

Σηματοδότες πόλης με χρήση Arduino

Ρετσέλης Αναστάσιος-Φαίδων
ΑΕΜ: 14648

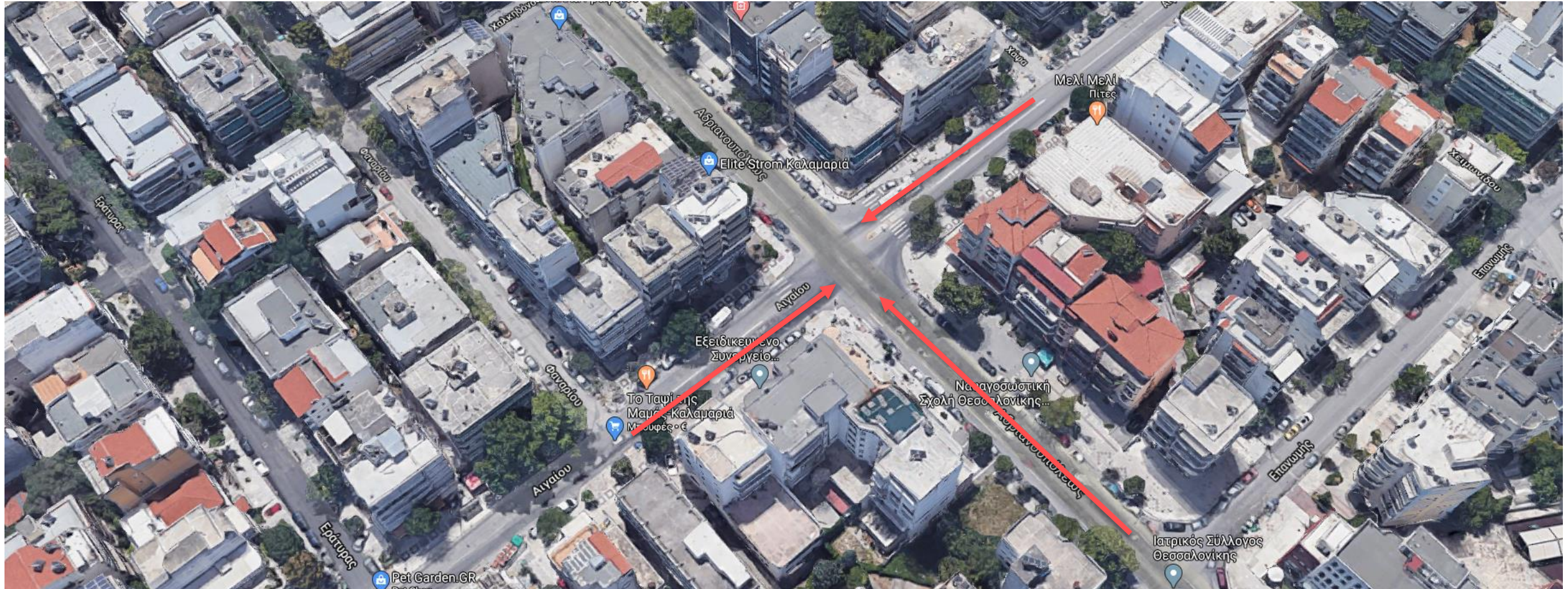


Στόχος project

Η ρεαλιστική αναπαράσταση μιας τυπικής διασταύρωσης με τη χρήση του μικροελεγκτή Arduino



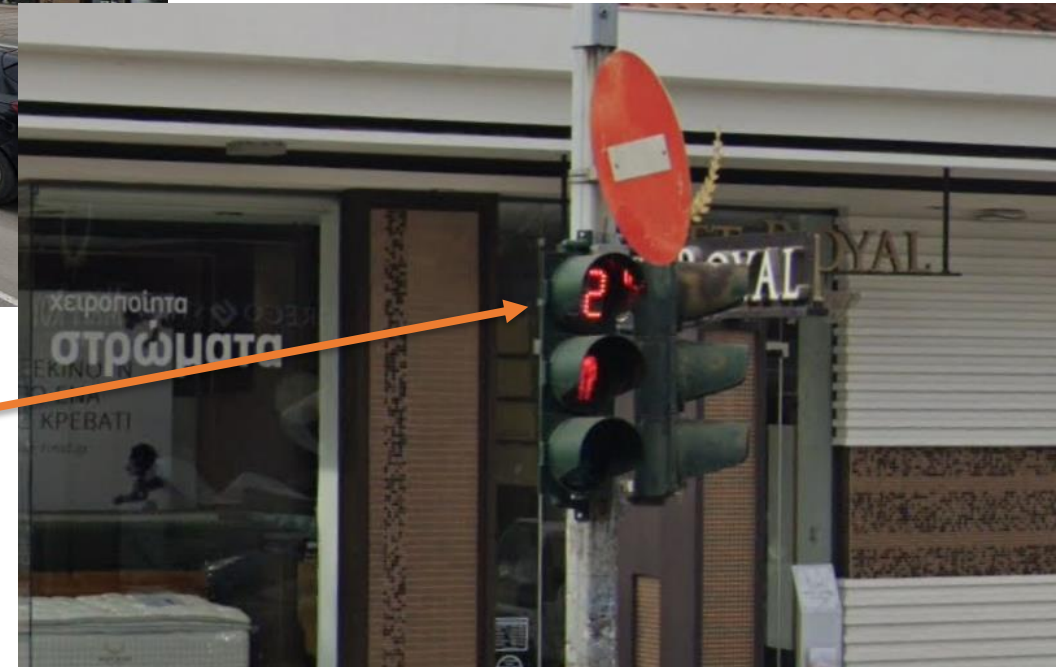
Η διασταύρωση (Αδριανουπόλεως-Αιγαίου)



