος θα σήμαινε αυτόματα και περισσότερα έξοδα, γεγονός που θέλαμε να αποφύγουμε.

## ΚΕΦΑΛΑΙΟ 6 – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Στο κεφάλαιο αυτό θα παρουσιάσουμε τις μελλοντικές επεκτάσεις που μπορούν να υλοποιηθούν με βάση την συγκεκριμένη συσκευή που υλοποιήσαμε, προκειμένου να εξυπηρετούνται καλύτερα οι ανάγκες των ανθρώπων.

Όπως, είπαμε και παραπάνω η συσκευή αυτή υλοποιήθηκε καθώς παρατηρήθηκε ένα κενό στην αγορά από το δίκτυο συγκοινωνιών. Συγκεκριμένα η συσκευή αυτή μπορεί να ενσωματωθεί στα διάφορα μέσα μεταφοράς που διασχίζουν όλη την Ελλάδα ώστε να παρακολουθούμε ανά πάσα στιγμή τα εμπορεύματα που μεταφέρουν. Και σε περίπτωση που κάποιος κλαπεί, μέσω της real-time επικοινωνίας θα μας έρχεται ειδοποίηση ότι κάποιος εμπόρευμα δεν ακολουθεί τον υπόλοιπο στόλο, άρα πιθανώς να έχουμε κλοπή, και συνεπώς θα μπορούμε να επέμβουμε άμεσα για την αποφυγή αυτής της κατάστασης. Οστόσο, όμως, δεν σημαίνει ότι η συσκευή αυτή δεν μπορεί να χρησιμοποιηθεί και από άλλους φορείς. Ίσα ίσα που λόγω της χαμηλής του τιμής και της υψηλής λειτουργικότητας που προσφέρει μπορεί να αξιοποιηθεί από πολλούς χρήστες ανεξαρτήτως επαγγέλματος.

### 6.1 Ενσωμάτωση Διαδραστικών Χαρτών

Μία πολύ σημαντική επέκταση της διπλωματικής αυτής αποτελεί η ενσωμάτωση διαδραστικών χαρτών στην συσκευή μας. Αυτό προϋποθέτει πρώτα απ' όλα τα στίγματα τοποθεσίας που λαμβάνει το GPS να στέλνονται μέσω GPRS σε ηλεκτρονικό υπολογιστή. Από εκεί οι πληροφορίες τοποθεσίας που λαμβάνονται από το GPS μεταφέρονται μέσω μιας σύνδεσης GPRS σε έναν απομακρυσμένο διακομιστή (server) μέσω μιας σύνδεσης TCP. Ο διακομιστής TCP αποθηκεύει τα εισερχόμενα δεδομένα θέσης σε μια βάση δεδομένων MySQL. Όταν ένας χρήστης επιλέξει τη σελίδα παρακολούθησης, ο Zope, ένας διακομιστής εφαρμογών ιστού ανοικτού κώδικα, ανοίγει μια σελίδα HTML με ενσωματωμένο κώδικα javascript. Το javascript θα τρέξει στο πρόγραμμα περιήγησης του χρήστη και έχει οδηγίες για ανάκτηση των πληροφοριών θέσης από τη βάση δεδομένων MySQL κάθε δευτερόλεπτο. Στη συνέχεια ενσωματώνει αυτές τις πληροφορίες στους Χάρτες Google μέσω του API Χαρτών Google που εμφανίζει τη θέση σε ένα χάρτη. Δεδομένου ότι οι πληροφορίες θέσης ανακτώνται κάθε δευτερόλεπτο και οι χάρτες ενημερώνονται με την ίδια συχνότητα, επιτυγχάνεται ένα φαινόμενο παρακολούθησης GPS σε πραγματικό χρόνο. Στο τέλος της διπλωματικής αυτής, στο Παράρτημα Γ', παρατίθεται το αρχείο που η Zope εξυπηρετεί στο απομακρυσμένο πρόγραμμα περιήγησης. Το ενσωματωμένο javascript παίρνει το αρχείο data.xml και συγκεντρώνει τους δείκτες DTML με το γεωγραφικό πλάτος και το γεωγραφικό μήκος των τελευταίων εισαγόμενων πληροφοριών θέσης από τη βάση δεδομένων MySQL.

### 6.2 Αισθητήρας Απόστασης

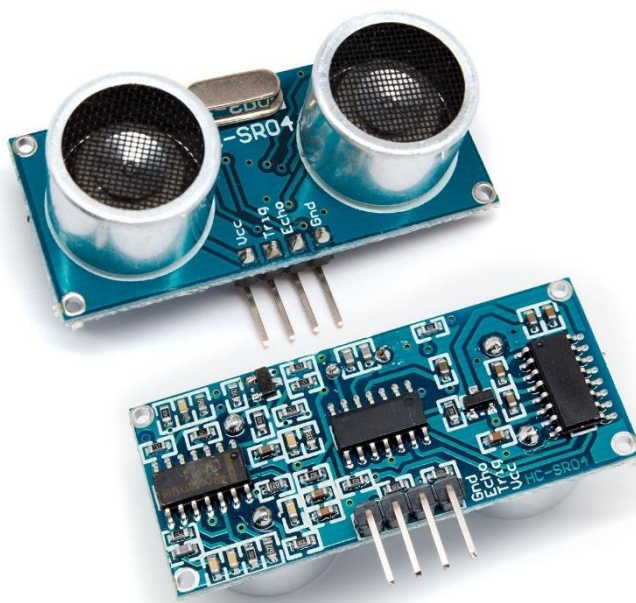
Επιλέον, μια πολύ ιδιαίτερη και συνάμα χρήσιμη επέκταση είναι η προσαρμογή στην ήδη υπάρχουσα συσκευή μας ενός αισθητήρα απόστασης. Ο αισθητήρας αυτός



θα είναι ιδιαίτερα χρήσιμος για τα βαγόνια των τραινών που διασχίζουν την Ελλάδα, καθώς θα μετράει την απόσταση του κάτω μέρους του βαγονιού από τις ράγες. Αυτό είναι χρήσιμο για τον εξής λόγο: κάθε βαγόκι ανάλογα με τα εμπορεύματα που μεταφέρει λόγω βάρους χαμηλώνει και πλησιάζει τις ράγες. Αντίστοιχα, όταν χάνει εμπορεύματα (χάνει δηλαδή και βάρος) τότε το βαγόκι απομακρύνεται από τις ράγες, καθώς είναι πιο ελαφρύ. Έχει, λοιπόν, έχει παρατηρηθεί από τους συγκοινωνιολόγους, ότι οι στιγμές που τα τρέινα σταματούν στους διάφορους σταθμούς και αφήνουν κάποια εμπορεύματα, ευδοκιμούν για κλοπές, καθώς το βαγόκι ανοίγει την πόρτα του. Εκμεταλλευόμενοι, λοιπόν, ορισμένοι αυτή την κατάσταση βρίσκουν ευκαρία να εισέλθουν στα βαγόνια, από τα οποία ξεφορτώνονται εμπορεύματα, και να κλέψουν κάποιο άλλο εμπόρευμα που συνεχίζει με το τρέινο για μία επόμενη στάση. Επομένως, το βαγόκι στο τέλος της μίας στάσης του, εκτός από τα εμπορεύματα που ξεφόρτωσε, βρίσκεται να έχει χάσει και κάποια άλλα εμπορεύματα λόγω κλοπής. Θέτοντας όμως αυτόν τον αισθητήρα στην συσκευή μας, που μετράει την απόσταση του βαγονιού από τις ράγες, δηλαδή με άλλα λόγια το βάρος του βαγονιού, μπορούμε να γνωρίσουμε επ ακριβώς τον αριθμό των εμπορευμάτων που μεταφέρονται ανά βαγόκι. Έτσι, μόλις το τρέινο ξεφορτώσει τα εμπορεύματα που πρέπει σε κάποιον σταθμό, ξαναξεκινάει και το κάθε βαγόκι θα πρέπει να έχει συγκεκριμένο βάρος ανάλογα με τα εμπορεύματα τα οποία έχει αφήσει. Αν, λοιπόν, γίνει κάποια κλοπή το βαγόκι θα είναι ελαφρύτερο φεύγοντας από τον σταθμό και άρα η απόσταση βαγονιού – γης θα μεγαλώσει και έτσι ο αισθητήρας θα μας ενημερώσει αμέσως για την αλλαγή αυτή. Με αυτήν την απλή προσθήκη στην συσκευή μας, είμαστε σε θέση να μειώσουμε σε πολύ σημαντικό βαθμό τις απώλειες εμπορευμάτων που μετακινούνται καθημερινώς από το συγκοινωνιακό δίκτυο της Ελλάδος.

