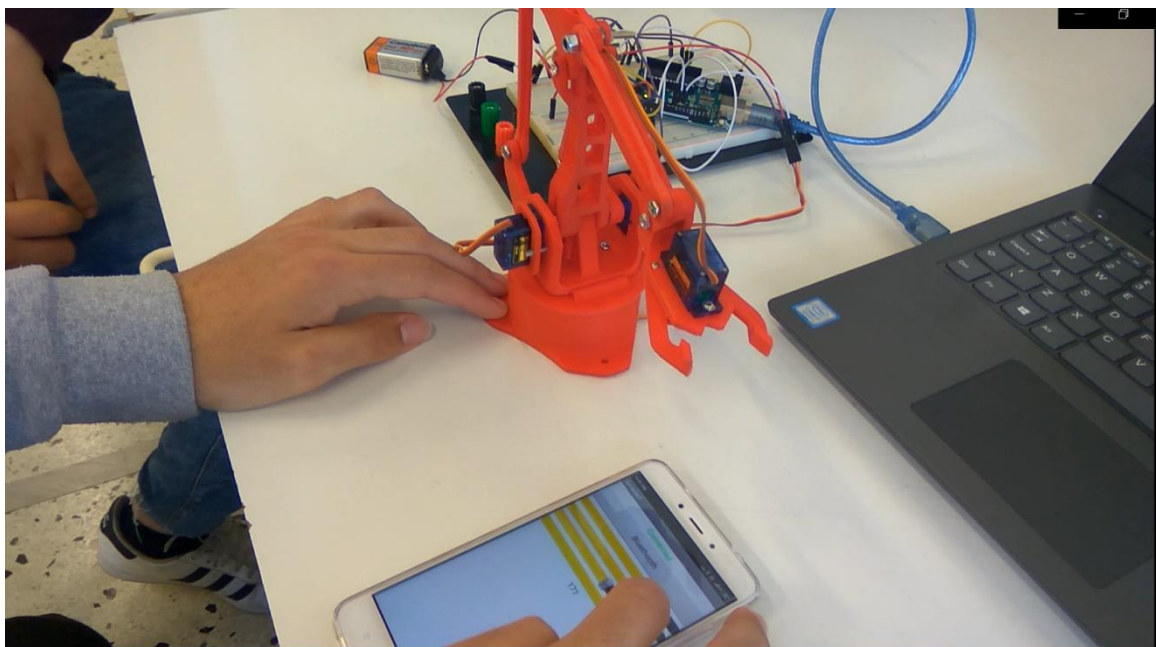


ΡΟΜΠΟΤΙΚΟΣ ΒΡΑΧΙΟΝΑΣ

Έχοντας στη διάθεση μας 3D εκτυπωτή, που επίσης αποτελεί φετινό project της ομάδας μας, σκεφτήκαμε να εκτυπώσουμε εμείς έναν βραχίονα. Εκτυπώσαμε λοιπόν τα διάφορα μέρη του και τον συναρμολογήσαμε. Ο βραχίονας μας έχει επιπλέον την δυνατότητα κίνησης μπροστά- πίσω καθώς και της περιστροφής. Έτσι χρησιμοποιεί 4 servo κινητήρες. Αρχικά έγινε η δοκιμή του μεμονωμένα με χρήση android εφαρμογής που υλοποιήσαμε ειδικά για αυτό τον σκοπό.



Δείτε το βίντεο : <https://www.youtube.com/watch?v=e6VJtG1KDQw&feature=youtu.be>

Περιγραφή του project:

