**CNC KALEM YAZICI YAPIMI VE KODLARI**

**CNC Makinenin Çalışması:**

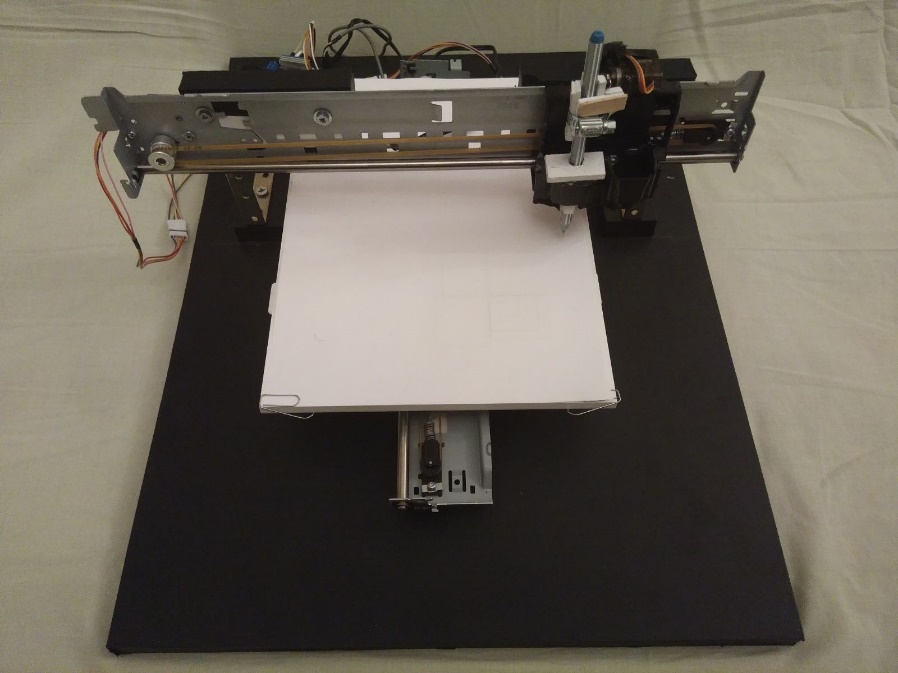
Bir **CNC çizim makinesinin** çalışması için 3 eksen gereklidir (x ekseni, y ekseni ve z ekseni. X ekseni ve y ekseni, düz bir kağıda 2D görüntü oluşturmak için birlikte çalışır. Eksen, düz yüzey üzerindeki herhangi bir noktanın belirli bir x ve y değeri ile tanımlanacağı şekilde birbirlerine 90 derece yerleştirilir, z ekseni kalemin aşağı yukarı hareketini sağlanır.

Çizilecek olan görüntüye bağlı olarak, bilgisayar uygun koordinatları üretecek ve bunları USB portu üzerinden mikrodenetleyiciye gönderecektir. Mikrodenetleyici bu koordinatları yorumlar ve ardından görüntüyü oluşturmak için motorların pozisyonlarını kontrol eder. Burada bu CNC makinesini inşa etmek için Mikrodenetleyici olarak Arduino kullandık.

**Donanım Gereksinimi:**

1. Alüminyum levha veya ahşap levha
2. Dekota levha
3. Eski HP / Epson yazıcının kartuşun sağa sola hareketini sağlayan parçası. Eski bilgisayar DVD sürücülerini kullanabilirsiniz
4. İki adet step motor (28 BYJ-48 Redüktörlü Step Motor kulanıldı.)
5. Step motorun ucuna takılması için dişli kasnak
6. Cıvata ve somun
7. Perspex camı
8. Arduino UNO
9. L293D motor sürücü kalkanı
10. Mini servo motor
11. Bir kalem

**Donanımın bitmiş halinin görüntüsü**

****

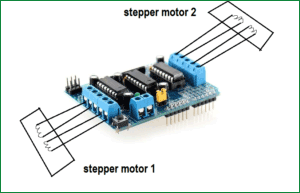
**Yazılımları:**

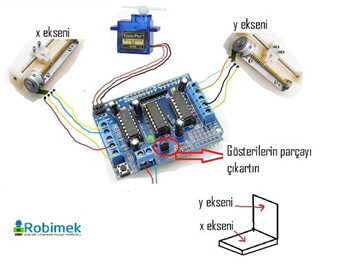
Bu makinenin verimli çalışması için, aşağıdaki yazılımlar kullanılır. Çeşitli web sitelerinden indirilebilir.