Η διασταύρωση (Αδριανουπόλεως-Αιγαίου)

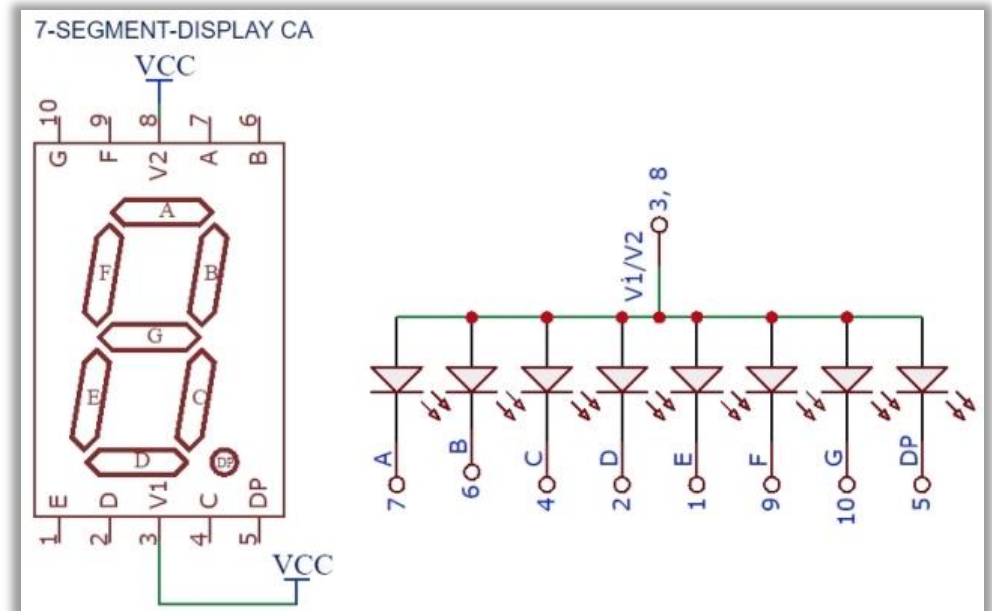
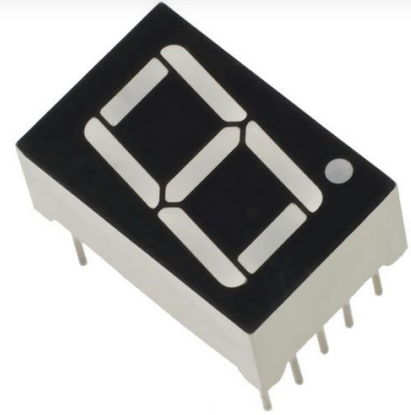


2x 7-segment οθόνες που
δείχνουν πόσος χρόνος
απομένει μέχρι την αλλαγή του
σήματος για τους πεζούς