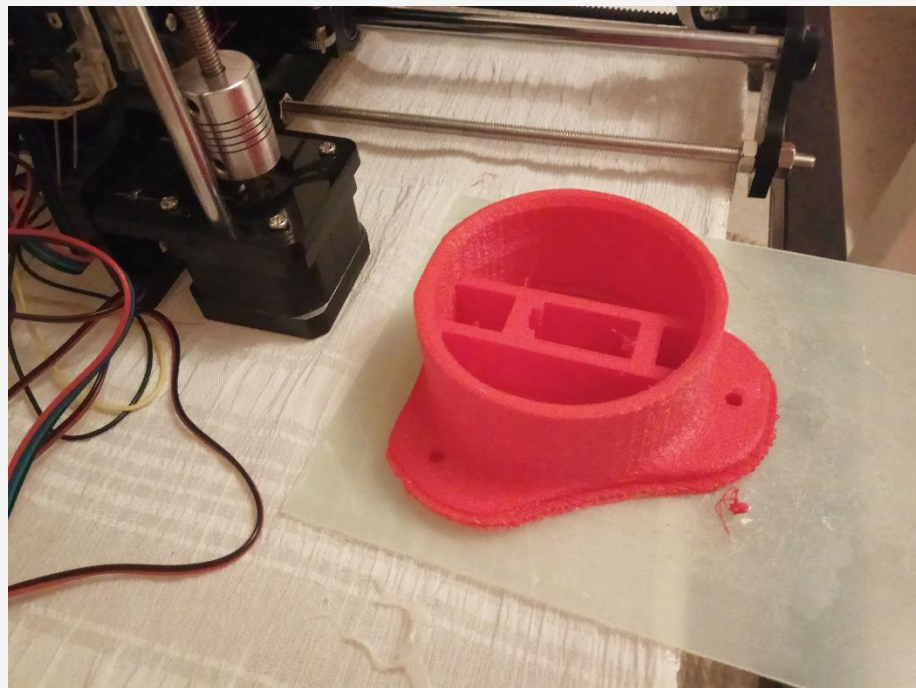
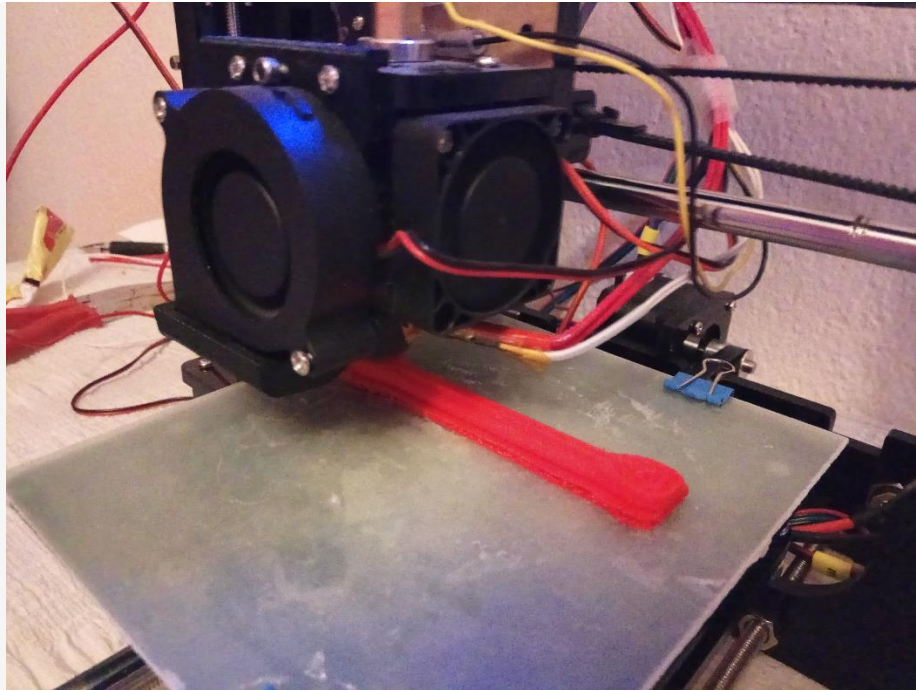
1^ο βήμα: 3D εκτύπωση και συναρμολόγηση