1. Arduino IDE sürüm 1.6.6 veya daha sonraki sürümleri [buradan](http://www.arduino.cc/en/Main/Software)
2. IDE sürüm 3.1.1 veya sonraki sürümü buradan [işleniyor](http://www.processing.org/download/?processing)
3. Inkscape sürümü 0.48.5. [Buradan](https://inkscape.org/en/download/) indirin .
4. Grbl denetleyici (isteğe bağlı)

**CNC Makine Kablolama ve Devresi:**

**L293D motor sürücü kalkanını** Arduino UNO kartına takın . Bu kalkan, iki step motorunu aynı anda ve iki servo motoru çalıştırabilir. İki step motorunu aşağıda gösterildiği gibi bağlayın. Motorlar bipoplar tip olduğundan, toprak bağlantısı bağlantısız bırakılmalıdır.

[](https://circuitdigest.com/fullimage?i=inlineimages/wiring-motor-driver-shield-for-arduino-CNC-machine.png)



**Ayrıca mini servo motoru** servo1'e takın. Motor sürücü korumasının güç bağlantı noktasına bir 7.5V - 9V güç kaynağı bağlayın. Makine şimdi test için hazır.

**Arduino CNC Makine Kodu ve Testi:**

Öncelikle **step motorları test etmeli** ve doğru şekilde bağlanıp bağlanmadıklarına bakmalıyız.

L293D motor sürücü kalkanını kullandığımızdan, [**AFmotor Kütüphanesini buradan indirmemiz gerekiyor**](https://github.com/adafruit/Adafruit-Motor-Shield-library/zipball/master) . Ardından Arduino IDE kütüphane klasörünüze ekleyin. *AFMotor'a* yeniden adlandırdığınızdan emin olun . Arduino IDE açıksa ve tekrar açın ve *dosyayı açın -> örnekler -> Adafruit Motor Kalkanı Kütüphanesi -> step* . Doğru portu seçip aletlere bindiğinizden emin olun ve ardından kodu Arduino kartına yükleyin. Step motor 1'de bazı hareketler izlenmelidir.

Motor 2'yi test etmek için, motor hattını aşağıdaki satırda 2'den 1'e değiştirin ve ardından kodu tekrar yükleyin.

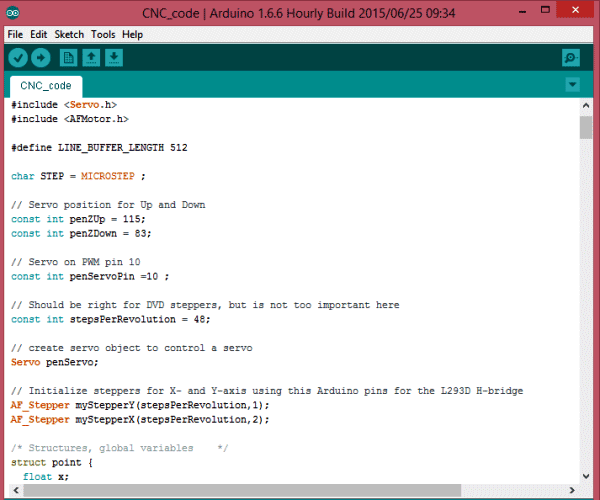
#include <AFMotor.h>

// Devir başına 48 basamaklı bir step motor bağlayın (7.5 derece)

// motor bağlantı noktası # 2'ye (M3 ve M4)

AF\_Stepper motoru (48, 2);

**CNC Makinesi için Arduino Kodu:**



Adım motorları uygun şekilde yanıt verdikten sonra, aşağıdaki kod bölümündeki **CNC makinesi için Arduino kodunu** kopyalayın ve Arduino kartına yükleyin. [burada da kaynak kod indirmek.](https://circuitdigest.com/sites/default/files/Arduino-code-for-CNC-machine.ino)

**CNC Makinesi için G Kodu:**

G - CODE, bilgisayarlı makinelere (CNC) bir şey yapmasını istediğimiz dildir. Temel olarak X, Y ve Z koordinatlarını içeren bir dosyadır.