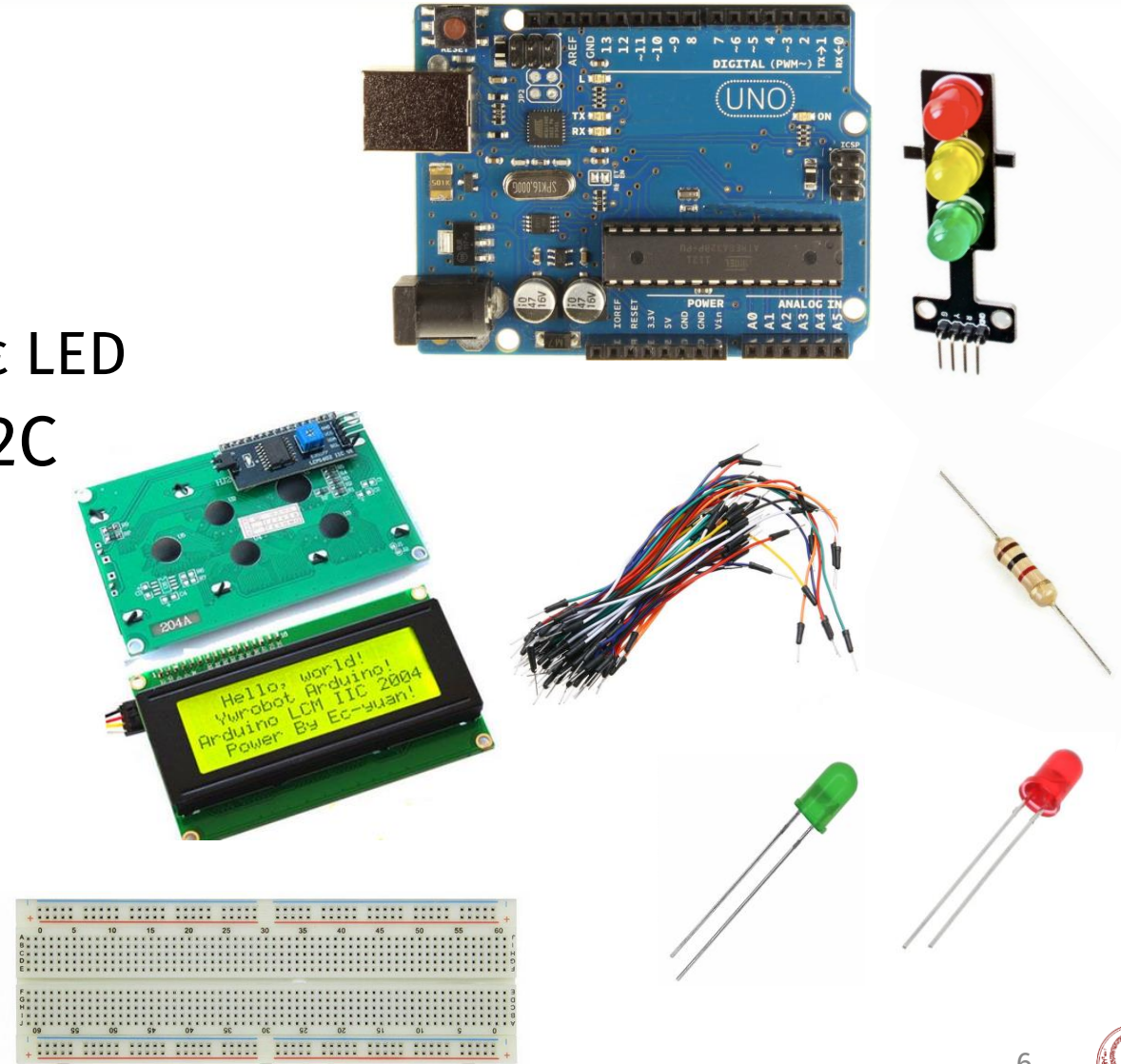
Οθόνες 7-τμημάτων (7-segment displays)

- Για τη σωστή λειτουργία της, απαιτούνται 7 pins τα οποία θα ελέγχονται από το Arduino.
- Πρόβλημα: Μετά την εγκατάσταση των LED για τα φανάρια μας απομένουν μόνο 4 pins ελεύθερα.
- **Λύση**: Χρήση μιας LCD οθόνης η οποία επικοινωνεί με το πρωτόκολλο I2C.

