STEM IB SUPPORT
BY EPAFOS

ΟΔΗΓΙΕΣ ΕΓΚΑΤΑΣΤΑΣΗΣ ARDUINO IDE

Επιτελική Δομή ΕΣΠΑ
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΘΡΗΣΚΕΥΜΑΤΩΝ ΚΑΙ ΑΘΛΗΤΙΣΜΟΥ
ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Υπουργείο Παιδείας, Θρησκευμάτων και Αθλητισμού
Ελλάδα 2.0
ΕΘΝΙΚΟ ΣΧΕΔΙΟ ΑΝΑΚΑΜΨΗΣ ΚΑΙ ΑΝΘΕΚΤΙΚΟΤΗΤΑΣ
Με τη χρηματοδότηση της Ευρωπαϊκής Ένωσης
NextGenerationEU

Περιεχόμενα

1. Εισαγωγή 2
2. Εγκατάσταση Εφαρμογής Arduino IDE 2
3. Εγκατάσταση βιβλιοθηκών 4
4. Αρχικοποίηση Θέσεων R3 (Χειρισμός Μέσω Joystick) 5
5. Ολοκλήρωση Οδηγού 12

2106993525 stem-ib-support.gr helpdesk@stem-ib-support.gr

1. Εισαγωγή

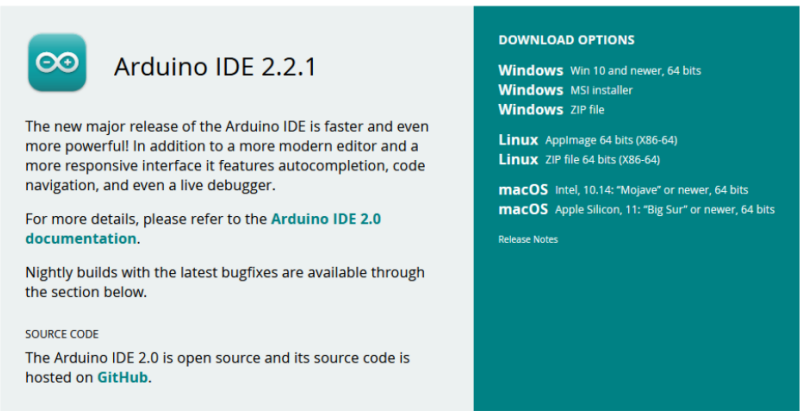
Στόχος του εγγράφου είναι να σας λυθούν απορίες που πιθανώς έχετε αναφορικά με την εγκατάσταση της εφαρμογής και την αρχική της σύνδεση με το ρομπότ.

Για να μπορέσει το ρομπότ να λειτουργήσει το ρομπότ και να εκτελέσει διάφορες λειτουργίες, πρέπει αρχικά να χρησιμοποιήσουμε το πρόγραμμα Arduino IDE, για να δεχτεί το πρόγραμμα με τις αρχικές παραμέτρους.

Σημείωση! Το συγκεκριμένο λογισμικό είναι δύσκολο να χρησιμοποιηθεί από παιδιά δημοτικού και για αυτό δεν προτείνεται για τον προγραμματισμό αυτού του ρομπότ. Ωστόσο, σε περίπτωση που θέλετε να πειραματιστείτε περισσότερο φυσικά μπορείτε να το χρησιμοποιήσετε.

2. Εγκατάσταση Εφαρμογής Arduino IDE

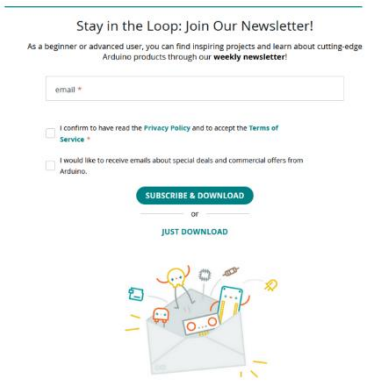
Μπορούμε να βρούμε όλες τις εκδόσεις του λογισμικού Arduino IDE στον παρακάτω σύνδεσμο: https://www.arduino.cc/en/software



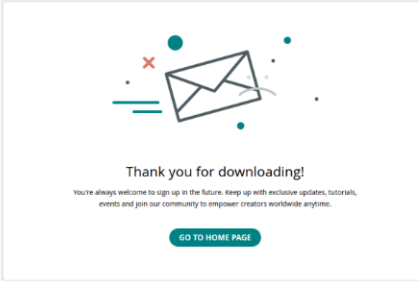
Θα πάρουμε ως παράδειγμα το σύστημα WINDOWS. Κάνουμε κλικ στην επιλογή "Windows zip file" , και εμφανίζεται η παρακάτω σελίδα. Επιλέγουμε "JUST DOWNLOAD"