Örneğin:

G17 G20 G90 G94 G54

G0 Z0.25X-0.5 Y0.

Z0.1

G01 Z0. F5.

G02 X0. Y0.5 I0.5 J0. F2.5

X0.5 Y0. I0. J-0.5

X0. Y-0.5 I-0.5 J0.

X-0.5 Y0. I0. J0.5

G01 Z0.1 F5.

G00 X0. Y0. Z0.25

Sadece basit bir kare için bir G Kodunun yazılması gerçekten zor olabilir ama neyse ki bir G Kodu oluşturmamıza yardımcı olabilecek bir yazılımımız var. Bu yazılıma " **Inkscape** " denir [**, buradan**](https://inkscape.org/en/download/) **indirebilirsiniz.**

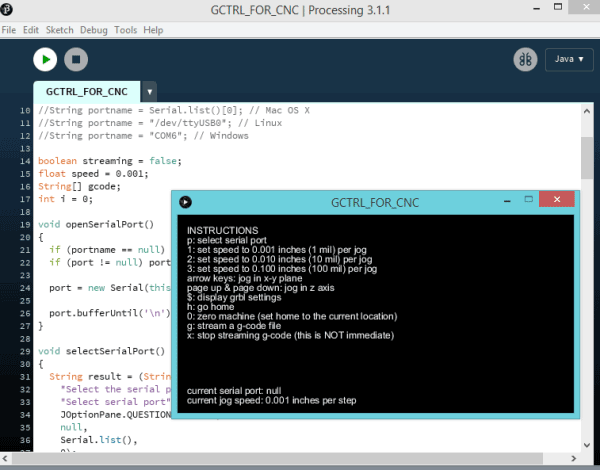
Bir sonraki bölümde açıkladığımız **Inkscape'i** kullanarak kendi G Kodunuzu oluşturabilir, ancak internetteki hazır G Kodlarını kullanabilirsiniz.

Inkscape'i kullanarak G Kodlarının nasıl oluşturulacağını göstermeden önce, bu G Kodlarının Arduino‘ya nasıl gönderileceğini tartışalım. G-Kodlarını Arduino‘ya göndermemize olanak tanıyan yazılım [İşleme](https://processing.org/) denir.

**G Kodu'nu yüklemek için IDE'yi işleyin:**

Bu platform G-Kodlarını Arduino panosuna göndermemize yardımcı olacak. Bunu yapmak için, GCTRL.PDE dosyasını indirmeniz gerekecektir.

[Buradan GCTRL.pde dosyasını indirin ve IDE'yi](https://circuitdigest.com/sites/default/files/GCTRL.pde) kullanarak açın



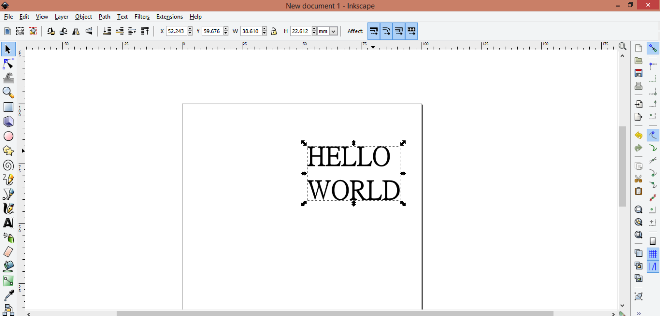
İşleme IDE’si içinde açtıktan sonra, çalıştır’ı tıklayın. Tüm talimatların bulunduğu bir pencere açılır. Klavyedeki **p** tuşuna basın . Sistem bir bağlantı noktası seçmenizi isteyecektir. Arduino kartınızın bağlı olduğu bağlantı noktasını seçin. Benim durumum **6 numaralı limanı.**