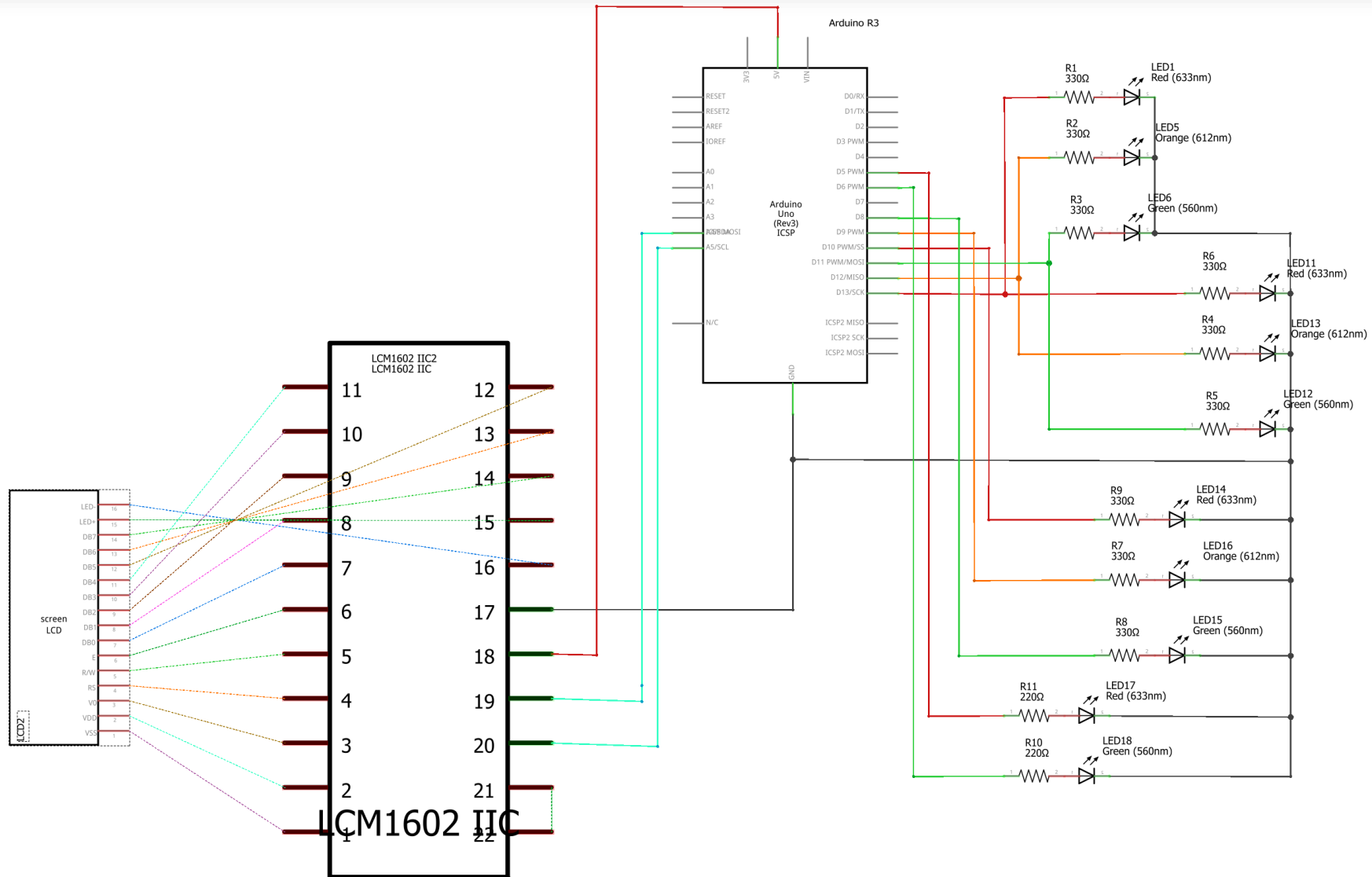


Λίστα υλικών

- 1x Arduino R3
- 3x Traffic Light Modules
 - Με αντίσταση $R=330\ \Omega$ για κάθε LED
- 1x 20x4 LCD Display Module I2C
- 1x Red LED
- 1x Green LED
- 2x 220 Ω Resistors
- 2x Breadboard
- Καλώδια