### 6.2.1 Υλοποίηση Αισθητήρα

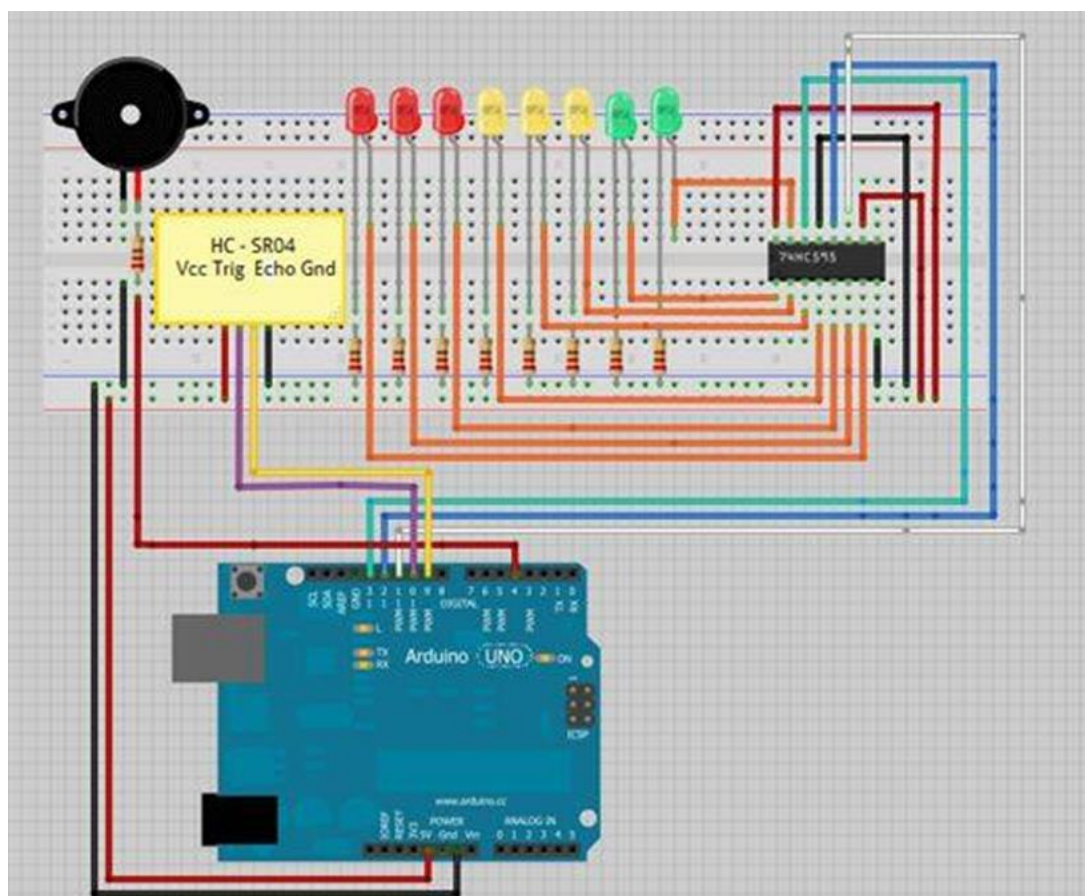
Λόγω του εξαιρετικού ενδιαφέροντος που παρουσιάζει η χρήση του αισθητήρα απόστασης, αποφασίσαμε να υλοποιήσουμε έναν με βάση πάντα το Arduino UNO, που αποτελεί και την βάση για τη συσκευή μας. Ο λόγος που χρησιμοποιήσαμε τον μικροελεγκτή αυτόν είναι γιατί το Arduino UNO είναι η βάση της συσκευής μας και επομένως μετά την υλοποίηση αυτή, ο αισθητήρας, θα μπορεί πολύ εύκολα να προσαρμοστεί στην υπάρχουσα συσκευή που κατασκευάσαμε. Σε πρώτη φάση κάναμε μια έρευνα αγοράς προκειμένου να διαλέξουμε τα κατάλληλα υλικά, με βάση την λειτουργικότητα τους άλλα και με βάση την τιμή τους. Όπως έχουμε πει θέλουμε η συσκευή να είναι όσο το δυνατόν πιο οικονομική γίνεται. Από την αναζήτηση αισθητήρων που κάναμε, καταλήξαμε ότι το καταλληλότερος είναι ο HC-SR04 Ultrasonic Sensor, ο οποίος είναι σε θέση να μετρά με απόλυτη ακρίβεια την απόσταση, αλλά και είναι ιδιαίτερα οικονομικός, μόλις 0.97\$.



Επιπλέον για την κατασκευή της συσκευής χρειαστήκαμε ακόμα, έναν buzzer, 2 πράσινα leds, 3 κίτρινα leds, 3 κόκκινα leds, ένα ολοκληρωμένο κύκλωμα - το 74HC595 8 bit shift register - και 8 αντιστάσεις των 220 Ohm. Στο σημείο, λοιπόν, αυτό μπορούμε να διαπιστώσουμε ότι η συνολική διάταξη του αισθητήρα δεν πρόκειται να κοστίσει πάνω από 15\$ (10\$ το Arduino UNO, 0,97\$ ο αισθητήρας και από 4\$ όλα τα υπόλοιπα υλικά), γεγονός που τον καθιστά πολύ οικονομικό.

Ο αισθητήρας χρησιμοποιεί υπέρυθρες για να υπολογίσει την απόσταση ενός αντικειμένου με μεγάλη ακρίβεια, όπως τα δελφίνια και οι νυχτερίδες. Η εμβέλεια του κυμαίνεται από 2 έως 400 εκατοστά και η ακρίβειά του φτάνει τα 3 mm. Επιπλέον ο αισθητήρας λειτουργεί με 5V DC τάση και διαρέεται από 15 mA. Τέλος, η λειτουργία του δεν επηρεάζεται από το φως.

Παρακάτω φαίνεται η σύνδεση που υλοποιήσαμε για την κατασκευή του αισθητήρα:



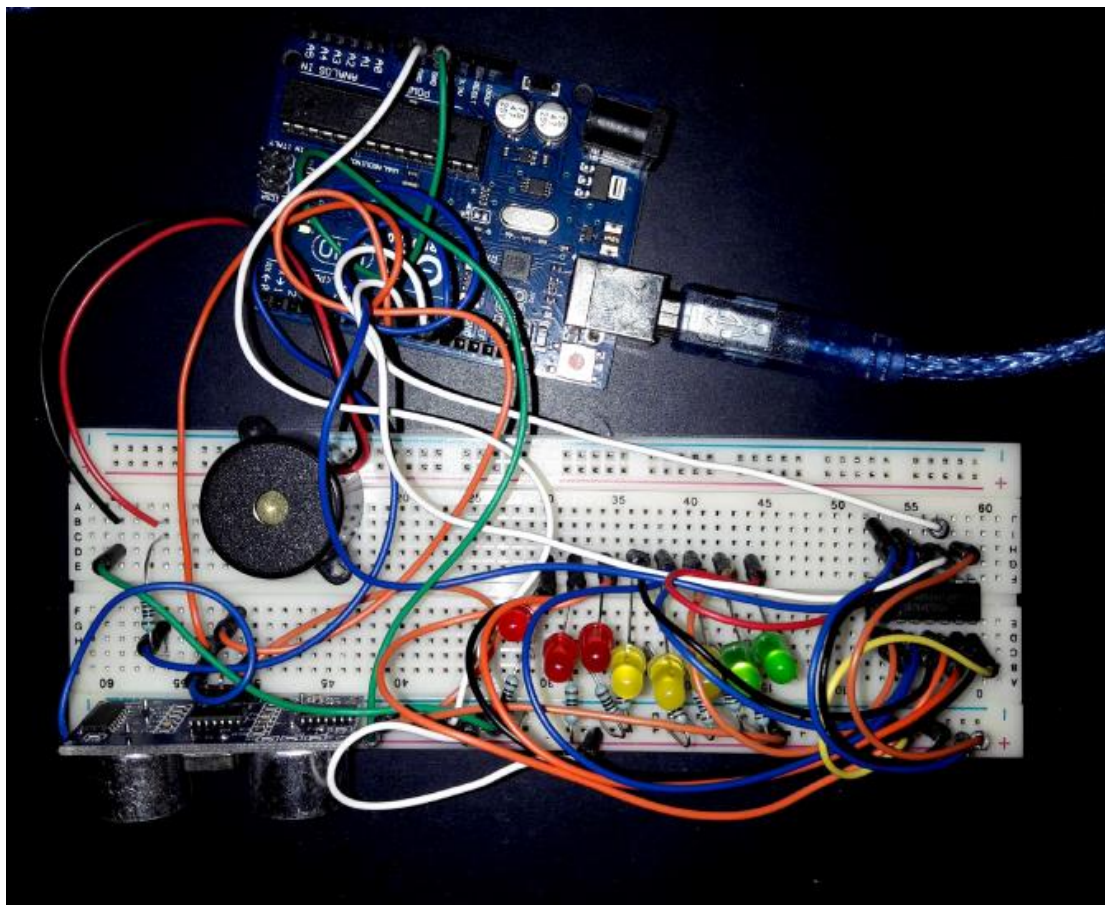
Στη συνέχεια αφού εκτελέσαμε τις σωστές συνδέσεις ήρθε η ώρα να γράψουμε τον αντίστοιχο κώδικα για την σωστή λειτουργία του αισθητήρα. Ο κώδικας, λοιπόν, που υλοποιήσαμε βρίσκεται στο τέλος της διπλωματικής. Η λογική του αλγορίθμου που υλοποιήσαμε είναι η εξής:

- Αν ο αισθητήρας δεν αντιλαμβάνεται μπροστά του κάποιο αντικείμενο, δηλαδή είναι πολύ μακριά από αυτόν, τότε να τυπώνει απλά τη λέξη “out of range”.
- Αλλιώς, αν αντιληφθεί κοντά του κάποιο αντικείμενο και αυτό έχει μεγάλη απόσταση (ικανή όμως να την αντιληφθεί ο αισθητήρας) τότε αρχίζει και χτυπάει ο buzzer ενώ παράλληλα ανάβουν τα πράσινα leds. Επίσης στην οθόνη εκτυπώνεται ο αριθμός των cm που απέχει το αντικείμενο από τον αισθητήρα.
- Άμα, το απομακρυσμένο αντικείμενο πλησιάσει κι άλλο, τότε ανάβουν και τα κίτρινα leds ενώ παράλληλα ο buzzer χτυπάει με μεγαλύτερη συχνότητα.
- Τέλος, άμα το αντικείμενο είναι πραγματικά κοντά στον αισθητήρα, τότε ανάβουν και τα κόκκινα leds και ο buzzer χτυπάει με ακόμα μεγαλύτερη συχνότητα, οριακά σαν συνεχόμενος και αδιάκοπος ήχος. Σε κάθε περίπτωση στην οθόνη του Arduino IDE τυπώνονται τα cm, που απέχει ο αισθητήρας από το σώμα.