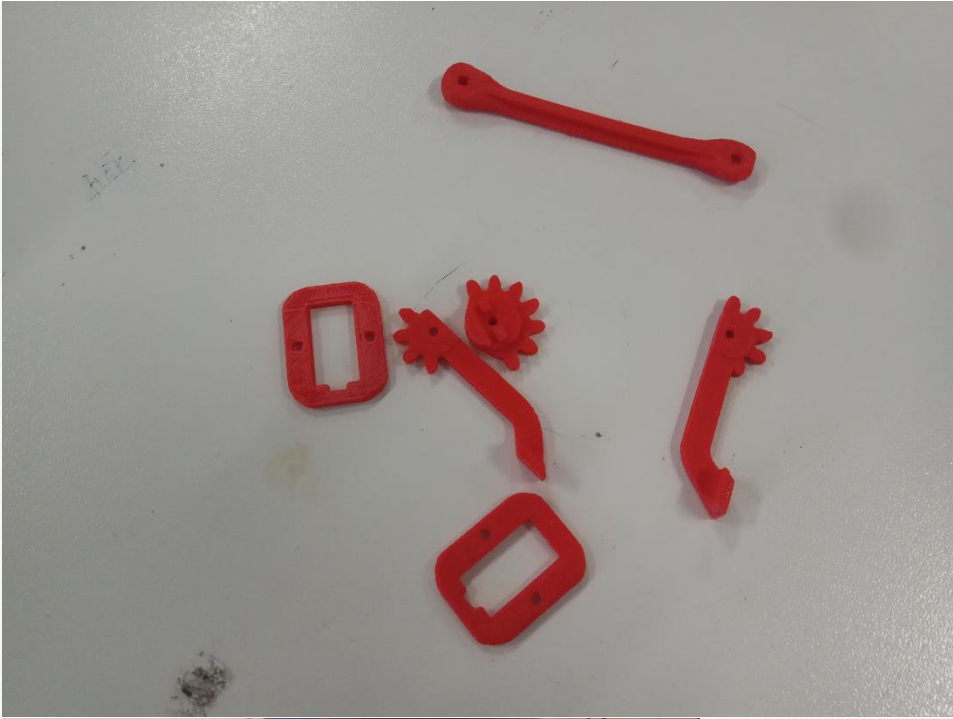
Αρχικά κατεβάσαμε τα σχέδια του βραχίονα από την thingiverse, συγκεκριμένα από τη διεύθυνση <https://www.thingiverse.com/thing:1015238>

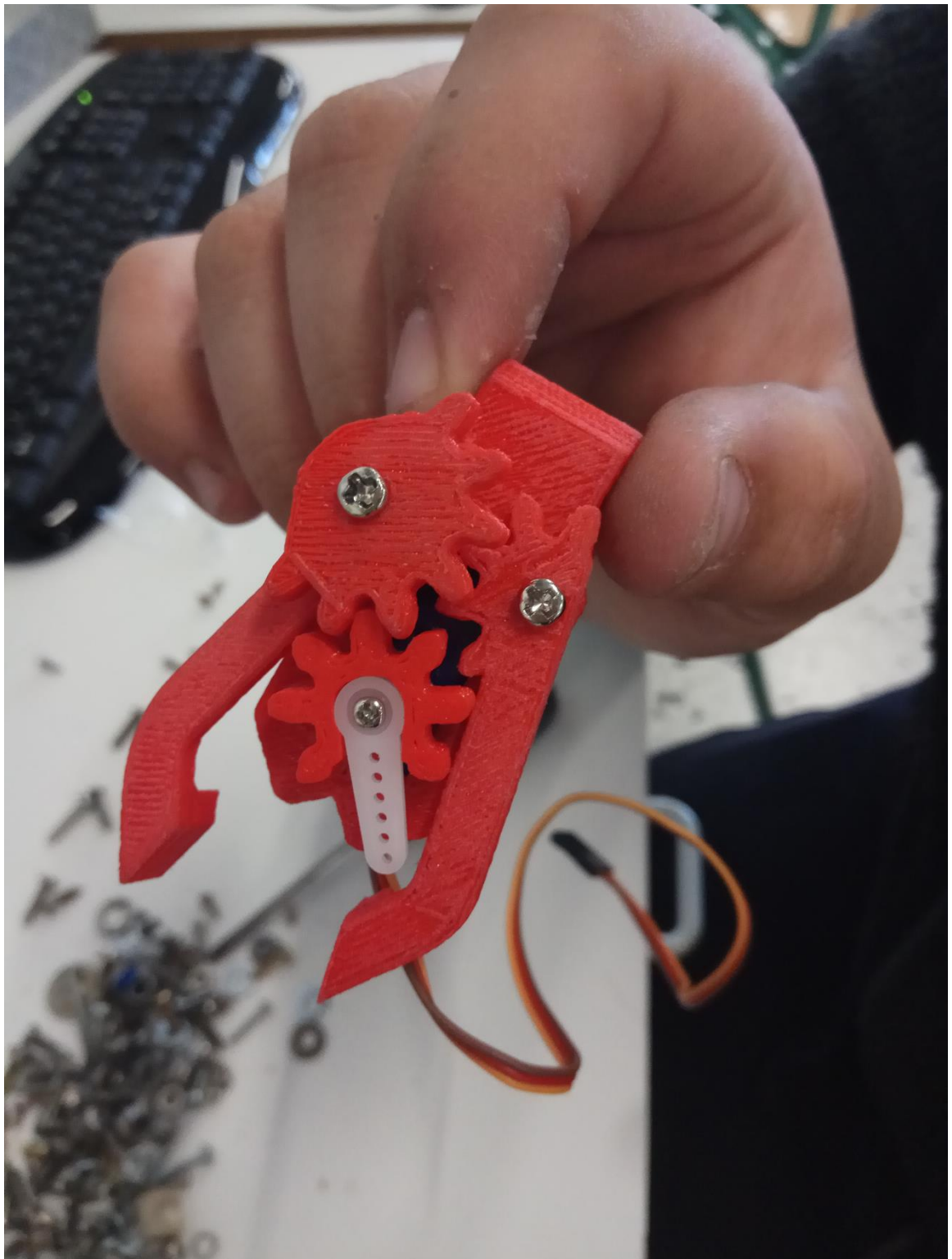
Το όνομα του βραχίονα είναι EEZYbotARM.