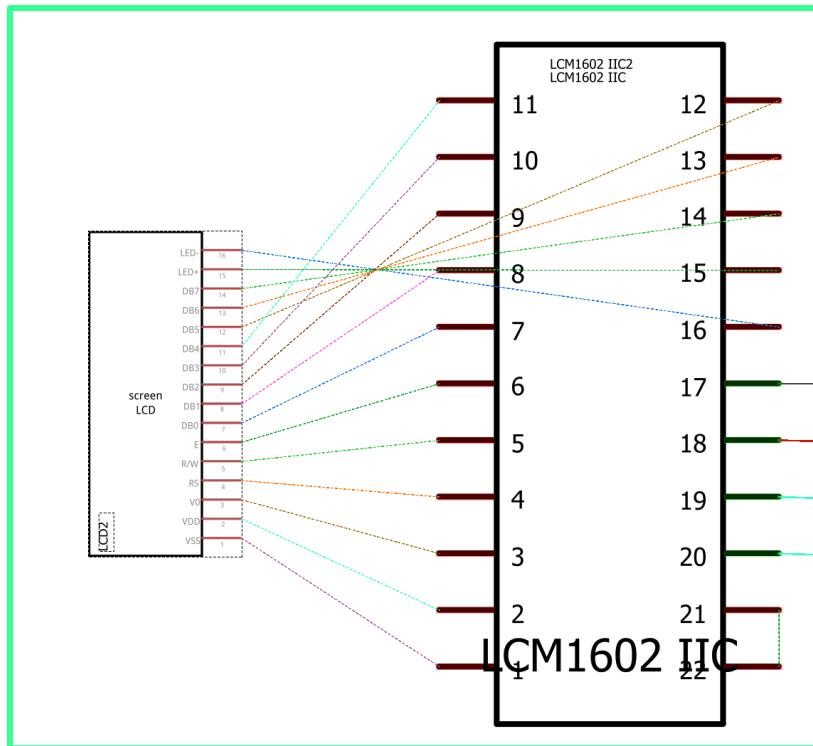


Σχηματικό κυκλώματος

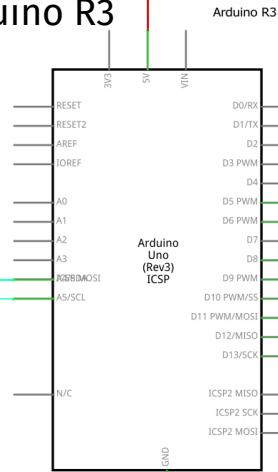


Σχηματικό κυκλώματος

Χρονόμετρο αλλαγής σήματος LCD
(Αδριανουπόλεως, πεζοί)

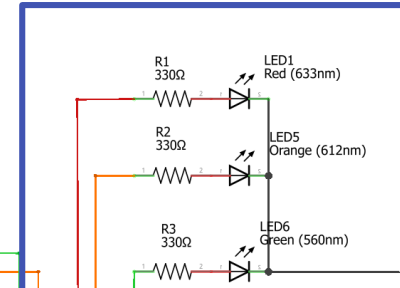


Arduino R3

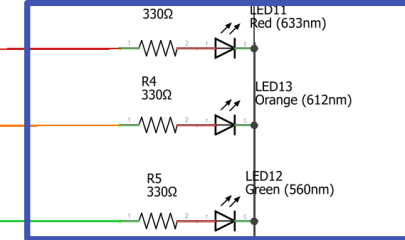


Arduino R3

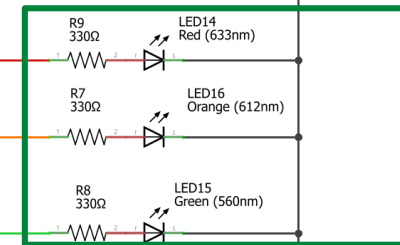
Αιγαίου (1^η κατεύθυνση)



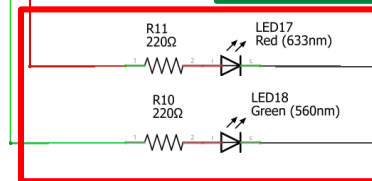
Αιγαίου (2^η κατεύθυνση)