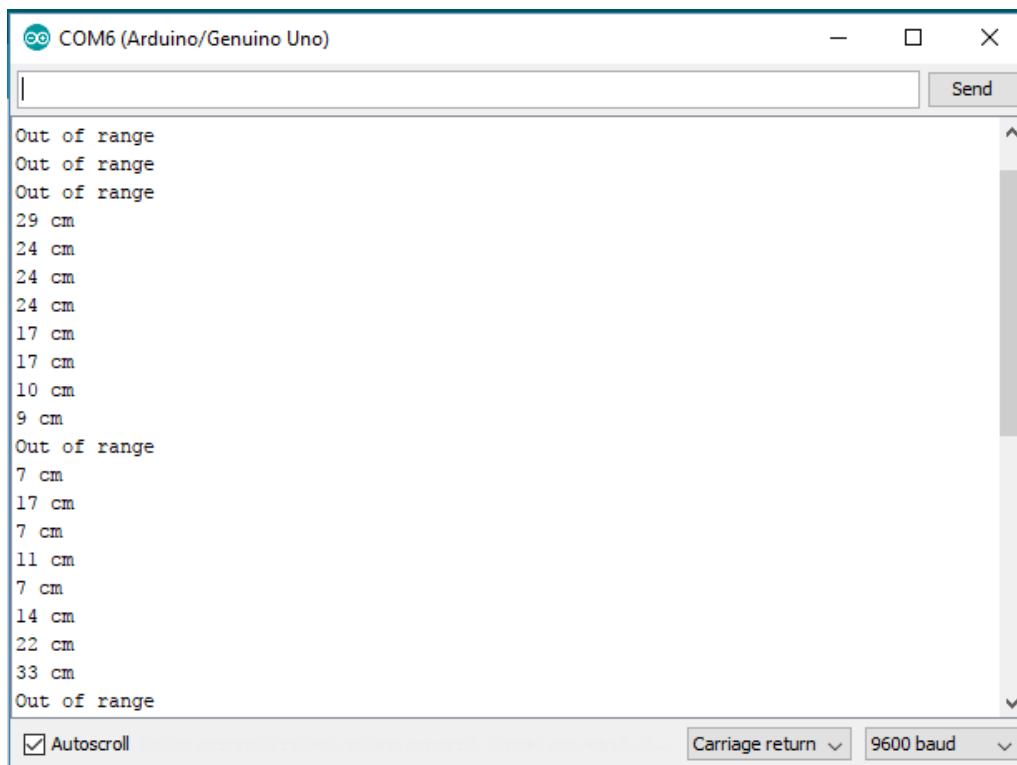
Τέλος να σημειωθεί ότι επιλέξαμε να βάλουμε στην διάταξη μας και ένα ολοκληρωμένο κύκλωμα, το 74HC595 8 bit shift register, που μετατρέπει την απόσταση σε δυαδικό αριθμό οκτώ bit. Έτσι, όταν ένα αντικείμενο εισέρχεται στην περιοχή εμβέλειας του αισθητήρα, αυτός στέλνει σήμα στο Arduino. Εκεί υπολογίζεται η απόσταση του αντικειμένου σε εκατοστά και αντιστοιχίζει την απόσταση σε ένα εύρος τιμών από 1 μέχρι 8. Στην συνέχεια, με βάση την τιμή που στένει στο ολοκληρωμένο, αυτό ανάβει τα LEDs ενώ το Buzzer παράγει ήχους διαφορετικής συχνότητας.

### 6.2.2 Φωτογραφίες από τη Διάταξη και από την Εκτέλεση του Κώδικα

Αρχικά παραθέτουμε την φωτογραφία της διάταξής μας:



Τέλος, παρουσιάζουμε και την οθόνη της σειριακής εξόδου του Arduino, κατά την εκτέλεση του κώδικα που γράψαμε:



Όπως μπορούμε να διαπιστώσουμε, στην οθόνη εκτυπώνεται η απόσταση μεταξύ αισθητήρα και αντικείμενου. Βλέπουμε ότι όταν δεν πλησιάζουμε κάποιο αντικείμενο κοντά στον αισθητήρα εκτυπώνεται στην οθόνη του υπολογιστή η ένδειξη “Out of range”. Αλλιώς, άμα πλησιάσουμε κάποιο σώμα μπροστά από τον αισθητήρα μας εκτυπώνει την μεταξύ τους απόσταση.

Επομένως, κατασκευάσαμε έναν αισθητήρα απόστασης, ιδιαίτερα οικονομικό, με βάση το Arduino UNO, ο οποίος μπορεί πολύ εύκολα να προσαρμοστεί στην συσκευή μας και να μας εξυπηρετήσει ακόμα περισσότερο κάποιες ανάγκες μας, όπως είπαμε πριν.

### 6.3 Επιτάχυνση-Επιβράδυνση

Μία ακόμα πολύ σημαντική μελλοντική επέκταση είναι η εύρεση της κίνησης που εκτέλεσε το όχημα που φέρει τη συσκευή μας, δηλαδή αν είναι επιταχυνόμενη ή επιβραδυνόμενη από σημείο σε σημείο. Αυτή η επέκταση μπορεί να γίνει πολύ εύκολα, καθώς ήδη έχουμε κάνει το πρώτο μεγάλο βήμα και έχουμε πετύχει αποστολή συντεταγμένων με σταθερό ρυθμό. Τώρα αρκεί να εκμεταλευτούμε τους εξής δύο τύπους:

*Ταχύτητα*

$$u = \frac{dx}{dt}$$

Όπου dx το διάστημα που διανύει το σώμα σε χρόνο dt. Και,

### Απόσταση

$$x = \sqrt{(a_2 - a_1)^2 + (b_2 - b_1)^2}$$

Όπου  $a_1$ ,  $b_1$  και  $a_2$ ,  $b_2$  οι προηγούμενες και οι τωρινές συντεταγμένες των σημείων αντίστοιχα.

Ουσιαστικά, λοιπόν, το πρόβλημα έγκειται απλά στον υπολογισμό της απόστασης που θα διανύσει το κινητό στο χρονικό διάστημα που θα επιλέξουμε εμείς. Αν, λοιπόν, ένα όχημα που φέρει πάνω τη συσκευή μας κινηθεί από το σημείο Α στο σημείο Β και απο εκεί πάει στο σημείο Γ, τότε αυτά που πρέπει να συγκριθούν είναι οι αποστάσεις (ΑΒ) και (ΒΓ). Εφόσον ο χρόνος που θα γίνεται η αποστολή των μηνυμάτων θα είναι σταθερός και ελεγχόμενος από εμάς, μπορούμε να εξάγουμε το εξής συμπέρασμα:

- Αν  $(AB) > (BG)$ , σημαίνει ότι το όχημα με την συσκευή μας επάνω, διένυσε στον ίδιο χρόνο μεγαλύτερη απόσταση αρχικά, και μικρότερη μετά. Άρα, προφανώς συμπεραίνουμε ότι η ταχύτητα του κινητού μειώθηκε, άρα έχουμε επιβράδυνση.
- Αντίστοιχα, αν  $(AB) < (BG)$  σημαίνει ότι το όχημα αρχικά διένυσε μικρότερο διάστημα στον ίδιο χρόνο απ' ότι μετά. Άρα, σημαίνει ότι η ταχύτητά του αυξήθηκε και άρα είχαμε επιτάχυνση.

Προφανώς, για την εφαρμογή της συγκεκριμένης επέκτασης, θα πρέπει ο χρόνος αποστολής να είναι σταθερός και σχετικά μεγάλος (π.χ. 10 sec) προκειμένου να έχει διανήσει το κινητό ένα ικανό διάστημα. Επιπλέον, στο αν έχουμε επιτάχυνση ή επιβράδυνση, δεν αρκεί μόνο να ισχύει  $(AB) > (BG)$ , καθώς υπάρχει πιθανότητα αποκλίσεων του σήματος του GPS ή ακόμα και του δικτύου τηλεφωνίας.

Για να είμαστε πιο ακριβείς και σωστοί στο άμα το σώμα κινήθηκε επιταχυνόμενα ή επιβραδυνόμενα, οι παραπάνω συνθήκες θα πρέπει να διαμορφωθούν στον τελικό κώδικα ως εξής:

- Αν  $(AB) > 2 * (AG)$  τότε είμαστε σίγουροι ότι το σώμα κινήθηκε επιβραδυνόμενα. Αλλιώς,
- Αν  $(AB) < \frac{1}{2} * (AG)$  τότε το σώμα εκτέλεσε επιταχυνόμενη κίνηση σίγουρα.

Με την παραπάνω εφαρμογή, μπορούμε να κάνουμε ακόμα μία πολύ σημαντική επέκταση. Μπορούμε να τοποθετήσουμε την συσκευή μας πάνω σε ένα όχημα και να το βάλουμε να διανύσει μια συγκεκριμένη διαδρομή και να σημειώνουμε πάνω στους διαδραστικούς χάρτες με χρώμα μπλε το μέρος της διαδρομής που η κίνηση του οχήματος ήταν επιταχυνόμενη, ενώ με χρώμα κόκκινο το μέρος της διαδρομής που το κινητό έκανε επιβραδυνόμενη κίνηση. Οι περιοχές της διαδρομής που θα είναι κόκκινες, δηλαδή τα μέρη όπου το όχημα επιβράδυνε, μπορεί να οφείλονται σε κίνηση στους δρόμους, κάποιο φανάρι ή ακόμα ανώμαλο έδαφος ή ανηφόρα. Στην πλειοψηφία τους, τα σημεία της διαδρομής που το σώμα αναγκάστηκε να επιβραδύνει, είναι σημεία που δημιουργούν καθυστερήσεις και αυξημένη κατανάλωση καυσίμων. Με την επέκταση όμως αυτή, θα είμαστε σε θέση να γνωρίζουμε τις «επιβραδυνόμενες περιοχές» ώστε να τις αποφύγουμε και έτσι να

επιλέξουμε διαφορετική πορεία. Αυτό θα έχει ως αποτέλεσμα όταν είναι έτοιμος ο υπόλοιπος στόλος να ξεκινήσει τη διαδρομή του, να ξέρει από πριν πως να κινηθεί. Αν παραδείγματος χάριν υπάρχει μία μεγάλη ανηφόρα που το όχημα αρχικά επιβράδυνε για μεγάλο διάστημα, καθυστέρησε και έχασε και αρκετά καύσιμα κατά τη διάβασή της, να γνωρίζει ο υπόλοιπος στόλος να αποφύγει τη συγκεκριμένη ανηφόρα και να επιλέξει άλλη διαδρομή που η κίνηση του οχήματος πριν ήταν επιταχυνόμενη. Με αυτόν τον τρόπο και γλιτώνουμε προβλήματα καθυστερήσεων που προκύπτουν πολλές φορές στους δρόμους, αλλά και εξοικονομούμε καύσιμα και κατά συνέπεια χρήματα, καθώς μια ανηφόρα σίγουρα θα αναγκάσει το όχημα να καταναλώσει πολύ περισσότερο καύσιμο σε σχέση με μια ευθεία.