Αναλυτικές οδηγίες για τη συναρμολόγηση, μπορείτε να βρείτε στη διεύθυνση: <https://www.instructables.com/id/EEZYbotARM/>

Εκτυπώσαμε στον 3D εκτυπωτή μας, τα διάφορα μέρη του βραχίονα και τον συναρμολογήσαμε όπως φαίνεται και στις παρακάτω εικόνες:









Τα βήματα της εκτύπωσης- συναρμολόγησης, μπορείτε να τα δείτε και στο παρακάτω βίντεο: <https://www.youtube.com/watch?v=rYqWAz2NFqQ&feature=youtu.be>

2^ο βήμα: Προγραμματισμός

Σκοπός μας είναι να ελέγχουμε τον βραχίονα μέσω Bluetooth, με την κατάλληλη android εφαρμογή που θα υλοποιήσουμε. Παρακάτω Δίνεται ο κώδικας που γράψαμε με την κατάλληλη επεξήγηση:

Κώδικας:

```
#include <Servo.h> //Συμπερίληψη βιβλιοθήκης
```

```
Servo gripper, forth_back, up_down,Base; //Ονομαζουμε τους 4 servo
```

```
int realservo; // Μεταβλητή για το δεδομένο που θα στέλνεται από το κινητό
```

```
// Ακέραιες μεταβλητές για τις γωνίες στροφής του κάθε servo και αρχικές τιμές.
```

```
int pos1 = 0; //for gripper
```

```
int pos2 = 60; //for right
```

```
int pos3 = 160; //for left
```

```
int pos4=90; //for base
```

```
void setup()
```

```
{ // bluetooth.begin(9600);
```

```
  Serial.begin(9600);
```

```
  gripper.attach(9);
```

```
  forth_back.attach(10);
```

```
  up_down.attach(11);
```

```
  Base.attach(5);
```

//gripper είναι η δαγκάνα, forth_back είναι ο δεξιός servo που πραγματοποιεί την κίνηση μπροστά-πίσω, up_down είναι ο αριστερός που πραγματοποιεί την κίνηση πάνω-κάτω και base είναι η βάση που περιστρέφει σε οριζόντιο επίπεδο τον βραχίονα.

```
  gripper.write(pos1);
```

```
  forth_back.write(pos2); //right servo
```

```
  up_down.write(pos3); //left servo
```

```
Base.write(pos4);  
delay(100);  
}
```