2106993525 stem-ib-support.gr helpdesk@stem-ib-support.gr

Στην επόμενη οθόνη επιλέγουμε “JUST DOWNLOAD”



Θα εμφανιστεί η επόμενη οθόνη και ο συμπιεσμένος φάκελος του οδηγού εγκατάστασης της εφαρμογής θα αρχίσει να κατεβαίνει στον υπολογιστή μας στον φάκελο των λήψεων, εμφανίζοντας το μήνυμα “Thank you for downloading!”.

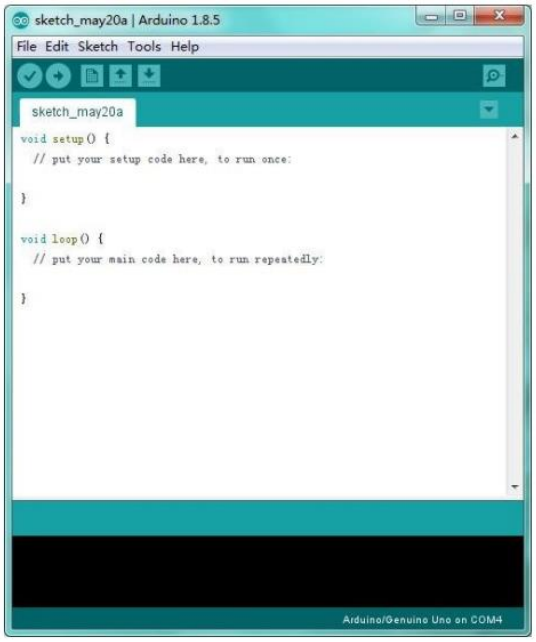


Κάνουμε αποκοπή του φακέλου και επικόλληση του στον φάκελο «Έγγραφα» του υπολογιστή. Αποσυμπιέζουμε το αρχείο.

Έχουμε ολοκληρώσει την εγκατάσταση του Arduino IDE!

Στην επιφάνεια εργασίας του υπολογιστή μας, θα εμφανιστεί το εικονίδιο του Arduino IDE. Αν δεν εμφανιστεί, κάνουμε αναζήτηση του προγράμματος από την «Έναρξη των Windows”.

Κάνουμε διπλό κλικ στο εικονίδιο του Arduino IDE, για να εισέλθουμε στο επιθυμητό περιβάλλον ανάπτυξης, που φαίνεται παρακάτω.



3. Εγκατάσταση βιβλιοθηκών

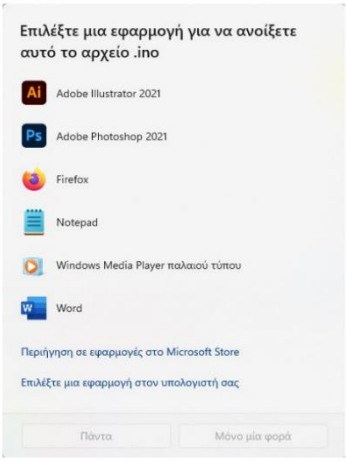
Για να μπορέσετε να τρέξετε τον αρχικό κώδικα στο ρομπότ R3, θα πρέπει να επικοινωνήσετε με την εταιρεία Polytech για να σας αποστείλει την ομάδα των βιβλιοθηκών που θα χρειαστείτε. Εφόσον σας αποσταλούν τα αρχεία, ακολουθείτε τα παρακάτω βήματα:

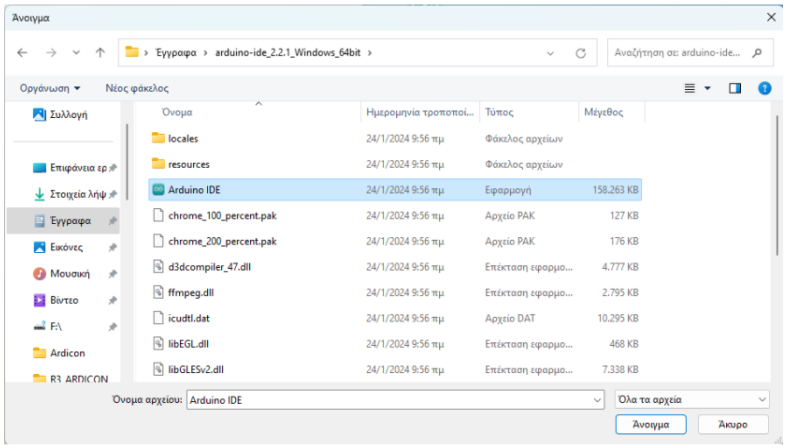
- Στον φάκελο εφαρμογών του ρομπότ, ανοίγουμε τον φάκελο “R3_Codes - Libraries”.
- Στη συνέχεια, ανοίγουμε τον φάκελο “Libraries”. Αντιγράφουμε όλα τα αρχεία που θα μας αποστείλουν μέσα στον υπο-φάκελο “Libraries” του ARDUINO IDE. Τον φάκελο αυτόν τον δημιουργούμε εμείς ώστε να δημιουργηθεί η διαδρομή «Έγγραφα» —> «ARDUINO» —> «Libraries».
- Στη συνέχεια, τρέχουμε το αρχείο INO “R3_setup”, που βρίσκεται στον φάκελο “R3_Code_Libraries”.

Σε περίπτωση που ο υπολογιστής μας δεν αναγνωρίζει τα αρχεία .INO ακολουθούμε την παρακάτω διαδικασία:

Τρέχουμε το αρχείο και στο παράθυρο που ανοίγει, επιλέγουμε την επιλογή “Επιλέξτε μια εφαρμογή στον υπολογιστή σας”.

Στο επόμενο παράθυρο, βρίσκουμε την εφαρμογή ARDUINO IDE μέσα στα αρχεία της εφαρμογής που έχουμε αποθηκεύσει στο φάκελο “Έγγραφα” του υπολογιστή, την επιλέγουμε και στη συνέχεια πατάμε «Άνοιγμα». Στο επόμενο παράθυρο επιλέγουμε “Πάντα”.





4. Αρχικοποίηση Θέσεων R3 (Χειρισμός Μέσω Joystick)

4.1. Κώδικας Αρχικοποίησης

Θα πρέπει να προχωρήσουμε στην αρχικοποίηση του ρομπότ μέσα από το λογισμικό Arduino IDE. Τον τρόπο εγκατάστασης του προγράμματος αρχικοποίησης μπορούμε να τα βρούμε στο χρήσιμα έγγραφα του ρομποτικού κιτ. Η διαδικασία αυτή, είναι απαραίτητη για να ρυθμιστούν/οριστούν σωστά τα σέρβο (αριστερό, δεξί και σέρβο δαγκάνας).

Εφόσον έχουμε εγκαταστήσει το λογισμικό Arduino IDE:

- Συνδέουμε το ρομπότ με τον υπολογιστή μας. Συνδέουμε το ρομπότ στην τροφοδοσία και πατάμε το κουμπί reset του ρομπότ.
- Μπορούμε είτε να γράψουμε τον κώδικα στο πρόγραμμα ARDUINO IDE, ή να τρέξουμε απευθείας το αρχείο INO “R3_Setup”, ακολουθώντας τη διαδρομή R3 CODES-LIBRARIES --> R3_Setup --> R3_Setup (Θα πρέπει να επικοινωνήσετε με την εταιρεία polytech για να λάβετε αυτό το αρχείο)

4.1.1. Κώδικας

```
**********************************************************************************

//R3 Set Up

#include <Servo.h> // add the servo libraries

Servo myservo1; // create servo object to control a servo Servo myservo2;

Servo myservo3; Servo myservo4;

const int pos1_initial = 90; const int pos2_initial = 0; const int pos3_initial = 129; const int pos4_initial
= 0;

int pos1 = pos1_initial; int pos2 = pos2_initial; int pos3 = pos3_initial; int pos4 = pos4_initial;

void setup()

{

// boot posture

myservo1.attach(A1); // set the control pin of servo 1 to A1 myservo1.write(pos1_initial);

myservo2.attach(A0); // set the control pin of servo 2 to A0 myservo2.write(pos2_initial);

myservo3.attach(6);      //set the control pin of servo 3 to D6 myservo3.write(pos3_initial);

myservo4.attach(9);      // set the control pin of servo 4 to D9 myservo4.write(pos4_initial);

Serial.begin(9600); // set the baud rate to 9600

}

void loop()
```

```
{

}
**********************************************************************
```

4.2. Φόρτωση τελικού κώδικα ρομπότ (Χρήση R3 app)

Για να προχωρήσουμε σε χρήση της εφαρμογής R3 για υπολογιστές, θα πρέπει να προχωρήσουμε στην φόρτωση ενός κώδικα στο ρομπότ μέσα από το λογισμικό Arduino IDE. Τον τρόπο εγκατάστασης της εφαρμογής R3, μπορούμε να τη βρούμε στα χρήσιμα έγγραφα του ρομποτικού κιτ.