#### 6.4 Βελτίωση Σήματος

Τέλος, η συσκευή μπορεί μελλοντικά με κατάλληλη τροποποίηση-επέκταση να αντιμετωπίσει τα δύο σοβαρότερα προβλήματα λειτουργίας της που αφορούν τις περιπτώσεις αδυναμίας εύρεσης σήματος GSM δικτύου, καθώς και σήματος GPS. Έτσι λοιπόν, σε περίπτωση που η συσκευή βρίσκεται σε περιοχή ανύπαρκτου ή πολύ χαμηλής στάθμης σήματος GSM δικτύου, προσθέτοντας στη συσκευή δυνατότητα RF ή Bluetooth λειτουργίας, καθίσταται δυνατή η αναζήτηση του κοντινότερου κινητού τηλεφώνου με επίσης ενεργοποιημένες τις προαναφερθείσες λειτουργίες. Μόλις η συσκευή εντοπίσει το κινητό τηλέφωνο, κάνει έκκληση για βοήθεια αποστέλλοντας τα στοιχεία του χρήστη και την τοποθεσία του κατά τα γνωστά. Για χρήση της συσκευής σε εσωτερικούς χώρους ή σε περιπτώσεις όπου ο ουρανός καλύπτεται από σύννεφα, όπου η συσκευή GPS εκ φύσεως δεν μπορεί να λάβει σήμα από το δίκτυο δορυφόρων, είναι δυνατή η εύρεση του ατόμου μέσω του δικτύου GSM, με εντοπισμό πρώτα της κυψελίδας στην οποία βρίσκεται και στη συνέχεια της ακριβούς του τοποθεσίας μέσα σε αυτήν.

## ΠΑΡΑΡΤΗΜΑ Α

### ΚΩΔΙΚΕΣ ΓΙΑ ΕΛΕΓΧΟ ΛΕΙΤΟΥΡΓΙΑΣ ΤΩΝ ΣΥΣΚΕΥΩΝ

#### GPS

```
#include <Adafruit_GPS.h>
#include <SoftwareSerial.h>

// If you're using a GPS module:
// Connect the GPS Power pin to 5V
// Connect the GPS Ground pin to ground
// If using software serial (sketch example default):
// Connect the GPS TX (transmit) pin to Digital 3
// Connect the GPS RX (receive) pin to Digital 2
// If using hardware serial (e.g. Arduino Mega):
// Connect the GPS TX (transmit) pin to Arduino RX1, RX2 or RX3
// Connect the GPS RX (receive) pin to matching TX1, TX2 or TX3

// If you're using the Adafruit GPS shield, change
// SoftwareSerial mySerial(3, 2); -> SoftwareSerial mySerial(8, 7);
// and make sure the switch is set to SoftSerial

// If using software serial, keep this line enabled
// (you can change the pin numbers to match your wiring):
SoftwareSerial mySerial(3, 2);

// If using hardware serial (e.g. Arduino Mega), comment out the
// above SoftwareSerial line, and enable this line instead
// (you can change the Serial number to match your wiring):

//HardwareSerial mySerial = Serial1;
```



```
Adafruit_GPS GPS(&mySerial);

// Set GPSECHO to 'false' to turn off echoing the GPS data to the Serial console
// Set to 'true' if you want to debug and listen to the raw GPS sentences.
#define GPSECHO true

// this keeps track of whether we're using the interrupt
// off by default!
boolean usingInterrupt = false;
void useInterrupt(boolean); // Func prototype keeps Arduino 0023 happy

void setup()
{

  // connect at 115200 so we can read the GPS fast enough and echo without dropping chars
  // also spit it out
  Serial.begin(115200);
  Serial.println("Adafruit GPS library basic test!");

  // 9600 NMEA is the default baud rate for Adafruit MTK GPS's- some use 4800
  GPS.begin(9600);

  // uncomment this line to turn on RMC (recommended minimum) and GGA (fix data)
  including altitude
  GPS.sendCommand(PMTK_SET_NMEA_OUTPUT_RMCGGA);
  // uncomment this line to turn on only the "minimum recommended" data
  //GPS.sendCommand(PMTK_SET_NMEA_OUTPUT_RMCONLY);

  // For parsing data, we don't suggest using anything but either RMC only or RMC+GGA
  since
```

```
// the parser doesn't care about other sentences at this time

// Set the update rate
GPS.sendCommand(PMTK_SET_NMEA_UPDATE_1HZ); // 1 Hz update rate
// For the parsing code to work nicely and have time to sort thru the data, and
// print it out we don't suggest using anything higher than 1 Hz

// Request updates on antenna status, comment out to keep quiet
GPS.sendCommand(PGCMD_ANTENNA);

// the nice thing about this code is you can have a timer0 interrupt go off
// every 1 millisecond, and read data from the GPS for you. that makes the
// loop code a heck of a lot easier!
useInterrupt(true);

delay(1000);
// Ask for firmware version
mySerial.println(PMTK_Q_RELEASE);
}

// Interrupt is called once a millisecond, looks for any new GPS data, and stores it
SIGNAL(TIMERO_COMPA_vect) {
  char c = GPS.read();
  // if you want to debug, this is a good time to do it!
#ifdef UDR0
  if (GPSECHO)
    if (c) UDR0 = c;
  // writing direct to UDR0 is much much faster than Serial.print
  // but only one character can be written at a time.
#endif
#endif
```

```
}

void useInterrupt(boolean v) {
  if (v) {
    // Timer0 is already used for millis() - we'll just interrupt somewhere
    // in the middle and call the "Compare A" function above
    OCR0A = 0xAF;
    TIMSK0 |= _BV(OCIE0A);
    usingInterrupt = true;
  } else {
    // do not call the interrupt function COMPA anymore
    TIMSK0 &= ~_BV(OCIE0A);
    usingInterrupt = false;
  }
}

uint32_t timer = millis();

void loop()          // run over and over again
{
  // in case you are not using the interrupt above, you'll
  // need to 'hand query' the GPS, not suggested :(
  if (!usingInterrupt) {
    // read data from the GPS in the 'main loop'
    char c = GPS.read();
    // if you want to debug, this is a good time to do it!
    if (GPSECHO)
      if (c) Serial.print(c);
  }

  // if a sentence is received, we can check the checksum, parse it...
  if (GPS.newNMEAreceived()) {
```

```
// a tricky thing here is if we print the NMEA sentence, or data
// we end up not listening and catching other sentences!
// so be very wary if using OUTPUT_ALLDATA and trying to print out data
//Serial.println(GPS.lastNMEA()); // this also sets the newNMEAreceived() flag to false

if (!GPS.parse(GPS.lastNMEA())) // this also sets the newNMEAreceived() flag to false
  return; // we can fail to parse a sentence in which case we should just wait for another
}

// if millis() or timer wraps around, we'll just reset it
if (timer > millis()) timer = millis();

// approximately every 2 seconds or so, print out the current stats
if (millis() - timer > 2000) {
  timer = millis(); // reset the timer

  Serial.print("\nTime: ");
  Serial.print(GPS.hour, DEC); Serial.print(':');
  Serial.print(GPS.minute, DEC); Serial.print(':');
  Serial.print(GPS.seconds, DEC); Serial.print('.');
  Serial.println(GPS.milliseconds);
  Serial.print("Date: ");
  Serial.print(GPS.day, DEC); Serial.print('/');
  Serial.print(GPS.month, DEC); Serial.print("/20");
  Serial.println(GPS.year, DEC);
  Serial.print("Fix: "); Serial.print((int)GPS.fix);
  Serial.print(" quality: "); Serial.println((int)GPS.fixquality);
  if (GPS.fix) {
    Serial.print("Location: ");
    Serial.print(GPS.latitude, 4); Serial.print(GPS.lat);
    Serial.print(", ");
```

```
Serial.print(GPS.longitude, 4); Serial.println(GPS.lon);
Serial.print("Location (in degrees, works with Google Maps): ");
Serial.print(GPS.latitudeDegrees, 4);
Serial.print(", ");
Serial.println(GPS.longitudeDegrees, 4);

Serial.print("Speed (knots): "); Serial.println(GPS.speed);
Serial.print("Angle: "); Serial.println(GPS.angle);
Serial.print("Altitude: "); Serial.println(GPS.altitude);
Serial.print("Satellites: "); Serial.println((int)GPS.satellites);
}
}
}
```

### GSM

```
#include <GSM.h>
#define PINNUMBER "" // initialize the library instance
GSM gsmAccess; // include a 'true' parameter for debug enabled
GSM_SMS sms;
// char array of the telephone number to send SMS
char remoteNumber[20]= "00306947050972";
// char array of the message
char txtMsg[200]="test";
```

```
void setup()
{
  // initialize serial communications
  Serial.begin(9600);
  Serial.println("SMS Messages Sender");
  // connection state
  boolean notConnected = true;
  // Start GSM shield
  // If your SIM has PIN, pass it as a parameter of begin() in quotes
  while(notConnected)
  {
    if(gsmAccess.begin(PINNUMBER)==GSM_READY)
      notConnected = false;
    else
    {
      Serial.println("Not connected");
      delay(1000);
    }
  }
  Serial.println("GSM initialized");
  sendSMS();
}

void loop()
{
  // nothing to see here
}

void sendSMS(){
  Serial.print("Message to mobile number: ");
  Serial.println(remoteNumber);
}
```

```
// sms text
Serial.println("SENDING");
Serial.println();
Serial.println("Message:");
Serial.println(txtMsg);

// send the message
sms.beginSMS(remoteNumber);
sms.print(txtMsg);
sms.endSMS();
Serial.println("\nCOMPLETE!\n");
delay(100000000);
}
```