void loop()

```
{  
  
  //Read from bluetooth and write to usb serial  
  
  if(Serial.available() > 0) {  
    realservo = Serial.read();  
    if (realservo >=0 && realservo <=60){  
      int pos1 = realservo;  
      pos1 = map(pos1,0,60,0,100);  
      gripper.write(pos1);  
  
      //Serial.println("gripper ON"); προαιρετικά αν θέλουμε να βλέπουμε ποιος servo  
      είναι ενεργός  
      //Serial.println(pos1);  
      delay(15);  }  
  
      if (realservo >=60 && realservo <=140){  
        int pos2 = realservo;  
        pos2 = map(pos2,60,140,20,160);  
        forth_back.write(pos2);  
        //Serial.println(" Rightservo On");  
        //Serial.println(pos2);  
        delay(15);    }  
  
        if (realservo >=140 && realservo <=180){  
          int pos3 = realservo;  
          pos3 = map(pos3, 140,180,100,180);
```

```

up_down.write(pos3);

//Serial.println("Leftservo On");

//Serial.println(pos3);

delay(15);    }

if (realservo >=180 && realservo <=250){

  int pos4 = realservo;

  pos4 = map(pos4, 180, 250,0,180);

  Base.write(pos4);

  //Serial.println("Base On");

  //Serial.println(pos4);

  delay(15);    }  }

```

Σημείωση: Μετά από δοκιμές καταλήξαμε στη περιοχή γωνιών στροφής του κάθε servo.

Επειδή το δεδομένο(realservo) που μας στέλνει το κινητό θα είναι 1 Byte δηλαδή από 0-255, κάνουμε καταμερισμό των τιμών σε τέσσερα διαστήματα , ένα για κάθε servo χρησιμοποιώντας την εντολή **if** .

Κατόπιν χρησιμοποιούμε την εντολή **map** για να προσαρμόσουμε το κάθε διάστημα τιμών στην περιοχή τιμών που απαιτούνται για κάθε servo.

3ο βήμα: Android εφαρμογή

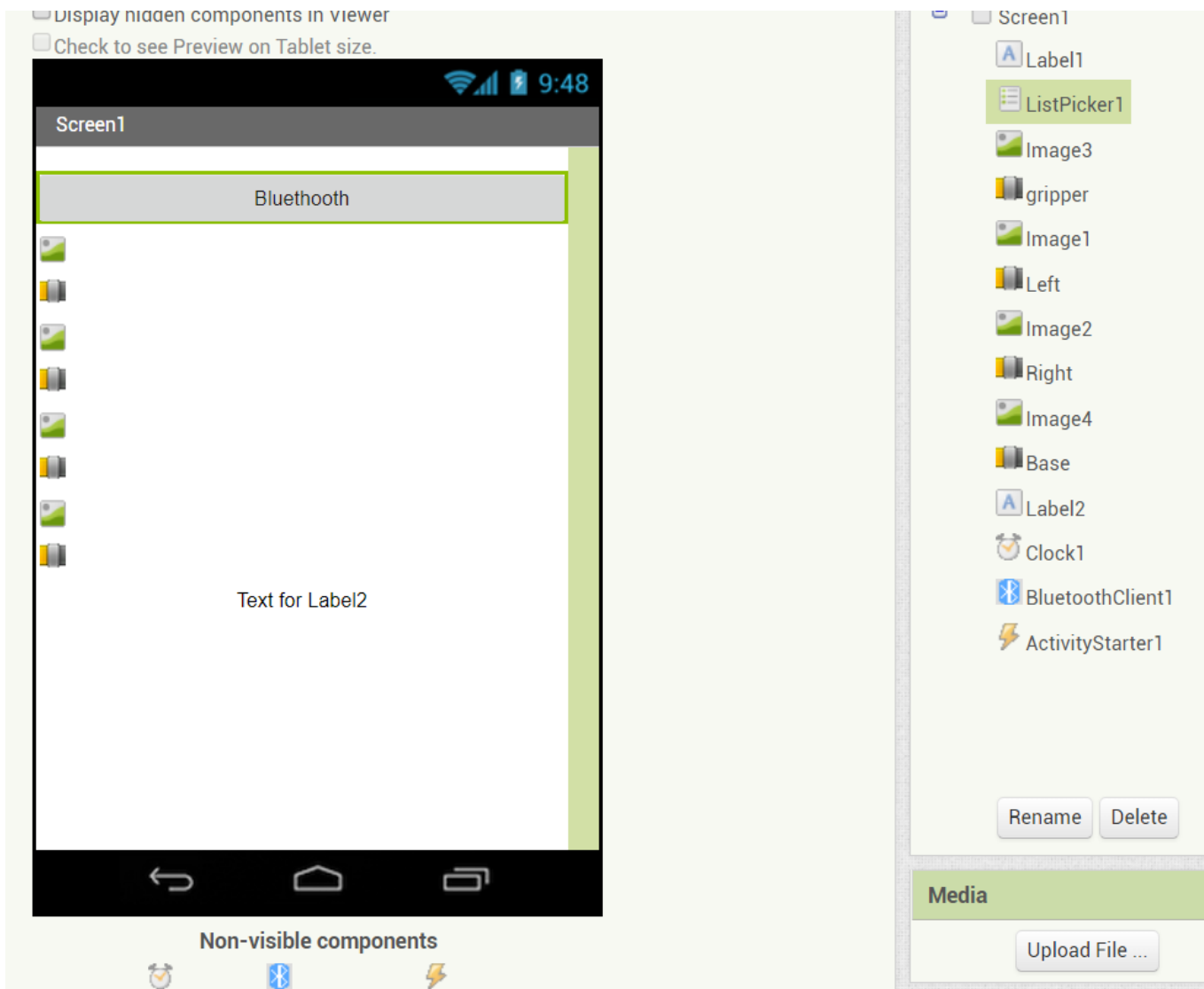
Χρησιμοποιούμε 4 sliders (μπάρες) για να μεταβάλουμε τις γωνίες των 4 servo. Σχεδιαστικά τροποποιούμε την περιοχή τιμών κύλισης της κάθε μπάρας, σύμφωνα με τον κώδικα.

