Bike Signals

## Πρόβλημα

Η χρήση ποδηλάτου για τις μετακινήσεις μας μέσα στην πόλη μας δεν είναι τόσο διαδεδομένη.

Ένας λόγος που το ποδήλατο δεν συμπεριλαμβάνεται στα βασικά μέσα μεταφοράς στην πόλη, είναι και θέματα ασφάλειας. Λόγω έλλειψης ποδηλατικής παιδείας, οι εν δυνάμει ποδηλάτες αγνοούν βασικά σήματα που αφορούν την προειδοποίηση αυτών που ακολουθούν. Πολύ περισσότερο οι μη ποδηλάτες είναι σχεδόν σίγουρο ότι δεν γνωρίζουν τα βασικά σήματα προειδοποίησης για ενέργειες όπως στάση, στροφή αριστερά, στροφή δεξιά , επιβράδυνση , παραχώρηση προτεραιότητας κ.λπ.

Ένας αυτοματισμός υποβοήθησης του ποδηλάτη με οπτικά και ηχητικά μηνύματα για τη χρήση σημάτων που ισχύουν διεθνώς , καθώς και η αυτόματη ενεργοποίηση κατάλληλων φωτεινών ενδείξεων led (stop, left, right) έτσι ώστε αυτός που ακολουθεί να αντιλαμβάνεται την πρόθεση του ποδηλάτη είναι η αρχική μας σκέψη.

## Μια πρόταση

Η πρότασή μας είναι ένας αυτοματισμός που συνδυάζει τέσσερα υποσυστήματα.

1. Βασικό σύστημα – Arduino Uno
2. Ένα επιταχυνσιόμετρο το οποίο αντιλαμβάνεται την επιβράδυνση, την επιτάχυνση και την κλίση αριστερά, δεξιά του ποδηλάτου.
3. Ένα GPS module το οποίο τοποθετεί χωρικά το ποδήλατο σε χάρτη κίνησης,
   1. καταγράφοντας τη θέση του
   2. και αντλώντας πληροφορίες που αφορούν κίνηση.
4. Ένα Bluetooth module το οποίο είναι ο δίαυλος επικοινωνίας με μια συσκευή τηλεφώνου το οποίο μπορεί να παίξει το ρόλο της πύλης πρόσβασης στο cloud με ταχύτητες που μελλοντικά μπορεί να είναι και αυτές του 5G.
5. Ένα relay και κατάλληλα φωτιστικά στοιχεία τα οποία αποτελούν τη διασύνδεση με το εξωτερικό υποσύστημα.

Ένα απλό σενάριο είναι το εξής:

* Ο ποδηλάτης επιβραδύνει ,
* το σύστημα αντιλαμβάνεται την επιβράδυνση και ταυτόχρονα
  + ενεργοποιεί έξοδο σε relay το οποίο αναβοσβήνει το stop
  + ενεργοποιεί κατάλληλο μήνυμα στο κινητό έτσι ώστε ο ποδηλάτης να κατεβάσει το αριστερό χέρι κάτω(stop).
* Το σύστημα αντιλαμβάνεται την κλίση αριστερά (δεξιά) και ταυτόχρονα
  + ενεργοποιεί έξοδο σε relay το οποίο αναβοσβήνει το αριστερό(δεξιό) LED
  + ενεργοποιεί κατάλληλο μήνυμα στο κινητό έτσι ώστε ο ποδηλάτης με το αριστερό του χέρι να δείχνει αριστερά (πάνω).
* Το σύστημα στέλνει τις συντεταγμένες μέσω ενός MQTT Broker σε μια υπηρεσία καταγραφής και απεικόνισης θέσης . Η αποστολή μπορεί να γίνει χρησιμοποιώντας σαν πύλη το κινητό τηλέφωνο (4G, 5G) ή με LoraWan . Σημειώνουμε ότι το Πανεπιστήμιο Δυτικής Μακεδονίας διαθέτει LoraWan GateWay που είναι διαθέσιμη για πειραματικές ενέργειες.
* Η πρόσβαση σε πληροφορίες θέσης θα μπορεί στη συνέχεια να γίνεται μέσω Internet.