## ΠΑΡΑΡΤΗΜΑ Β

### ΤΕΛΙΚΟΙ ΚΩΔΙΚΕΣ ΕΡΓΑΣΙΑΣ

#### ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Α

```
// Kwdikas 1: To GPS an lavei tin leksi "Location" apo kinito tilefwno tote tou epistrefei tis
syntetagmenes tou simeiou sto opoio vrisketai to GPS.

// Epipleon an patisoume kapoio koumpi apothikeuetai ekeini ti stigmi i topothesia tou GPS.
Stis 7 apothikeuseis mas stelνονται epi topou oles oi syntetagmenes pou apithikeutikan kata
ti diadromi

// Author: Iason-Stylianios Kasapleris

// A.M.: 03112051

#include <AltSoftSerial.h> //Include all relevant libraries, emulate an additional serial port,
allowing you to communicate with another serial device

#include <TinyGPS++.h> //this library provides compact and easy-to-use methods for
extracting position, date, time, altitude, speed, and course from consumer GPS devices.

#include <GSM.h> //this library enables an Arduino board to do most of the operations you
can do with a GSM phone: place and receive voice calls, send and receive SMS, and connect
to the internet over a GPRS network.

#include <stdio.h>

#define PINNUMBER "" // PIN Number for the SIM - leave blank unless your SIM has a pin,
this is inserted between ""

static const uint32_t GPSPbaud = 9600; //Baud rate for communication with the GPS, Adafruit
GPS = 9600, your GPS may well be 4800, check the spec

TinyGPSPlus gps; // The TinyGPS++ object for interfacing with the GPS

AltSoftSerial ss; // The serial connection object to the GPS device

String yourPassword = "Location"; // Put the password here between the ""

String password; // Temporary variable used for comparison of passwords

GSM gsmAccess; // Initialise the library instances

GSM_SMS sms;

char senderNumber[20]="00306947050972"; // Array to hold the number a SMS is retrieved
from

// Array which holds the coordinates and the color of the button pressed.
```



```
int led1 = 13;
int led2 = 12;
const int GREEN_BUTTON = 4;
const int YELLOW_BUTTON = 7;
int yellow_val = 0;
int green_val = 0;
String txtmsg = "";

void setup()
{
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  ss.begin(GPSBaud); // begin the GPS serial connection
  Serial.begin(9600); // begin Serial communication with the computer at 9600 baud rate
  Serial.println("Hello"); // Print to the computer
  Serial.println("Initialising...");
  boolean notConnected = true; // connection state
  while(notConnected) // until it connects
  {
    if(gsmAccess.begin(PINNUMBER)==GSM_READY) // if it succeeds connecting
      notConnected = false; // connected
    else
    {
      Serial.println("Not connected"); // print to the computer
      delay(1000); //delay
    }
  }
  Serial.println("GSM initialized"); // print to the computer
  Serial.println("Waiting for messages");
}
```

```
void loop()
{
  while (ss.available() > 0) //while there is stuff in the buffer
  {
    if (gps.encode(ss.read())) //if it can successfully decode it, do it. Else try again when more
    characters are in the buffer
  }
  if (sms.available()) // if a text has been recieved
  {
    Serial.println("Message received from:"); // print to the computer
    sms.remoteNumber(senderNumber, 20); // assign the sender number to the
    "senderNumber" variable
    Serial.println(senderNumber); // print the sender number to the computer
    password = ""; // flush the temporary variable
    char c;
    while(c=sms.read())
    {
      password += c; // append the sms to the "password" variable
    }
    Serial.println(password); // print the contents of the sms
    Serial.println("\nEND OF MESSAGE"); // print to the computer

    sms.flush(); // delete message from modem buffer
    Serial.println("MESSAGE DELETED"); // print to the computer
    if (password == yourPassword) // if the sms contains the correct password
    {
      Serial.println("\nPASSWORD VALID"); // print to the computer
      digitalWrite(led2, LOW);
      digitalWrite(led1, HIGH);
      sms.beginSMS(senderNumber); // begin an sms to the sender number
      sms.print(gps.location.lat(), 6); // append the lat to the sms
      sms.print(","); // append a comma
      sms.print(gps.location.lng(), 6); // append the lon to the sms
    }
  }
}
```

```
    sms.endSMS(); //send the sms
}
else {
    Serial.println("\nPASSWORD NOT VALID"); // print to the computer
    digitalWrite(led1, LOW);
    digitalWrite(led2, HIGH);
}
}
green_val = digitalRead(GREEN_BUTTON);
if (green_val == HIGH) {
    delay(500);
    sms.print(gps.location.lat(), 6); // append the lat to the sms
    sms.print(","); // append a comma
    sms.print(gps.location.lng(), 6); // append the lon to the sms
    sms.print("\n");
    txtmsg += String(gps.location.lat(), 6);
    txtmsg += ",";
    txtmsg += String(gps.location.lng(), 6);
    txtmsg += "g\n";
    Serial.println(txtmsg);
    if (txtmsg.length() > 120) {
        sms.beginSMS(senderNumber);
        sms.print(txtmsg);
        sms.endSMS();
        txtmsg = "";
    }
}
Serial.println("outside the buttons");
yellow_val = digitalRead(YELLOW_BUTTON); // read input value and store it
if (yellow_val == HIGH) {
    delay(500);
```

```

sms.print(gps.location.lat(), 6); // append the lat to the sms
sms.print(","); // append a comma
sms.print(gps.location.lng(), 6); // append the lon to the sms
sms.print("\n");
txtmsg += String(gps.location.lat(), 6);
txtmsg += ",";
txtmsg += String(gps.location.lng(), 6);
txtmsg += "\n";
Serial.println(txtmsg);
if (txtmsg.length() > 120) {
  sms.beginSMS(senderNumber);
  sms.print(txtmsg);
  sms.endSMS();
  txtmsg = "";
}
}
delay(1000); // delay
}

```

### ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Β

```

// Kwdikas 2: To GPS lamvanei sima kathws o kwdikas trexei synexws kai kathe 5
stigmata topothesias, to GSM stelnei ta dedomena se kinito tilefwno automata
// Author: Iason-Stylianos Kasapleris
// A.M.: 03112051
#include <AltSoftSerial.h> //Include all relevant libraries
#include <TinyGPS++.h>
#include <GSM.h>
#include <stdio.h>

```

```
#define PINNUMBER "" // PIN Number for the SIM - leave blank unless your SIM has
a pin, this is inserted between ""

static const uint32_t GPSBaud = 9600; //Baud rate for communication with the GPS,
Adafruit GPS = 9600, your GPS may well be 4800, check the spec

TinyGPSPlus gps; // The TinyGPS++ object for interfacing with the GPS

AltSoftSerial ss; // The serial connection object to the GPS device

String yourPassword = "Location"; // Put the password here between the ""

String password; // Temporary variable used for comparison of passwords

GSM gsmAccess; // Initialise the library instances

GSM_SMS sms;

char senderNumber[20]="00306947050972"; // Array to hold the number a SMS is
retrieved from

// Array which holds the coordinates and the color of the button pressed.

int led1 = 13;

int led2 = 12;

const int GREEN_BUTTON = 4;

const int YELLOW_BUTTON = 7;

int yellow_val = 0;

int green_val = 0;

String txtmsg = "";

void setup()
{
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  ss.begin(GPSBaud); // begin the GPS serial connection
  Serial.begin(9600); // begin Serial communication with the computer at 9600 baud
rate
  Serial.println("Hello"); // Print to the computer
  Serial.println("Initialising...");
```

```
boolean notConnected = true; // connection state
while(notConnected) // until it connects
{
  if(gsmAccess.begin(PINNUMBER)==GSM_READY) // if it succeeds connecting
    notConnected = false; // connected
  else
  {
    Serial.println("Not connected"); // print to the computer
    delay(1000); //delay
  }
}
Serial.println("GSM initialized"); // print to the computer
Serial.println("Waiting for messages");
}

void loop()
{
  while (ss.available() > 0) //while there is stuff in the buffer
  {
    if (gps.encode(ss.read())) //if it can successfully decode it, do it. Else try again
    when more characters are in the buffer

    if (sms.available()) // if a text has been recieved
    {
      Serial.println("Message received from:"); // print to the computer
      sms.remoteNumber(senderNumber, 20); // assign the sender number to the
"senderNumber" variable
      Serial.println(senderNumber); // print the sender number to the computer
      password = ""; // flush the temporary variable
      char c;
      while(c=sms.read())
```

```
{
  password += c; // append the sms to the "password" variable
}
Serial.println(password); // print the contents of the sms
Serial.println("\nEND OF MESSAGE"); // print to the computer
sms.flush(); // delete message from modem buffer
Serial.println("MESSAGE DELETED"); // print to the computer
if (password == yourPassword) // if the sms contains the correct password
{
  Serial.println("\nPASSWORD VALID"); // print to the computer
  digitalWrite(led2, LOW);
  digitalWrite(led1, HIGH);
  sms.beginSMS(senderNumber); // begin an sms to the sender number
  sms.print(gps.location.lat(), 6); // append the lat to the sms
  sms.print(","); // append a comma
  sms.print(gps.location.lng(), 6); // append the lon to the sms
  sms.endSMS(); //send the sms
}
else {
  Serial.println("\nPASSWORD NOT VALID"); // print to the computer
  digitalWrite(led1, LOW);
  digitalWrite(led2, HIGH);
}
}
delay(5000);
txtmsg += String(gps.location.lat(), 6);
txtmsg += ",";
txtmsg += String(gps.location.lng(), 6);
txtmsg += "\n";
```

```

Serial.println(txtmsg);
if (txtmsg.length() > 250) {           //tyxaios arithmos poy thesame emeis
    sms.beginSMS(senderNumber);
    sms.print(txtmsg);
    //sms.endSMS();
    txtmsg = "";
    Serial.println("midenistike");
}
delay(1000); // delay
}

```

### ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Γ

```

// Kwdikas 3: To GPS lamvanei sima kathws o kwdikas trexei synexws kai mas
stelnei ana synexws tin topothesia
// Author: Iason-Stylianos Kasapleris
// A.M.: 03112051
#include <AltSoftSerial.h> //Include all relevant libraries
#include <TinyGPS++.h>
#include <GSM.h>
#include <stdio.h>
#define PINNUMBER "" // PIN Number for the SIM - leave blank unless your SIM
has a pin, this is inserted between ""
static const uint32_t GPSBaud = 9600; //Baud rate for communication with the GPS,
Adafruit GPS = 9600, your GPS may well be 4800, check the spec
TinyGPSPlus gps; // The TinyGPS++ object for interfacing with the GPS
AltSoftSerial ss; // The serial connection object to the GPS device
String yourPassword = "Location"; // Put the password here between the ""
String password; // Temporary variable used for comparison of passwords
GSM gsmAccess; // Initialise the library instances