Στο προγραμματιστικό μέρος της εφαρμογής (Blocks) πέρα από τις αρχικές ρυθμίσεις για την επιλογή και σύνδεση με Bluetooth, πρέπει να δημιουργήσουμε άλλα 4 blocks για τα 4 sliders. Όταν αλλάζει η θέση του δρομέα σε κάθε slider, να στέλνεται ένας αριθμός της περιοχής τιμών κύλισης της κάθε μπάρας.

Αναλυτικά βήματα, αντίστοιχης εργασίας μπορείτε να δείτε στο διαδίκτυο:

<https://www.hackster.io/slantconcepts/control-arduino-robot-arm-with-android-app-1c0d96>

Το σχεδιαστικό μέρος (Designer) της εφαρμογής:



Το προγραμματιστικό μέρος της εφαρμογής(Blocks)

```
when Screen1.Initialize
do
  set ActivityStarter1.Action to "android.bluetooth.adapter.action.REQUEST_ENABLE"
  call ActivityStarter1.StartActivity
```

```
when ListPicker1.BeforePicking
do
  set ListPicker1.Elements to BluetoothClient1.AddressesAndNames
```

```
when ListPicker1.AfterPicking
do
  if call BluetoothClient1.Connected
    address ListPicker1.Selection
  then
    set ListPicker1.Elements to BluetoothClient1.AddressesAndNames
```

```
when Clock1.Timer
do
  if BluetoothClient1.IsConnected
  then
    set Label1.Text to "Connected"
    set Label1.TextColor to green
  else
    set Label1.Text to "Not connected"
    set Label1.TextColor to red
```

```
when gripper.PositionChanged
  thumbPosition
do
  set Label2.Text to "Gripper ON"
  call BluetoothClient1.Send1ByteNumber
    number round gripper.ThumbPosition
```

```
when Left.PositionChanged
  thumbPosition
do
  set Label2.Text to "Left ON"
  call BluetoothClient1.Send1ByteNumber
    number round Left.ThumbPosition
```

```
when Right.PositionChanged
  thumbPosition
do
  set Label2.Text to "Right ON"
  call BluetoothClient1.Send1ByteNumber
    number round Right.ThumbPosition
```

```
when Base.PositionChanged
  thumbPosition
do
  set Label2.Text to "Base ON"
  call BluetoothClient1.Send1ByteNumber
    number round Base.ThumbPosition
```