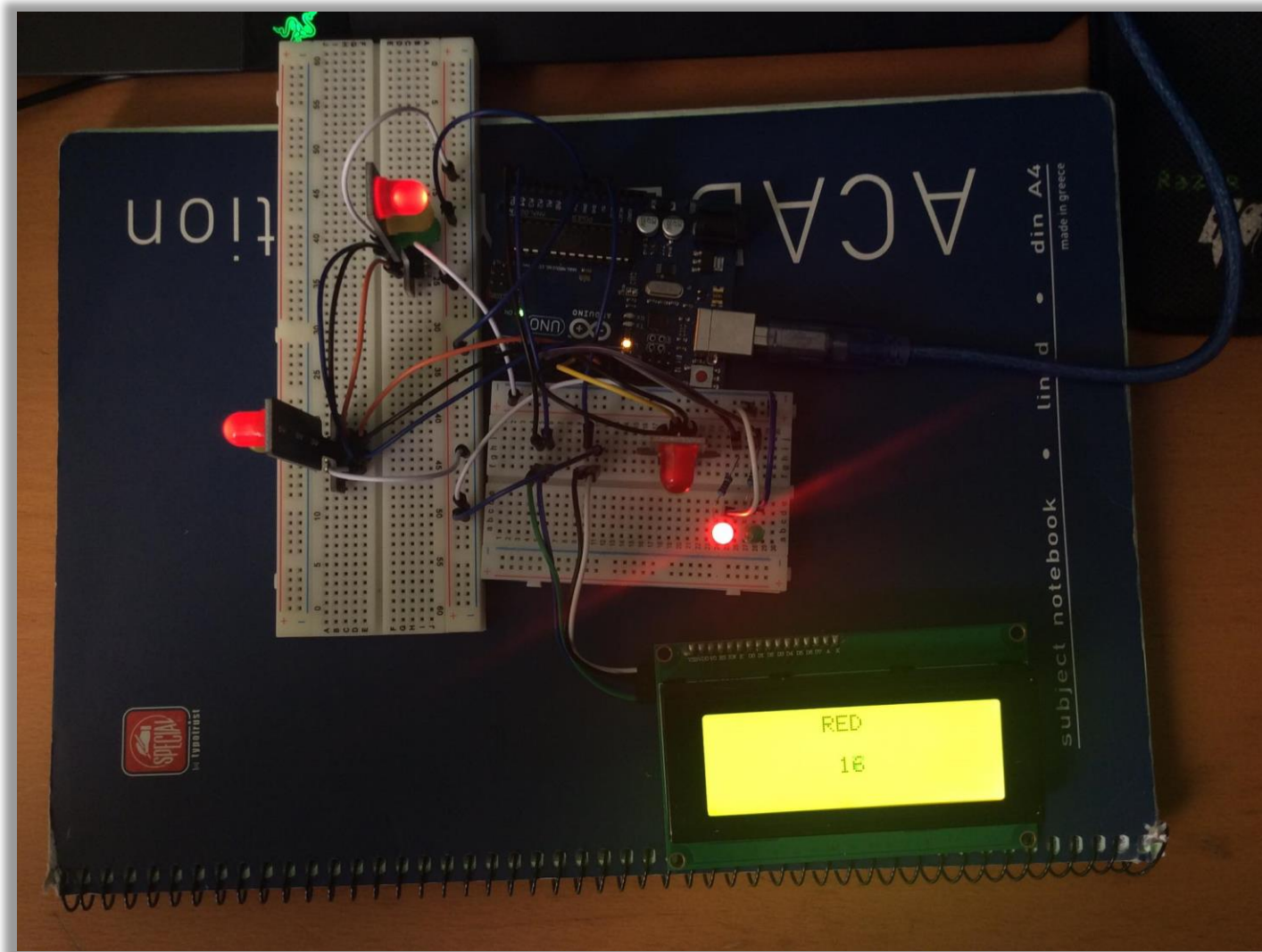
Αδριανουπόλεως



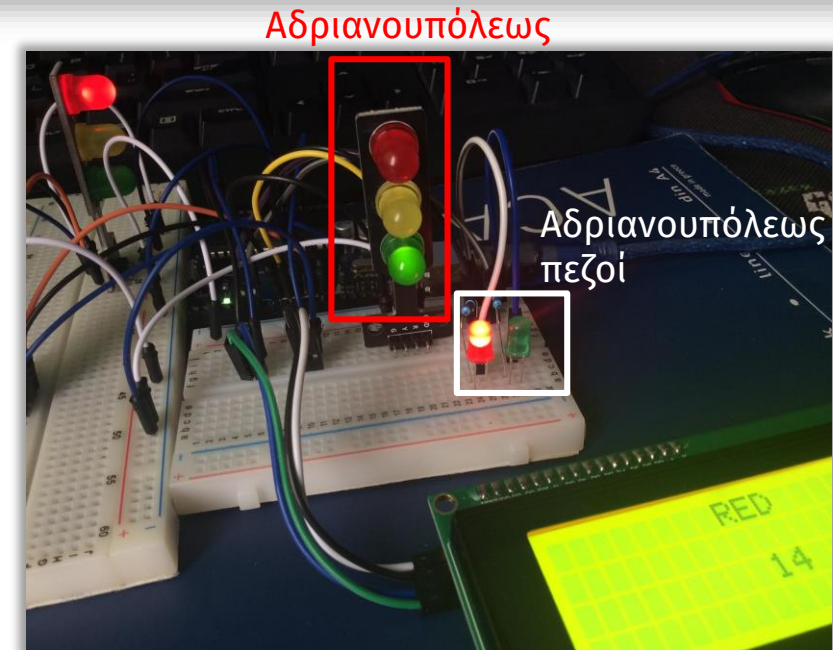
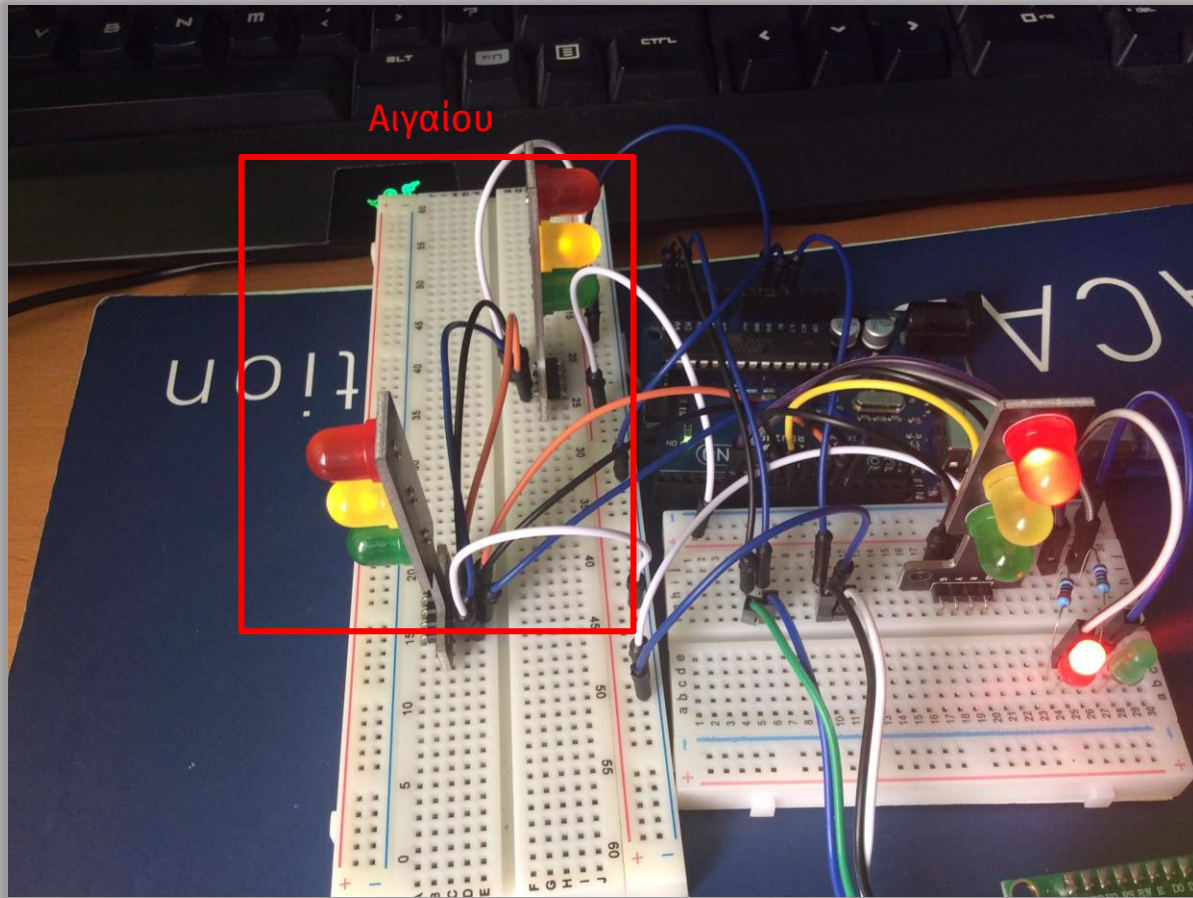
Αδριανουπόλεως (πεζοί)