```



```
GSM_SMS sms;

char senderNumber[20]="00306947050972"; // Array to hold the number a SMS is
retrieved from

// Array which holds the coordinates and the color of the button pressed.

int led1 = 13;
int led2 = 12;

const int GREEN_BUTTON = 4;
const int YELLOW_BUTTON = 7;

int yellow_val = 0;
int green_val = 0;
String txtmsg = "";
float y = 0;

void setup()
{
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  ss.begin(GPSBaud); // begin the GPS serial connection
  Serial.begin(9600); // begin Serial communication with the computer at 9600 baud
rate
  Serial.println("Hello"); // Print to the computer
  Serial.println("Initialising...");
  boolean notConnected = true; // connection state
  while(notConnected) // until it connects
  {
    if(gsmAccess.begin(PINNUMBER)==GSM_READY) // if it succeeds connecting
      notConnected = false; // connected
    else
    {
      Serial.println("Not connected"); // print to the computer
      delay(1000); //delay
    }
  }
}
```

```
}  
}  
  
Serial.println("GSM initialized"); // print to the computer  
Serial.println("Waiting for messages");  
}  
  
void loop()  
{  
  while (ss.available() > 0) //while there is stuff in the buffer  
  {  
    if (gps.encode(ss.read())) //if it can successfully decode it, do it. Else try again  
    when more characters are in the buffer  
  
  }  
  
  if (sms.available()) // if a text has been recieved  
  {  
    Serial.println("Message received from:"); // print to the computer  
    sms.remoteNumber(senderNumber, 20); // assign the sender number to the  
"senderNumber" variable  
    Serial.println(senderNumber); // print the sender number to the computer  
    password = ""; // flush the temporary variable  
    char c;  
    while(c=sms.read())  
    {  
      password += c; // append the sms to the "password" variable  
    }  
    Serial.println(password); // print the contents of the sms  
    Serial.println("\nEND OF MESSAGE"); // print to the computer  
    sms.flush(); // delete message from modem buffer  
    Serial.println("MESSAGE DELETED"); // print to the computer  
    if (password == yourPassword) // if the sms contains the correct password  
    {
```

```
Serial.println("\nPASSWORD VALID"); // print to the computer
digitalWrite(led2, LOW);
digitalWrite(led1, HIGH);
sms.beginSMS(senderNumber); // begin an sms to the sender number
sms.print(gps.location.lat(), 6); // append the lat to the sms
sms.print(","); // append a comma
sms.print(gps.location.lng(), 6); // append the lon to the sms
sms.endSMS(); //send the sms
}
else {
  Serial.println("\nPASSWORD NOT VALID"); // print to the computer
  digitalWrite(led1, LOW);
  digitalWrite(led2, HIGH);
}
}
y = millis(); //rwame to Arduino tin wra mexri twra
sms.beginSMS(senderNumber);
txtmsg += String(gps.location.lat(), 6);
txtmsg += ",";
txtmsg += String(gps.location.lng(), 6);
txtmsg += " ";
txtmsg += y;
txtmsg += "\n";
Serial.println(txtmsg);
sms.print(gps.location.lat(), 6); // append the lat to the sms
sms.print(","); // append a comma
sms.print(gps.location.lng(), 6); // append the lon to the sms
sms.print(" ");
sms.print(y);
sms.print("\n");
```

```

sms.endSMS();
delay(1000); // delay
Serial.println("after");
}

```

### ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Δ

```

// Kwdikas 4: To GPS lamvanei sima kathws o kwdikas trexei synexws kai mas
stelnei ana synexws tin topothesia
// Author: Iason-Stylianos Kasapleris
// A.M.: 03112051
#include <AltSoftSerial.h> //Include all relevant libraries
#include <TinyGPS++.h>
#include <GSM.h>
#include <stdio.h>
#define PINNUMBER "" // PIN Number for the SIM - leave blank unless your SIM
has a pin, this is inserted between ""
static const uint32_t GPSBaud = 9600; //Baud rate for communication with the GPS,
Adafruit GPS = 9600, your GPS may well be 4800, check the spec
TinyGPSPlus gps; // The TinyGPS++ object for interfacing with the GPS
AltSoftSerial ss; // The serial connection object to the GPS device
String yourPassword = "Location"; // Put the password here between the ""
String password; // Temporary variable used for comparison of passwords
GSM gsmAccess; // Initialise the library instances
GSM_SMS sms;
char senderNumber[20]="00306947050972"; // Array to hold the number a SMS is
retrieved from
// Array which holds the coordinates and the color of the button pressed.
int led1 = 13;
int led2 = 12;
String txtmsg = "";

```

```
unsigned int x = 0;
unsigned int t = 0;

void setup()
{
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  ss.begin(GPSBaud); // begin the GPS serial connection
  Serial.begin(9600); // begin Serial communication with the computer at 9600 baud
  rate
  Serial.println("Hello"); // Print to the computer
  Serial.println("Initialising...");
  boolean notConnected = true; // connection state
  while(notConnected) // until it connects
  {
    if(gsmAccess.begin(PINNUMBER)==GSM_READY) // if it succeeds connecting
      notConnected = false; // connected
    else
    {
      Serial.println("Not connected"); // print to the computer
      delay(1000); //delay
    }
  }
  Serial.println("GSM initialized"); // print to the computer
  Serial.println("Waiting for messages");
}

void loop()
{
  while (ss.available() > 0) //while there is stuff in the buffer
  {
    if (gps.encode(ss.read())) //if it can successfully decode it, do it. Else try again
    when more charachters are in the buffer
  }
}
```

```
if (sms.available()) // if a text has been recieved
{
  Serial.println("Message received from:"); // print to the computer
  sms.remoteNumber(senderNumber, 20); // assign the sender number to the
"senderNumber" variable
  Serial.println(senderNumber); // print the sender number to the computer
  password = ""; // flush the temporary variable
  char c;
  while(c=sms.read())
  {
    password += c; // append the sms to the "password" variable
  }
  Serial.println(password); // print the contents of the sms
  Serial.println("\nEND OF MESSAGE"); // print to the computer
  sms.flush(); // delete message from modem buffer
  Serial.println("MESSAGE DELETED"); // print to the computer
  if (password == yourPassword) // if the sms contains the correct password
  {
    Serial.println("\nPASSWORD VALID"); // print to the computer
    digitalWrite(led2, LOW);
    digitalWrite(led1, HIGH);
    sms.beginSMS(senderNumber); // begin an sms to the sender number
    sms.print(gps.location.lat(), 6); // append the lat to the sms
    sms.print(","); // append a comma
    sms.print(gps.location.lng(), 6); // append the lon to the sms
    sms.endSMS(); //send the sms
  }
  else {
    Serial.println("\nPASSWORD NOT VALID"); // print to the computer
    digitalWrite(led1, LOW);
```

```
    digitalWrite(led2, HIGH);
  }
}

t=millis();
if(t >= x+2000){
  x = millis();
  sms.beginSMS(senderNumber);
  txtmsg += String(gps.location.lat(), 6);
  txtmsg += ",";
  txtmsg += String(gps.location.lng(), 6);
  txtmsg += " ";
  txtmsg += x;
  txtmsg += "\n";
  Serial.println(txtmsg);
  sms.print(gps.location.lat(), 6); // append the lat to the sms
  sms.print(","); // append a comma
  sms.print(gps.location.lng(), 6); // append the lon to the sms
  sms.print(" ");
  sms.print(x);
  sms.print("\n");
  sms.endSMS();
}
//delay(1000); // delay
Serial.println("after");
}
```

ΤΕΛΙΚΟΣ ΚΩΔΙΚΑΣ Ε

```

// Κωδικας 5: Το GPS lamvanei sima kathws o kwdikas trexei synexws kai mas
stelnei ana synexws tin topothesia

// Author: Iason-Stylianos Kasapleris

// A.M.: 03112051

#include <AltSoftSerial.h> //Include all relevant libraries

#include <TinyGPS++.h>

#include <GSM.h>

#include <stdio.h>

#define PINNUMBER "" // PIN Number for the SIM - leave blank unless your SIM
has a pin, this is inserted between ""

static const uint32_t GPSBaud = 9600; //Baud rate for communication with the GPS,
Adafruit GPS = 9600, your GPS may well be 4800, check the spec

TinyGPSPlus gps; // The TinyGPS++ object for interfacing with the GPS

AltSoftSerial ss; // The serial connection object to the GPS device

String yourPassword = "Location"; // Put the password here between the ""

String password; // Temporary variable used for comparison of passwords

GSM gsmAccess; // Initialise the library instances

GSM_SMS sms;

char senderNumber[20]="00306947050972"; // Array to hold the number a SMS is
retrieved from

// Array which holds the coordinates and the color of the button pressed.

int led1 = 13;

int led2 = 12;

int i=0;

String txtmsg = "";

unsigned int x = 0;

unsigned int t = 0;

void setup()
{