Şimdi **g** tuşuna basın ve G-CODE'nizi kaydettiğiniz klasöre göz atın. Doğru G-KOD'u seçin ve enter tuşuna basın. Her şey doğru bir şekilde bağlanmışsa, kağıt üzerinde çizim yapmaya başladığınız bir cihaz görmelisiniz.

İşlemi sonlandırmak isterseniz, **x** düğmesine basmanız **yeterlidir** ve cihaz ne yapıyorsa onu durduracaktır.

**Kendi G Kodunuzu Nasıl Oluşturulur?**

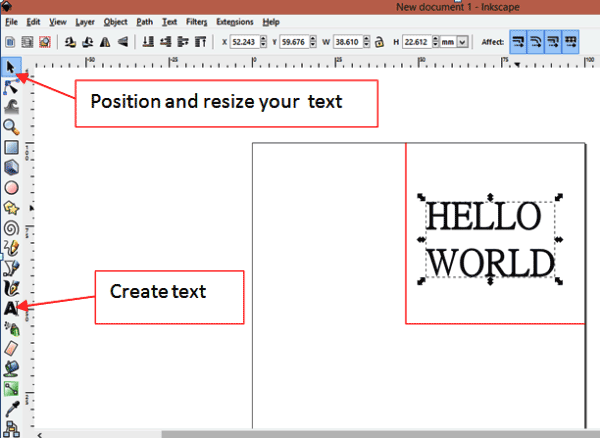
Biz belirtti **Inkscape** bizim G-KODLARI oluşturmak için kullanacağı bir yazılımdır. Bu örnekte, aşağıda gösterildiği gibi basit bir metin ( **HELLO WORLD)** oluşturacağız.



***Not***: Inkscape’in *dosyaları G-CODE olarak kaydetme yolu yoktur* . Bu nedenle, dışa aktarma görüntülerinin G-CODE dosyalarına izin veren**bir Eklenti yüklemeniz** gerekir. Bu[MakerBot Unicorn eklentisini](http://github.com/martymcguire/inkscape-unicorn) kurulum notlarıyla[buradan](http://github.com/martymcguire/inkscape-unicorn) indirin.

Yükleme başarılı olursa, Inkscape'i açın, Dosya menüsüne gidin ve "Belge Özellikleri" ni tıklayın. İlk değişiklik boyutları px'den mm'ye. Ayrıca genişliği ve yüksekliği 90 mm'ye indirin. Şimdi bu pencereyi kapat. Çizim alanı olarak bir kare belirir. Bu, metnimizi yazmak için kullanacağımız alan.

Şimdi soldaki çubukta, metin nesnesi oluştur ve düzenle sekmesine tıklayın. " **HELLO WORLD** " metnini yazın ve aşağıda gösterildiği gibi karenin sağ üst köşesine yerleştirin.

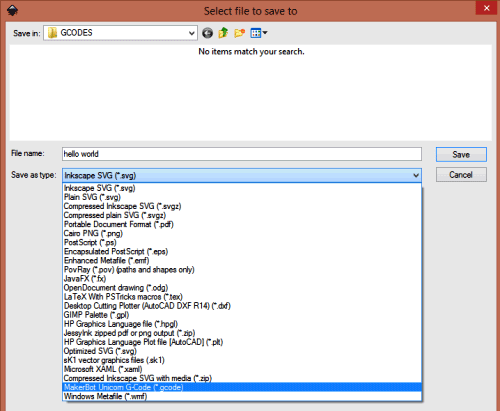


Metne tıklayın ve tercih ettiğiniz yazı tipi stilini seçin. Uygula ve kapat'ı tıklayın.