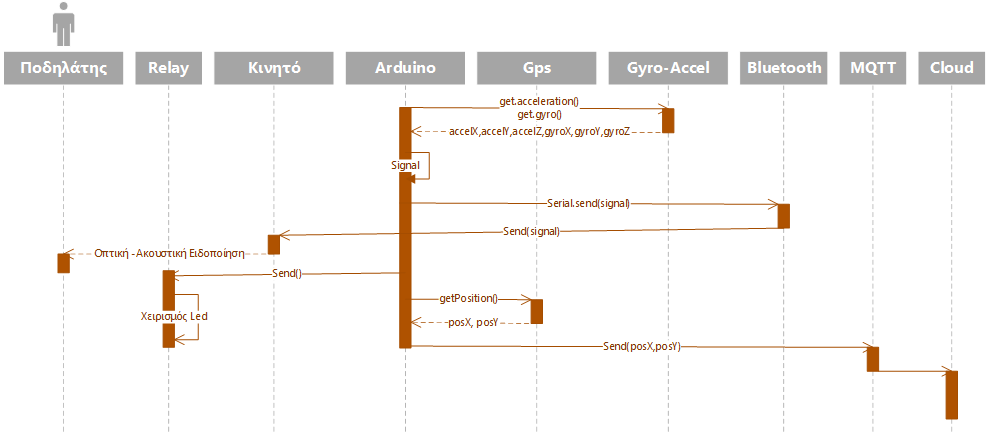


Figure 1 - Ένα σενάριο επικοινωνίας

# Σκοπός μας

Η εφαρμογή μας είναι μια πρόταση να ενταχθεί το ποδήλατο στα μέσα μεταφοράς όσο το δυνατόν περισσότερων πολιτών της πόλης, καθώς και ένας βοηθός εκμάθησης ποδηλατικής συμπεριφοράς στους δρόμους μιας πόλης.

Επίσης είναι μια ευκαιρία να αναζητηθούν τεχνολογίες IoT που θα μπορούν να χρησιμοποιηθούν για πληροφόρηση, απομακρυσμένο έλεγχο, αυτοματοποιημένες ενέργειες από συστήματα που χρησιμοποιούμε καθημερινά.

# Οργάνωση

Στο Project συμμετείχαν τέσσερις μαθητές της Α’ Λυκείου οι οποίοι διδάσκονται το app Inventor και κατασκευές με Arduino, στο μάθημα Εφαρμογές Υπολογιστών.

Ο καθένας αναζήτησε πληροφορίες και υλοποίησε αρχικά ένα ξεχωριστό project.

1. Θωμαϊδου Σοφία – Λειτουργία του GY-521 με Arduino. (Επιταχυνσιόμετρο – Γυροσκόπιο)
2. Αμανατίδου Άννα – Λειτουργία GPS GPS6MU2 με Arduino.
3. Καράμοβ Γρηγόρης – Επικοινωνία Android με Arduino με χρήση του HC-05(Blutooth)
4. Χρυσανίδου Χριστίνα – Μετρήσεις παλμών και κορεσμού οξυγόνου με το MAX30102

Στη συνέχεια έγινε ένα ενιαίο Project το οποίο συμπεριελάμβανε όλες τις παραπάνω λειτουργίες σε ενιαίο κώδικα Arduino το οποίο επικοινωνεί μέσω Bluetooth με Android συσκευή.