```



```
pinMode(led1, OUTPUT);
pinMode(led2, OUTPUT);
ss.begin(GPSBaud); // begin the GPS serial connection
Serial.begin(9600); // begin Serial communication with the computer at 9600 baud
rate
Serial.println("Hello"); // Print to the computer
Serial.println("Initialising...");
boolean notConnected = true; // connection state
while(notConnected) // until it connects
{
  if(gsmAccess.begin(PINNUMBER)==GSM_READY) // if it succeeds connecting
    notConnected = false; // connected
  else
  {
    Serial.println("Not connected"); // print to the computer
    delay(1000); //delay
  }
}
Serial.println("GSM initialized"); // print to the computer
Serial.println("Waiting for messages");
}

void loop()
{
  while (ss.available() > 0) //while there is stuff in the buffer
  {
    if (gps.encode(ss.read())) //if it can successfully decode it, do it. Else try again
    when more characters are in the buffer
  }
  if (sms.available()) // if a text has been recieved
  {
    Serial.println("Message received from:"); // print to the computer
  }
}
```

```
sms.remoteNumber(senderNumber, 20); // assign the sender number to the
"senderNumber" variable

Serial.println(senderNumber); // print the sender number to the computer

password = ""; // flush the temporary variable

char c;

while(c=sms.read())

{

    password += c; // append the sms to the "password" variable

}

Serial.println(password); // print the contents of the sms

Serial.println("\nEND OF MESSAGE"); // print to the computer

sms.flush(); // delete message from modem buffer

Serial.println("MESSAGE DELETED"); // print to the computer

if (password == yourPassword) // if the sms contains the correct password

{

    Serial.println("\nPASSWORD VALID"); // print to the computer

    digitalWrite(led2, LOW);

    digitalWrite(led1, HIGH);

    sms.beginSMS(senderNumber); // begin an sms to the sender number

    sms.print(gps.location.lat(), 6); // append the lat to the sms

    sms.print(","); // append a comma

    sms.print(gps.location.lng(), 6); // append the lon to the sms

    sms.endSMS(); //send the sms

}

else {

    Serial.println("\nPASSWORD NOT VALID"); // print to the computer

    digitalWrite(led1, LOW);

    digitalWrite(led2, HIGH);

}

}
```

```
t=millis();
txtmsg += String(gps.location.lat(), 6);
txtmsg += ",";
txtmsg += String(gps.location.lng(), 6);
txtmsg += "\n";
Serial.println(txtmsg);

if(txtmsg.length() > 80 && t >= x+2000){
  x = millis();
  sms.beginSMS(senderNumber);
  txtmsg += x;
  Serial.println(txtmsg);
  sms.print(txtmsg);
  sms.endSMS();
  txtmsg = "";
}
delay(1000); // delay
}
```

## ΠΑΡΑΡΤΗΜΑ Γ

ΚΩΔΙΚΑΣ ΓΙΑ ΕΝΣΩΜΑΤΩΣΗ ΧΑΡΤΗ

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Real Time GPRS based GPS Tracker</title>
    <script src="http://maps.google.com/maps?
file=api&v=1&key=ABQIAAAo5XEKUPvhGcY9vU1QM1PRsbFT8cebFMj-rF7e0WXBj8rVlbsRQYa6IQ9wMzggK1qb79r8MLnzi8LQ" type="text/javascript"></script>
  <!-- Make the document body take up the full screen -->
  <style type="text/css">
    v\:* {behavior:url(#default#VML);}
    html, body {width: 100%; height: 100%;}
    body {margin-top: 0px; margin-right: 0px; margin-left: 0px; margin-bottom: 0px;}
  </style>

  <script type="text/javascript">
    //
    function load(){
      var map = new GMap(document.getElementById("map"));
      var point = new GPoint(0,0);
      map.addControl(new GLargeMapControl());
      map.addControl(new GMapTypeControl());
      map.centerAndZoom(point, 1);
      window.setTimeout(function(){reloadMap(map)},1000);
    }

    function reloadMap(map) {

      var request = GXmlHttp.create();
      request.open("GET", "data.xml", true);
      request.onreadystatechange = function() {
        if (request.readyState == 4) {
          var xmlDoc = request.responseXML;
          var markers = xmlDoc.documentElement.getElementsByTagName("marker");
          for (var i = 0; i &lt; markers.length; i++) {
            var point = new GPoint(parseFloat(markers[i].getAttribute("lng")),
            parseFloat(markers[i].getAttribute("lat")));
            var marker = new GMarker(point);
            map.clearOverlays();
            map.addOverlay(marker);
            map.centerAtLatLng(point);
          }
        }
      }
      request.send(null);
      window.setTimeout(function(){reloadMap(map)},1000);
    }

    // Monitor the window resize event and let the map know when it occurs
    if (window.attachEvent) {
      window.attachEvent("onresize", function() {this.map.onResize()} );
    } else {
      window.addEventListener("resize", function() {this.map.onResize()} , false);
    }

  //]]&gt;
  &lt;/script&gt;
&lt;/head&gt;
&lt;body onload="load()"&gt;
  &lt;div id="map" style="width: 100%; height:100%;"&gt;&lt;/div&gt;

&lt;/body&gt;
&lt;/html&gt;
</pre>
</div>
<div data-bbox="478 921 516 939" data-label="Page-Footer">116</div>
```

## ΠΑΡΑΡΤΗΜΑ Δ

ΚΩΔΙΚΑΣ ΑΙΣΘΗΤΗΡΑ

```
int tonePin = 4; //Tone - Red Jumper
int trigPin = 9; //Trig - violet Jumper
int echoPin = 10; //Echo - yellow Jumper
int clockPin = 11; //IC Pin 11 - white Jumper
int latchPin = 12; //IC Pin 12 - Blue Jumper
int dataPin = 13; //IC Pin 14 - Green Jumper
byte possible_patterns[9] = {
B00000000,
B00000001,
B00000011,
B00000111,
B00001111,
B00011111,
B00111111,
B01111111,
B11111111,
};
int proximity = 0;
int duration;
int distance;
int frequency;

void setup() {
//Serial Port
Serial.begin (9600);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
```

```
pinMode(clockPin, OUTPUT);
pinMode(latchPin, OUTPUT);
pinMode(dataPin, OUTPUT);
pinMode(tonePin, OUTPUT);
}

void loop() {
digitalWrite(latchPin, LOW);
digitalWrite(trigPin, HIGH);
delayMicroseconds(1000);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = (duration / 2) / 29.1;
if (distance >= 45 || distance <= 0){
Serial.println("Out of range");
}
else {
Serial.print(distance);
Serial.println(" cm");
}
proximity = map(distance, 0, 45, 8, 0);
//Serial.println(proximity);
if (proximity <= 0) {
proximity = 0;
}
else if (proximity == 1 || proximity == 2) {
tone(tonePin, 8000, 200);
frequency = 0;
}
else if (proximity == 3 || proximity == 4 || proximity == 5) {
tone(tonePin, 5000, 200);
```