Η κατασκευή



Η κατασκευή



Κώδικας Arduino

- Βασικές απαιτήσεις από τον κώδικα:
 - ❑ Εύκολη ρύθμιση της διάρκειας κάθε εναλλαγής (για άμεση αλλαγή από την τροχαία εάν απαιτείται)
 - ❑ Χρήση ίδιων pins για την οδό Αιγαίου ώστε να εξοικονομηθούν 3 pins
 - ❑ Μη στιγμιαία εναλλαγή σήματος αυτοκινήτων μετά από την εναλλαγή σε κόκκινο για τους πεζούς, ώστε να έχουν οι πεζοί χρόνο να διασχίσουν την οδό με ασφάλεια
 - ❑ Ένδειξη του χρόνου μέχρι την αλλαγή σήματος για τους πεζού στην LCD οθόνη
 - ❑ Χρήση συναρτήσεων για εύκολες μελλοντικές τροποποιήσεις (π.χ. να ανάβει πορτοκαλί και πριν την αλλαγή από κόκκινο σε πράσινο, όπως συμβαίνει σε πολλές χώρες του εξωτερικού)



Ορισμός pins και setup του Arduino

```
#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>
#define DELAY_ADRIANOUPOLEOS 15
#define DELAY_AIGAIΟΥ 25

LiquidCrystal_I2C lcd(0x27,2,1,0,4,5,6,7);

/*Pin declaration*/

int prasino_aigaiou = 11;
int portokali_aigaiou = 12;
int kokkino_aigaiou = 13;
int prasino_adrianoupoleos = 8;
int portokali_adrianoupoleos = 9;
int kokkino_adrianoupoleos = 10;
int kokkino_pezon_adrianoupoleos = 5;
int prasino_pezon_adrianoupoleos = 6;
int i=0,j=0;

void setup() {
  pinMode(prasino_aigaiou, OUTPUT);
  pinMode(portokali_aigaiou, OUTPUT);
  pinMode(kokkino_aigaiou, OUTPUT);
  pinMode(prasino_adrianoupoleos, OUTPUT);
  pinMode(portokali_adrianoupoleos, OUTPUT);
  pinMode(kokkino_adrianoupoleos, OUTPUT);
  pinMode(kokkino_pezon_adrianoupoleos, OUTPUT);
  pinMode(prasino_pezon_adrianoupoleos, OUTPUT);
  lcd.begin (20,4); // 16 x 2 LCD module
  lcd.setBacklightPin(3,POSITIVE); // BL, BL_POL
  lcd.setBacklight(HIGH);
}
```

