# Χρησιμοποιώντας την σειριακή οθόνη

#### ΣΚΟΠΟΣ

Να είναι σε θέση οι μαθητές να χρησιμοποιούν τη σειριακή οθόνη, στέλνοντας μηνύματα και τιμές μεταβλητών από ένα πρόγραμμα που έχουμε φορτώσει στη μονάδα μας.

#### ΘΕΩΡΗΤΙΚΟ ΜΕΡΟΣ

#### Τρόπος λειτουργίας σειριακής

Συχνά στις κατασκευές μας θα χρειαστεί να στείλουμε ή να λάβουμε δεδομένα στο Arduino uno. Αυτή η επικοινωνία γίνεται μέσω της σειριακής θύρας του Arduino και πιο συγκεκριμένα μέσω των pins Rx και Tx που υπάρχουν πάνω σε αυτό.

Για να ενεργοποιήσουμε τη σειριακή θύρα επικοινωνίας αρκεί να δώσουμε στη διαδικασία setup() την εντολή Serial.begin(BaudRate), όπου το BaudRate εκφράζει το ρυθμό με τον οποίο θα μεταδίδονται τα bits (μια τιμή στα 9600 είναι συνήθως αρκετή).

Σε προγραμματιστικό επίπεδο, ώστε να μπορέσουμε να πούμε στο Arduino πως θα επικοινωνήσουμε με αυτό, η εντολή που χρησιμοποιούμε είναι η Serial.begin(9600); Η τιμή 9600 αναφέρεται στον ρυθμό μετάδοσης που είπαμε παραπάνω. Την εντολή αυτή την βάζουμε πάντα στο κομμάτι της void setup().

```
void setup()
{
   Serial.begin(9600);
}
```

Για να μας εκτυπώσει το Arduino ένα μήνυμα στην σειριακή οθόνη, μια εντολή που μας βοηθάει σε αυτό είναι η **print()**, που εκτυπώνει ένα μήνυμα ή τιμές ή η **println()** που λειτουργεί ακριβώς το ίδιο αλλά εκτυπώνοντας με αλλαγή γραμμής κάθε φορά.

Για παράδειγμα:

Serial.print("H epikoinwnia ksekinhse"); /\* Θα εμφανίσει το

μήνυμα αυτό στην οθόνη χωρίς να αλλάξει γραμμή μετά \*/

Serial.println(distance); /\* Θα εμφανίσει την τιμή της μεταβλητής

distance σε μια γραμμή \*/

Μπορούμε να χρησιμοποιήσουμε τη σειριακή θύρα στις εφαρμογές για αμφίδρομη επικοινωνία, δηλαδή να στείλουμε και να λάβουμε δεδομένα. Μία απλή περίπτωση χρήση της επικοινωνίας αυτής είναι για εκσφαλμάτωση (debugging) των προγραμμάτων μας, να μπορούμε δηλαδή να δούμε τι τιμές μας δίνουν μετρητές και τι τιμές έχουν οι μεταβλητές μας μέσω της οθόνης σειριακής επικοινωνίας

Όταν έχετε συνδέσει το Arduino σας με τη θύρα USB στον υπολογιστή, η σειριακή οθόνη ενεργοποιείται από το εικονίδιο πάνω δεξιά "Σειριακή Οθόνη", και στο παράθυρο που ανοίγει μπορείτε να βλέπετε όλα τα μηνύματα που στέλνονται από τον κώδικα που έχει φορτωθεί ήδη και τρέχει στην πλακέτα.