Şimdi yola tıklayın ve " **yola gönder** " seçeneğini seçin.

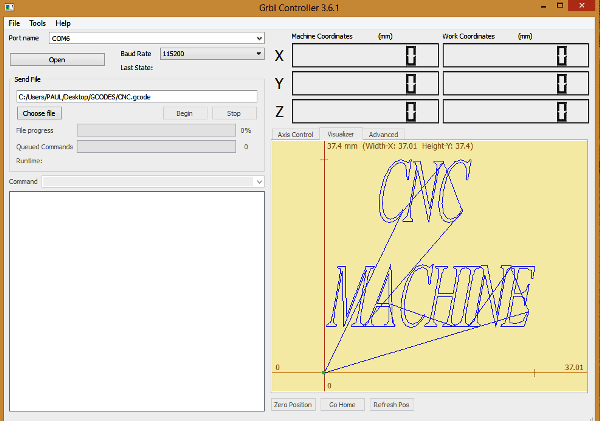
Metniniz artık G-CODE olarak kaydedilmeye hazır. *Dosyaya* tıklayın *-> farklı kaydet* ve dosya adını "merhaba dünya" olarak yazın.

Aşağıdaki resimde gösterildiği gibi dosya tipini *"MakerBot Unicon G-Code"* olarak değiştirin. Bu sadece Eklenti yüklemesi başarılı olursa görünecektir. Son olarak, kaydet'e tıklayın ve pop-up pencerede tamam'a tıklayın.



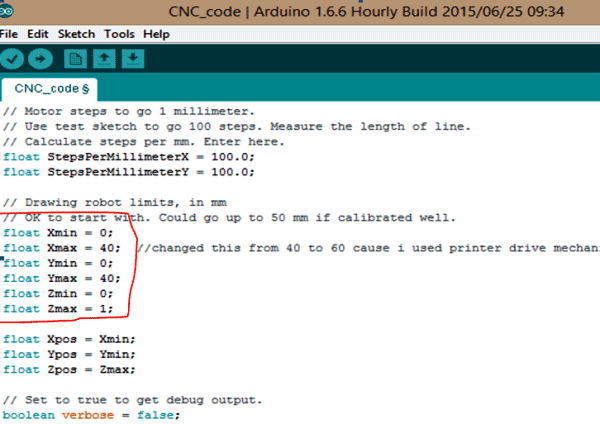
Bir G Kodu oluşturdunuz ve önceki prosedürleri kullanarak çizilebilir.

**GRBL Denetleyici:**



Inkscape kullanarak bir G Kodu oluşturmayı başardıktan sonra, çizim sınırlarının içinde olduğundan emin olmak için G-Kodunu görüntülemek gerekebilir.

**Çizim sınırları çok çaba isteyen CNC KOD ayarlanır** aşağıda gösterilen çizgilerle:



Yukarıdaki GRBL denetleyicisindeki görüntü, yukarıdaki **CNC Arduino kodunda** gösterilen sınırın ötesine geçmemelidir. Bu sınırın ötesine geçerse, örneğin x ekseninin negatif tarafına doğru, negatif taraftaki kısım çizilmeyecektir.

Bu örnekte x ve y değerleri 0 mm ila 40 mm arasındadır.

Daha geniş bir alana çizilebilen yazıcı parçaları kullandığım için, max değerlerini 40mm'den 60mm'ye değiştiriyorum.