Ο κώδικας βρίσκεται διαθέσιμος και ενημερωμένος στο link:

<https://gitlab.com/retse/traffic-light-intersection>



Main loop

```
void loop() {  
    j=DELAY_AIGAIΟΥ-1;  
    adrianoupoleos_sequence();  
    aigaiou_sequence();  
}
```

Ο κώδικας βρίσκεται διαθέσιμος και ενημερωμένος στο link:

<https://gitlab.com/retse/traffic-light-intersection>



Συνάρτηση adrianoupoleos_sequence()

```
void adrianoupoleos_sequence() {  
    digitalWrite(kokkino_pezon_adrianoupoleos, HIGH);  
    digitalWrite(prasino_pezon_adrianoupoleos, LOW);  
    for(i=0;i<DELAY_AIGAIΟΥ-2;i++) {  
        lcd.clear();  
        lcd.setCursor(8,0);  
        lcd.print("RED");  
        lcd.setCursor(9,2);  
        lcd.print(j);  
        delay(1000);  
        j=j-1;}  
    digitalWrite(prasino_adrianoupoleos, LOW );  
    digitalWrite(portokali_adrianoupoleos, HIGH );  
    for(i;i<DELAY_AIGAIΟΥ;i++) {  
        lcd.clear();  
        lcd.setCursor(8,0);  
        lcd.print("RED");  
        lcd.setCursor(9,2);  
        lcd.print(j);  
        delay(1000);  
        j=j-1;}  
    digitalWrite(portokali_adrianoupoleos, LOW );  
    digitalWrite(kokkino_adrianoupoleos, HIGH );  
    digitalWrite(kokkino_aigaiou, LOW );  
    digitalWrite(prasino_aigaiou, HIGH);  
}
```

Ο κώδικας βρίσκεται διαθέσιμος και ενημερωμένος στο link:

<https://gitlab.com/retse/traffic-light-intersection>



Συνάρτηση aigaiou_sequence()

```
void aigaiou_sequence() {  
    digitalWrite(kokkino_pezon_adrianoupoleos, LOW);  
    digitalWrite(prasino_pezon_adrianoupoleos, HIGH );  
    j=DELAY_ADRIANOUPOLEOS-4;  
    for(i=0;i<DELAY_ADRIANOUPOLEOS-4;i++) {  
        lcd.clear();  
        lcd.setCursor(7,0);  
        lcd.print("GREEN");  
        lcd.setCursor(9,2);  
        lcd.print(j);  
        delay(1000);  
        j=j-1;}  
    digitalWrite(kokkino_pezon_adrianoupoleos, HIGH);  
    digitalWrite(prasino_pezon_adrianoupoleos, LOW );  
    j=DELAY_AIGAIU+3;  
    for(i;i<DELAY_ADRIANOUPOLEOS-2;i++) {  
        lcd.clear();  
        lcd.setCursor(8,0);  
        lcd.print("RED");  
        lcd.setCursor(9,2);  
        lcd.print(j);  
        delay(1000);  
        j=j-1;}  
    digitalWrite(prasino_aigaiou, LOW);  
    digitalWrite(portokali_aigaiou, HIGH);  
    for(i;i<DELAY_ADRIANOUPOLEOS;i++) {  
        lcd.clear();  
        lcd.setCursor(8,0);  
        lcd.print("RED");  
        lcd.setCursor(9,2);  
        lcd.print(j);  
        delay(1000);  
        j=j-1;}  
    digitalWrite(portokali_aigaiou, LOW);  
    digitalWrite(kokkino_aigaiou, HIGH );  
    digitalWrite(kokkino_adrianoupoleos, LOW );  
    digitalWrite(prasino_adrianoupoleos, HIGH );  
}
```

Ο κώδικας βρίσκεται διαθέσιμος και ενημερωμένος στο link:

<https://gitlab.com/retse/traffic-light-intersection>



Μελλοντικά βήματα

- Χρήση ηχείου για προειδοποίηση τυφλών
- Προσθήκη του σηματοδότη για τους πεζούς στην Αιγαίου
- Δυνατότητα αλλαγής της διάρκειας του πράσινου σήματος για τα αυτοκίνητα:
 - ❑ Μέσω διαδικτύου (από το κέντρο της διεύθυνσης τροχαίας)
 - ❑ Ανάλογα τη χρονική στιγμή της ημέρας
- Επέκταση δυνατοτήτων για 1, 2 ή 4 κατευθύνσεις