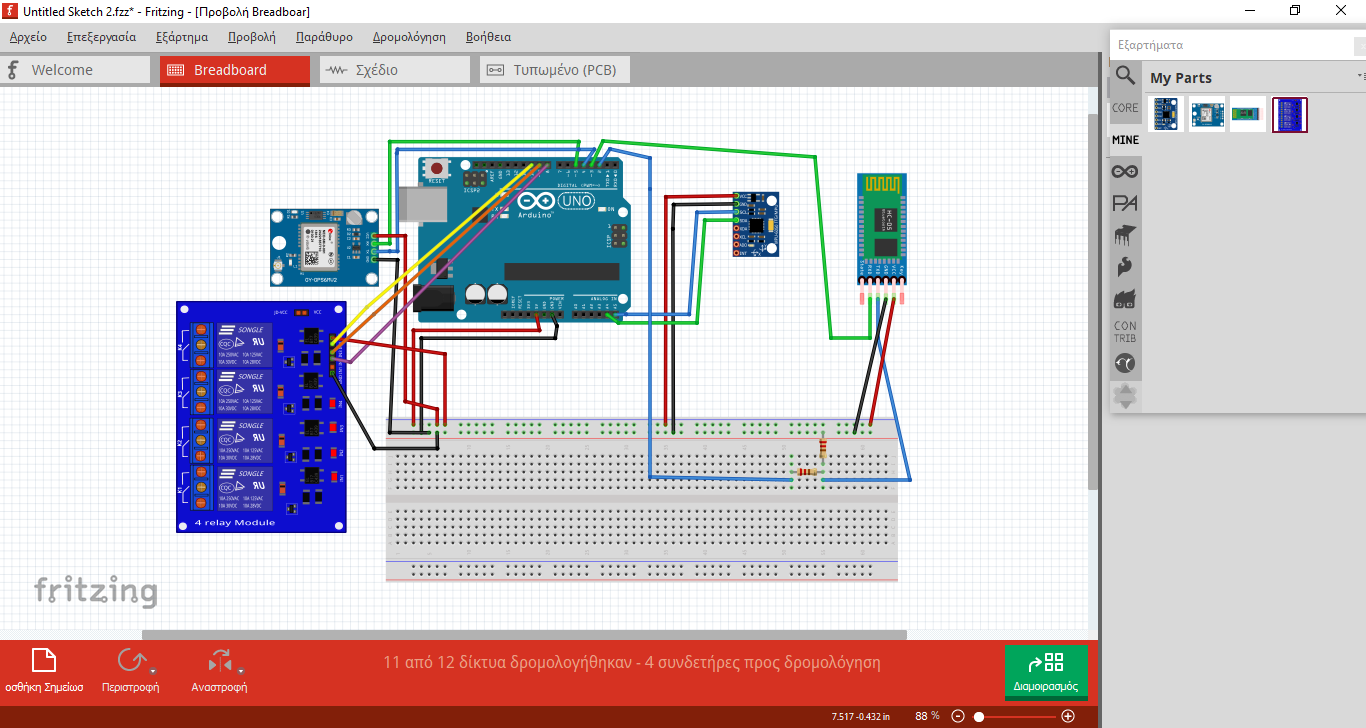
Επιπλέον χωρίστηκαν οι μαθητές σε δύο ομάδες για την διερεύνηση της επέκτασης για επικοινωνία

1. Η 1η ομάδα με LoraWan
2. Η 2η ομάδα με TCP/IP (4G,5G)

# Υλοποίηση

Υλικά που χρησιμοποιήθηκαν

|  |  |  |
| --- | --- | --- |
| Α/Α | Υλικό | Κόστος |
| 1 | Dragino Yun Shield | 32 € |
| 2 | MPU6050 Bluetooth Module | 5 € |
| 3 | NEO 6M GPS module | 13 € |
| 4 | MAX30102 – οξύμετρο, παλμογράφος | 12 € |
| 5 | Dragino LoRa Shield - support 868M frequency with gps | 46 € |
| 6 | 4 Relay Module | 5 € |



# Arduino code

#include <SoftwareSerial.h>

#include <TinyGPS.h>

#include <Adafruit\_MPU6050.h>

#include <Wire.h>

/\* This sample code demonstrates the normal use of a TinyGPS object.

It requires the use of SoftwareSerial, and assumes that you have a

4800-baud serial GPS device hooked up on pins 5(rx) and 4(tx).