- Συνδέουμε το ρομπότ με τον υπολογιστή μας. Συνδέουμε το ρομπότ στην τροφοδοσία και πατάμε το κουμπί reset του ρομπότ.
- Μπορούμε είτε να γράψουμε τον κώδικα στο πρόγραμμα ARDUINO IDE, ή να τρέξουμε απευθείας το αρχείο INO “R3_Final”, ακολουθώντας τη διαδρομή R3 CODES-LIBRARIES --> R3_Final --> R3_Final (Θα πρέπει να επικοινωνήσετε με την εταιρεία polytech για να λάβετε αυτό το αρχείο).

4.2.1. Κώδικας R3 APP


```
**************************************************************************
#include <Servo.h> // add the servo libraries

Servo myservo1; // create servo object to control a servo Servo myservo2;

Servo myservo3; Servo myservo4;

const int pos1_initial = 90; const int pos2_initial = 0; const int pos3_initial = 129; const int pos4_initial
= 0;

int pos1 = pos1_initial; int pos2 = pos2_initial; int pos3 = pos3_initial; int pos4 = pos4_initial;

const int right_X = A2; // define the right X pin to A2 const int right_Y = A5; // define the right Y pin to
A5

const int right_key = 7; // define the right key pin to 7（that is the value of Z）

const int left_X = A3; // define the left X pin to A3 const int left_Y = A4; // define the left X pin to A4

const int left_key = 8; //define the left key pin to 8（that is the value of Z）

int x1, y1, z1; // define the variable, used to save the joystick value it read. int x2, y2, z2;

unsigned long previousMillis = 0; unsigned long currentMillis = 0; const int interval = 1000;

const int numValues = 4; // number of values to receive byte buffer[256]; // buffer to hold incoming
data
```

```
int values[numValues]; // array to hold parsed values
void setup()
{
// boot posture
myservo1.attach(A1); // set the control pin of servo 1 to A1 myservo1.write(pos1_initial);
myservo2.attach(A0); // set the control pin of servo 2 to A0 myservo2.write(pos2_initial);
myservo3.attach(6);      //set the control pin of servo 3 to D6 myservo3.write(pos3_initial);
myservo4.attach(9);      // set the control pin of servo 4 to D9 myservo4.write(pos4_initial);
pinMode(right_key, INPUT);      // set the right/left key to INPUT pinMode(left_key, INPUT);
Serial.begin(9600); // set the baud rate to 9600
}
void loop()
{
x1 = analogRead(right_X); y1 = analogRead(right_Y); z1 = digitalRead(right_key);
x2 = analogRead(left_X); y2 = analogRead(left_Y); z2 = digitalRead(left_key);
```



```
// rotate zhuandong();
// upper arm xiaobi();
//lower arm dabi();
// claw zhuazi(); readPosition();
}

void readPosition() { currentMillis = millis();
// check to see if the interval time is passed. if (currentMillis - previousMillis >= interval) {
//print to serial monitor to see number results Serial.print("$"); Serial.print(myservo1.read());
Serial.print("#"); Serial.print(myservo2.read()); Serial.print("#"); Serial.print(myservo3.read());
Serial.print("#"); Serial.print(myservo4.read()); Serial.println("$");
// save the time when we displayed the string for the last time

previousMillis = currentMillis;
```

```
}
}

//*****************************************************
// turn
void zhuandong()
{
if (x1 < 50) // if push the right joystick to the right
{
pos1--; //pos1 subtracts 1
if (pos1 < 46) // limit the angle when turn right
{
pos1 = 45;
}
}
if (x1 > 1000) // if push the right joystick to the let
{
pos1++; //pos1 plus 1
if (pos1 > 134) // limit the angle when turn left
{
pos1 = 135;
}
}
myservo1.write(pos1); delay(2.5);
}
//*********************************************************/
```