```
frequency = 1;
}
else if (proximity == 6 ) {
tone(tonePin, 1000, 200);
frequency = 2;
}
else if (proximity == 7){
tone(tonePin, 1000, 200);
frequency = 2;
}
else if (proximity == 8){
tone(tonePin, 1000, 1000);
frequency = 2;
}
shiftOut(dataPin, clockPin, MSBFIRST, possible_patterns[proximity]);
digitalWrite(latchPin, HIGH);
if (frequency == 0){
delay(600);
}
else if (frequency == 1){
delay(500);
}
else{
delay(400);
}
//noTone(tonePin);
}
```

## BIBΛΙΟΓΡΑΦΙΑ

- [1] Fleet Management  
[https://en.wikipedia.org/wiki/fleet\\_management](https://en.wikipedia.org/wiki/fleet_management)
- [2] John R. Coffee, Richard W. Rudow, Robert F. Allen, David A. Dye, Kevin M. Marvin, Mark Billings, Mark L. Kirchner, Robert W. Lewis, Robert D. Sleeper, Willialm A. Tekniepe, Vehicle tracking, communication and fleet management system, June 28,2001
- [3] Arduino Uno  
[www.arduino.cc/en/main/arduinoBoardUno](http://www.arduino.cc/en/main/arduinoBoardUno)
- [4] Arduino Uno - Adafruit  
[www.adafruit.com/product/50](http://www.adafruit.com/product/50)
- [5] Ultimate GPS – Adafruit  
[www.adafruit.com/product/746](http://www.adafruit.com/product/746)
- [6] Τηλεματική  
<https://en.wikipedia.org/wiki/Telematics>
- [7] David S. Breed, Telematics system for vehicle diagnostics, May 18, 2004
- [8] David J. Germann, Gregory E. Stewart, Marc Light, Dinkar Mylaraswamy, System and method for asset fleet monitoring and predictive diagnostics using analytics for large and varied data sources, Sep 29, 2016
- [9] Everything about Arduino  
[www.arduino.cc](http://www.arduino.cc)
- [10] Λογισμικό Arduino  
[www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software)
- [11] General Contract of use for wagons  
[http://www.dtd.org.tr/\\_files/uluslararasi/gcuvagonkullanc4b1mc4b1naicc87liscca7kinsocc88zlescca7me11-0.pdf](http://www.dtd.org.tr/_files/uluslararasi/gcuvagonkullanc4b1mc4b1naicc87liscca7kinsocc88zlescca7me11-0.pdf)
- [12] Ενσωματωμένα Συστήματα  
[https://en.wikipedia.org/wiki/Embedded\\_system](https://en.wikipedia.org/wiki/Embedded_system)
- [13] GPS  
[http://en.wikipedia.org/wiki/Global\\_Positioning\\_System](http://en.wikipedia.org/wiki/Global_Positioning_System)
- [14] Geolocation  
<http://en.wikipedia.org/wiki/Geolocation>
- [15] Geocoding  
<http://en.wikipedia.org/wiki/Geocoding>
- [16] Adafruit Microcontroller  
[www.adafruit.it/aMn](http://www.adafruit.it/aMn)



[17] Adafruit Microcontroller  
[www.adafruit.it/d2Q](http://www.adafruit.it/d2Q)

[18] Adafruit Microcontroller  
[www.adafruit.it/aMo](http://www.adafruit.it/aMo)

[19] Adafruit Microcontroller  
[www.adafruit.it/aMp](http://www.adafruit.it/aMp)

[20] Adafruit Microcontroller  
[www.adafruit.it/aPo](http://www.adafruit.it/aPo)

[21] Adafruit Microcontroller  
[www.adafruit.it/aMq](http://www.adafruit.it/aMq)

[22] Adafruit Microcontroller  
[www.adafruit.it/aMr](http://www.adafruit.it/aMr)

[23] Adafruit Microcontroller  
[www.adafruit.it/aTi](http://www.adafruit.it/aTi)

[24] Adafruit Microcontroller  
[www.adafruit.it/aMs](http://www.adafruit.it/aMs)

[25] Adafruit Microcontroller  
[www.adafruit.it/aMu](http://www.adafruit.it/aMu)

[26] Adafruit Microcontroller  
[www.adafruit.it/aMy](http://www.adafruit.it/aMy)

[27] Google Play Real-Time GPS Tracker:  
<https://play.google.com/store/apps/details?id=com.greenalp.RealtimeTracker>

[28] Google Play Family Locator –GPS Tracker:  
<https://play.google.com/store/apps/details?id=com.sygic.familywhere.android>

[29] Google Play Sprint Family Locator:  
<https://play.google.com/store/apps/details?id=com.locationlabs.finder.sprint>

[30] Google Play Life360:  
<https://play.google.com/store/apps/details?id=com.life360.android.safetymapd&hl=el>

[31] Babalis C., GPS shields, 2015

[32] Σύγκριση μικροελεγκτών  
<http://randomnerdtutorials.com/arduino-vs-raspberry-pi-vs-beaglebone-vs-pcduino/>

[33] Σύγκριση μικροελεγκτών  
<https://www.makershed.com/pages/microcontroller-comparison>

- [34] Raspberry pi  
<https://www.raspberrypi.org/>
- [35] BeagleBone black  
<http://beagleboard.org/bone>
- [36] PCduino  
<http://www.linksprite.com/linksprite-pcduino/>
- [37] Goldilocks  
<http://www.robotshop.com/media/files/pdf/user-manual-goldilocks.pdf>
- [38] Intel Galileo  
<https://www.arduino.cc/en/ArduinoCertified/IntelGalileo>
- [39] SparkCore  
<https://www.adafruit.com/product/2127>
- [40] Digispark  
<https://digistump.com/wiki/digispark/tutorials/connecting>
- [41] ebay  
<http://www.ebay.com/>
- [42] skroutz  
<http://www.skroutz.gr/>
- [43] sparkfun  
<http://www.sparkfun.com/>
- [44] Συγκρίσεις GPS trackers  
<http://www.toptenreviews.com/electronics/gps/best-gps-trackers/>
- [45] Συγκρίσεις GPS trackers  
<https://www.lifewire.com/best-gps-trackers-1683621>
- [46] Amber Alert GPS  
[www.amberalertgps.com/](http://www.amberalertgps.com/)
- [47] BrickHouse Security Spark nano GPS  
<http://www.brickhousesecurity.com/product/spark+nano+wireless+gps+tracker.do>
- [48] Trackimo  
<https://trackimo.com/gps-tracker/>
- [49] AngelSense Guardian  
<https://www.angelsense.com/>
- [50] Spy Tec GL-300  
[www.spytecinc.com/gl-200-real-time-gps-tracker.html](http://www.spytecinc.com/gl-200-real-time-gps-tracker.html)

[51] Trax

<https://traxfamily.com/>

[52] Spot Gen 3

<http://andrewskurka.com/2015/spot-gen3-satellite-gps-messenger-review-long-term/>

[53] Yepzon

[www.yepzon.com/](http://www.yepzon.com/)

[54] Amcrest AM-GL300

<https://amcrest.com/am-gl300-real-time-gps-tracker-229.html>

[55] Vectu

[www.aspenta.com/aspenta-product/portable-vehicle-tracker](http://www.aspenta.com/aspenta-product/portable-vehicle-tracker)

[56] PAM-7Q

[https://www.parallax.com/sites/default/files/downloads/28509-PAM-7Q-GPS-Module-Product-Guide-v1.0\\_1.pdf](https://www.parallax.com/sites/default/files/downloads/28509-PAM-7Q-GPS-Module-Product-Guide-v1.0_1.pdf)

[57] PMB-648

<https://www.parallax.com/sites/default/files/downloads/PMB-648-v0.1.pdf>

[58] Surf500F

[https://www.sparkfun.com/datasheets/GPS/Modules/SUP500F\\_v3.pdf](https://www.sparkfun.com/datasheets/GPS/Modules/SUP500F_v3.pdf)

[59] EM406

[https://www.sparkfun.com/commerce/product\\_info.php?products\\_id=465](https://www.sparkfun.com/commerce/product_info.php?products_id=465)

[60] Copernicus

[https://cdn.sparkfun.com/datasheets/Sensors/GPS/63530-10\\_Rev-B\\_Manual\\_Copernicus-II.pdf](https://cdn.sparkfun.com/datasheets/Sensors/GPS/63530-10_Rev-B_Manual_Copernicus-II.pdf)

[61] U-blox 6M

[https://www.u-blox.com/sites/default/.../NEO-6\\_DataSheet\\_\(GPS.G6-HW-09005\).pdf](https://www.u-blox.com/sites/default/.../NEO-6_DataSheet_(GPS.G6-HW-09005).pdf)

[62] Adafruit FONA 808 - GPS Breakout + Mini Cellular GSM

<https://www.adafruit.com/product/2542>

[63] GSM SM5100B-D

<https://www.sparkfun.com/products/9533>

[64] Machine to Machine

[https://en.wikipedia.org/wiki/Machine\\_to\\_machine](https://en.wikipedia.org/wiki/Machine_to_machine)

[65] Arduino GSM shield 2

<https://www.arduino.cc/en/Main/ArduinoGSMShield>

[66] Keyestudio SIM900 GSM/GPRS Module

[www.keyestudio.com/keyestudio-sim900-gsm-gprs-module-shield.html](http://www.keyestudio.com/keyestudio-sim900-gsm-gprs-module-shield.html)

[67] SIM900 Quad-band GSM/GPRS Shield

<https://www.banggood.com/SIM900-Quad-band-GSM-GPRS-Shield-Development-Board-For-Arduino-p-964229.html>

[68] SIMCOM SIM900 Quad-band GSM GPRS Shield

<https://www.amazon.com/Geeetech-SIMCOM-Quad-band-Development-Arduino/dp/B00A8DDYB6>

[69] ITEAD SIM5216E 3G Module WCDMA HSDPA GSM / GPRS /EDGE Shield

<https://www.itead.cc/itead-3g-shield-sim5216j.html>

[70] Adafruit FONA 800 Shield - Voice/Data Cellular GSM

<https://www.adafruit.com/product/2468>

[71] Geeetech SIM900 Quad-band GSM GPRS Shield

<https://www.amazon.com/Geeetech-SIMCOM-Quad-band-Development-Arduino/dp/B00A8DDYB6>

[72] Amazon

[www.amazon.com](http://www.amazon.com)

[73] GSM

<https://en.wikipedia.org/wiki/GSM>

[74] GPRS

[https://en.wikipedia.org/wiki/General\\_Packet\\_Radio\\_Service](https://en.wikipedia.org/wiki/General_Packet_Radio_Service)

[75] Ηλεκτρονικό κατάστημα:

[http://www.espy.gr/siskeii-parakolouthisis-me-karta-SIM-kinitis-tilefonias-\(korios\).html](http://www.espy.gr/siskeii-parakolouthisis-me-karta-SIM-kinitis-tilefonias-(korios).html)

[76] Ηλεκτρονικό κατάστημα:

<http://www.clevermarket.gr/pd/GPS-tracker-doruforikou-entopismou-oximatou-se-pragmatiko-xrono-me-dunatotita-ananeosis-GPS-SIMatos-apomakrusmeni-epanekkinisi-kai-amfidromi-epikoinonia-teleutaio-temaxio-12766.htm?lang=el&path=350448124>

[77] Ηλεκτρονικό κατάστημα:

<http://www.clevermarket.gr/pd/GPS-pet-tracker-mini-entopismou-katoikidion-zoon-anthropon-klp-14575.htm?lang=el&path=-737269747>

[78] Google Play GPS Tracker Car TK SMS:

<https://play.google.com/store/apps/details?id=com.austral.GPScartracker>

[79] Google Play GPS Tracker by SMS -Pro:

<https://play.google.com/store/apps/details?id=br.com.GPStrackerbySMS>

[80] MobileGo

<https://www.tunesbro.com/transfer-sms-from-android-to-computer.html>

[81] Dual-Port RAM

[https://en.wikipedia.org/wiki/Dual-ported\\_RAM](https://en.wikipedia.org/wiki/Dual-ported_RAM)

[82] Synchronous Dual-Port RAMs

<https://www.idt.com/products/memory-logic/multi-port-memory/synchronous-dual-port-rams>

[83] Asynchronous Dual-Port RAMs

<https://www.idt.com/products/memory-logic/multi-port-memory/asynchronous-dual-port-rams>

[84] Διάφορες Dual-Port RAMs

<http://www.futureelectronics.com/en/memory/dual-ports.aspx>

[85] FPGA

[https://en.wikipedia.org/wiki/Field-programmable\\_gate\\_array](https://en.wikipedia.org/wiki/Field-programmable_gate_array)

[86] Alfredo J. Berard, James L. Mentzer, David C. Nixon, Cellular/GPS system for vehicle tracking, Aug 17, 1994

[87] Devyani Bajaj, Neelesh Gupta, GPS Based Automatic Vehicle Tracking Using RFID, January 2012

[88] Michael Austin Halcrow, Dustin Kirkland, On-device mapping of WIFI hotspots via direct connection of WIFI-enabled and GPS-enabled mobile devices, Jan 19, 2006

[89] R. Immanuel Rajkumar ; P. E. Sankaranarayanan ; G. Sundari, GPS and Ethernet based real time train tracking system, Nov 11, 2013

[90] A.J. Elbirt; W. Yip; B. Chetwynd; C. Paar, An FPGA-based performance evaluation of the AES block cipher candidate algorithm finalists, August 2001

[91] AES Algorithm

[https://en.wikipedia.org/wiki/Advanced\\_Encryption\\_Standard](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard)

[92] Arduino GSM Shield 2

<http://www.hobbytronics.co.uk/arduino-gsm-shield-2>

[93] Arduino GSM Shield 2 - Documentation

<http://www.tme.eu/en/Document/fe888cd6b309a17f5b31ca8a0b666ba5/A000105.pdf>

[94] Arduino GSM Shield

<https://www.arduino.cc/en/Guide/ArduinoGSMShield>

[95] Arduino Forum

<http://forum.arduino.cc/index.php?topic=181294.0>

[96] Norman F. Krasner, Combined gps positioning system and communications system utilizing shared circuitry, October 09, 1995

[97] Meir S. Cohen, Eli Finkelman, Vehicle On Board Diagnostic Port Device with GPS Tracking, Auto-Upload, and Remote Manipulation, Jun 21, 2010

[98] GPS with E-Blocks

<http://www.matrixttl.com/webshop/e-blocks-gps-board.html>

**ΤΕΛΟΣ**