\*/

Adafruit\_MPU6050 mpu;

TinyGPS gps;

SoftwareSerial ss(7,6);

SoftwareSerial MyBlue(2, 3); // RX | TX

int flag = 0;

int LED = 8;

int TLeft=9;

int Stop=10;

int TRight=11;

void setup()

{

Serial.begin(115200);

ss.begin(4800);

MyBlue.begin(9600);

pinMode(LED, OUTPUT);

if (!mpu.begin()) {

Serial.println("Failed to find MPU6050 chip");

while (1) {

delay(10);

}

}

Serial.println("MPU6050 Found!");

mpu.setAccelerometerRange(MPU6050\_RANGE\_8\_G);

mpu.setGyroRange(MPU6050\_RANGE\_500\_DEG);

mpu.setFilterBandwidth(MPU6050\_BAND\_21\_HZ);

delay(1000);

}

static void smartdelay(unsigned long ms)

{

unsigned long start = millis();

do

{

while (ss.available())

{

//ss.print(Serial.read());

gps.encode(ss.read());

}

} while (millis() - start < ms);

}

void loop()

{

bool newData = false;

unsigned long chars;

unsigned short sentences, failed;

// For one second we parse GPS data and report some key values

for (unsigned long start = millis(); millis() - start < 1000;)

{

while (ss.available())

{

char c = ss.read();

//Serial.write(c); // uncomment this line if you want to see the GPS data flowing

if (gps.encode(c)) // Did a new valid sentence come in?

newData = true;

}

}

//if (newData)

//{

float flat, flon,falt;

unsigned long age;

gps.f\_get\_position(&flat, &flon, &age);

falt=gps.f\_altitude(); //get altitude

flat == TinyGPS::GPS\_INVALID\_F\_ANGLE ? 0.0 : flat, 6;

flon == TinyGPS::GPS\_INVALID\_F\_ANGLE ? 0.0 : flon, 6;

falt == TinyGPS::GPS\_INVALID\_F\_ANGLE ? 0.0 : falt, 2;//save two decimal places

smartdelay(1000);

Serial.print("LAT=");

Serial.print(flat);

Serial.print(" LON=");

Serial.print(flon);

MyBlue.print(flon);

MyBlue.print("'");

MyBlue.print(flat);

MyBlue.print(",");

MyBlue.print(falt);

delay(1000);

//}

if (MyBlue.available())

flag = MyBlue.read();

if (flag == 1)

{

digitalWrite(LED, HIGH);

}

else if (flag == 0)

{

digitalWrite(LED, LOW);

}

sensors\_event\_t a, g, temp;

mpu.getEvent(&a, &g, &temp);

float dx=0;

float day=0;

dx=a.acceleration.x;

if (dx<0){

digitalWrite(8,HIGH);

MyBlue.print(1);

}

else {

digitalWrite(8,LOW);

MyBlue.print(2);

}

day=g.gyro.x;

if (day<0){

digitalWrite(7,HIGH);

digitalWrite(9,LOW);

MyBlue.print(3);

}

else {

digitalWrite(9,HIGH);

digitalWrite(7,LOW);

MyBlue.print(4);

}

/\* Print out the values \*/

Serial.print("Acceleration X: ");

Serial.print(a.acceleration.x);

Serial.print(", Y: ");

Serial.print(a.acceleration.y);

Serial.print(", Z: ");

Serial.print(a.acceleration.z);

Serial.println(" m/s^2");

Serial.print("Rotation X: ");

Serial.print(g.gyro.x);

Serial.print(", Y: ");

Serial.print(g.gyro.y);

Serial.print(", Z: ");

Serial.print(g.gyro.z);

Serial.println(" rad/s");

Serial.print("Temperature: ");

Serial.print(temp.temperature);

Serial.println(" degC");

Serial.println("");

delay(500);

}

# Πληροφορίες

<https://playground.arduino.cc/Main/MPU-6050/>

<https://create.arduino.cc/projecthub/electropeak/getting-started-with-hc-05-bluetooth-module-arduino-e0ca81>

<https://create.arduino.cc/projecthub/ruchir1674/how-to-interface-gps-module-neo-6m-with-arduino-8f90ad>

<http://arduinolearning.com/code/5-volt-4-channel-arduino-relay-module-example.php>

<https://create.arduino.cc/projecthub/Nicholas_N/how-to-use-the-accelerometer-gyroscope-gy-521-6dfc19>

<http://iot.appinventor.mit.edu/assets/tutorials/MIT_App_Inventor_Basic_Connection.pdf>

<https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/8009-handsignals.pdf>