2106993525 stem-ib-support.gr helpdesk@stem-ib-support.gr

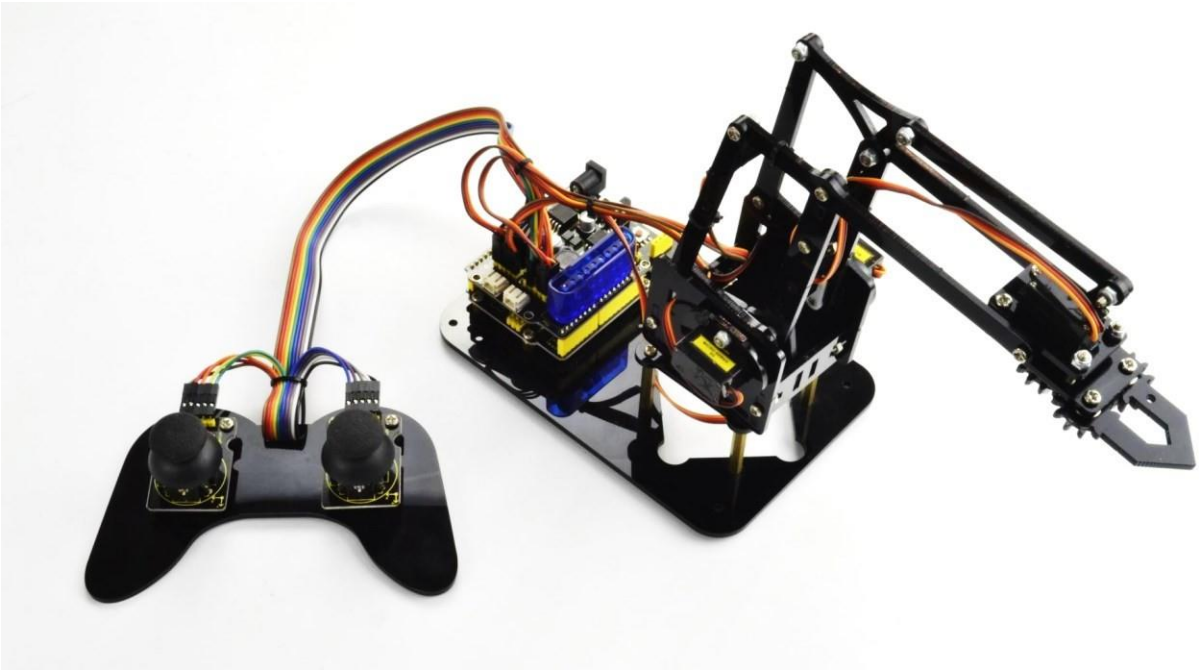
```
//upper arm void xiaobi()
{
if (y1 > 1000) // if push the right joystick upward
{
pos2--;
if (pos2 < 0) // limit the lifting angle
{
pos2 = 0;
}
}
if (y1 < 50) // if push the right joystick downward
{
pos2++;
if (pos2 > 79) // limit the angle when go down
{
pos2 = 80;
}
}
myservo2.write(pos2); delay(1);
}
```

```
//*************************************************************/
// lower arm void dabi()
{
if (y2 < 50) // if push the left joystick upward
{
pos3++;
if (pos3 > 144) // limit the stretched angle
{
pos3 = 145;
}
}
if (y2 > 1000) // if push the left joystick downward
{
pos3--;
if (pos3 < 36) // limit the retracted angle
{
pos3 = 35;
}
}
myservo3.write(pos3); delay(1);
}
```

```
//*************************************************************/
//claw
void zhuazi()
{
//claw
if (x2 < 50) // if push the left joystick to the right
{
pos4--;
if (pos4 < 2) // if pos4 value subtracts to 2, the claw in 37 degrees we have tested is closed.
{ //（should change the value based on the fact）
pos4 = 0; // stop subtraction when reduce to 2
}
}
if (x2 > 1000) //// if push the left joystick to the left
{
pos4++; // current angle of servo 4 plus 8（change the value you plus, thus change the open speed of claw）
if (pos4 > 107) // limit the largest angle when open the claw
{
pos4 = 60;
}
}
myservo4.write(pos4); delay(2.5);
```

5. Ολοκλήρωση Οδηγού

Συγχαρητήρια! Το κιτ του ρομποτικού βραχίονα έχει ολοκληρωθεί!



Ολοκληρώσαμε την καλωδίωση, την αρχικοποίηση των θέσεων του ρομπότ και φορτώσαμε τον κώδικα λειτουργίας του για την εφαρμογή R3.