### Παράδειγμα χρήσης της σειριακής

Έστω ότι έχουμε τις παρακάτω απαιτήσεις

- 1. Ενεργοποιήστε τη σειριακή θύρα στη συνάρτηση setup() με ρυθμό δεδομένων 9600.
- Στη συνάρτηση loop(), δηλαδή σε κάθε επανάληψη στείλτε ένα μήνυμα της μορφής "Η epikoinwnia ksekinhse:" χωρίς αλλαγή γραμμής.
- 3. Με μια επαναληπτική διαδικασία δώστε τιμές σε μια μεταβλητή από το 1 μέχρι το 20 και τις τιμές αυτές στείλτε τις στη σειριακή οθόνη να εκτυπωθούν σε μια γραμμή η κάθε μία.
- 4. Φτιάξτε ένα πιο σύνθετο μήνυμα χρησιμοποιώντας κείμενο και τιμή μεταβλητής, χρησιμοποιώντας εκτύπωση με και χωρίς αλλαγή γραμμής όταν πρέπει, π.χ. "Twra vlepw thn epanalhpsh : X", 'όπου στο X θα εμφανίζεται η αντίστοιχη τιμή της μεταβλητής σε κάθε επανάληψη.
- Συνδέστε τη μονάδα σας και αφού επιλέξετε τα απαραίτητα (τύπο μονάδας και θύρα), φορτώστε το πρόγραμμά σας.
- Ακολούθως, ενεργοποιήστε τη σειριακή οθόνη και παρακολουθήστε τα μηνύματα που βγαίνουν σε αυτή.
- 7. Προσέξτε ότι δυστυχώς τα ελληνικά δεν εμφανίζονται σωστά.

#### Το πρόγραμμα που υλοποιεί τα παραπάνω είναι:

```
int i;
void setup() {
    Serial.begin(9600);
}
void loop() {
    Serial.print("H epikoinwnia ksekinhse");
for(i=1;i<=20;i++)</pre>
```

```
{
    Serial.println(i);
    delay(100);
}
for(i=1;i<=20;i++) {
    Serial.print("Twra vlepw thn epanalhpsh: ");
    Serial.println(i);
    delay(200);
}
}
```

# ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

# Γραφική παράσταση συνάρτησης

# Συναρτήσεις

Εκτός από τις ενσωματωμένες συναρτήσεις της γλώσσας μπορούμε να φτιάξουμε και εμείς τις δικές μας συναρτήσεις. Οι συναρτήσεις αυτές βοηθούν στο να απλοποιηθεί και να δομηθεί καλύτερα ο κώδικας.

Για παράδειγμα αν θέλουμε να υπολογίζουμε το τετράγωνο ενός αριθμού ο παραπάνω κώδικας ορίζει μία συνάρτηση με όνομα tetragono που κάνει αυτή τη δουλειά.

```
int tetragono( int x)
{
  return x*x;
}
```

Τη χρησιμοποιούμε ακριβώς με τον ίδιο τρόπο όπως τις ενσωματωμένες συναρτήσεις. Αν π.χ. θέλουμε να τυπώσουμε το τετράγωνο του 2 θα γράψουμε Serial.println( tetragono( 2))

Μία συνάρτηση μπορεί να μην επιστρέφει τιμή, να είναι δηλαδή void για παράδειγμα

```
void paula()
{
  digital.write( pinLed, HIGH);
  sleep( 200);
  digital.write( pinLed, LOW);
  sleep( 100);
}
```

# Δημιουργία γραφικής παράστασης

Στη σειριακή θύρα εκτός από να το βλέπουμε τι τυπώνει το Arduino μπορούμε να πάρουμε και μία γραφική παράσταση των αριθμητικών τιμών. Για να δούμε τις γραφικές αυτές παραστάσεις θα πρέπει από το μενού "Εργαλεία" να διαλέξουμε το "Σχεδιογράφος Σειριακής" (υπάρχει στην έκδοση 1.8.5 και μετά. Αν έχετε παλιότερη έκδοση κατεβάστε τη νεώτερη από <u>https://www.arduino.cc/en/main/software</u>) Για παράδειγμα ο παρακάτω κώδικας

for(float i=0; i < 100; i=i+0.5) { Serial.println( i); delay( 5); }

θα παράγει το



### Ενώ ο παρακάτω κώδικας

for(float i=0; i < 100; i=i+0.5) { Serial.println( i\*i); delay( 1); } Θα παράγει το



# <u>Άσκηση:</u>

Να γίνει η γραφική παράσταση του y(x)=x2+2\*x+1 στο διάστημα [-10,10] με χρήση του σχεδιογράφου σειριακής οθόνης και με βήμα 0.1. Η παραπάνω συνάρτηση να υλοποιηθεί σαν συνάρτηση της C με το όνομα grafima. Δεν χρειάζεται να βάλετε κάποια καθυστέρηση.

Προσοχή: Το x² θα το γράψετε σαν x\*x