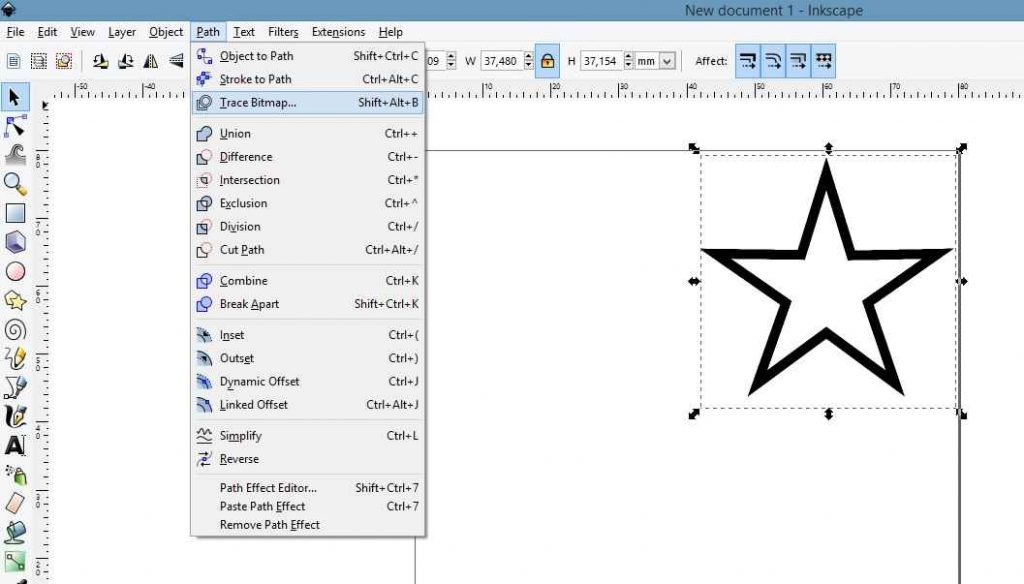
Inkscape kullanarak bir G Kodu oluşturduğunuzda, ilk önce bu G-Kodunu GRBL programında bu sınırlar içinde olup olmadığını görmek için açabilirsiniz. İçerisinde değilse, Inkscape'deki resminizi sınırlarınızda olana kadar yeniden boyutlandırmanız gerekir.

**İnkscape ile Resmin Gcode Çıktısını Alma :**

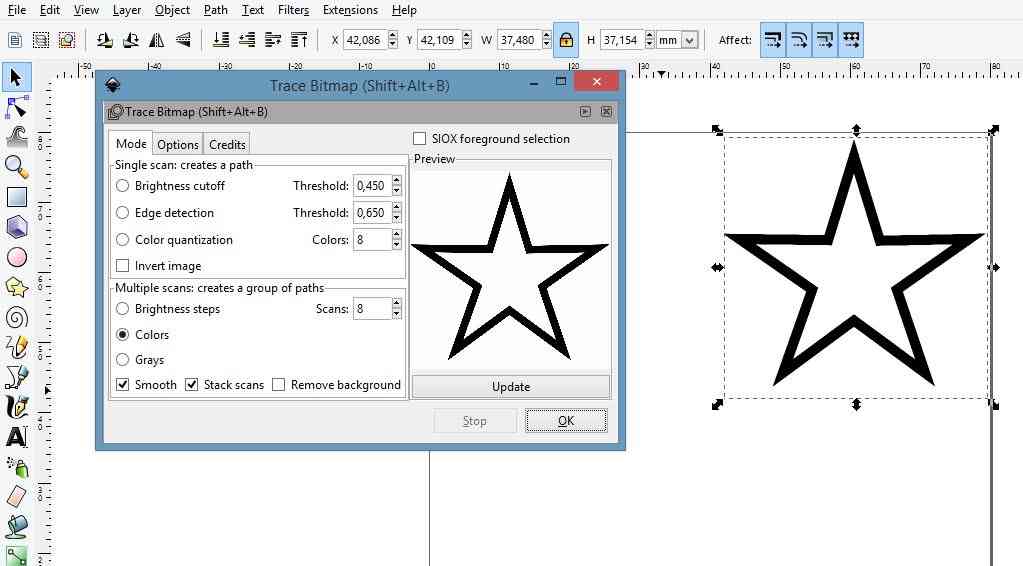
Yukarıda anlattığım gibi sayfa boyutlarını ve metrik sistemi ayarlayın. Çizim alanı da yine sayfanın dörtte biri olan sağ üst alanı olacak. Bu kısımları ayarladıktan sonra aşağıdaki adımları takip ediniz.

**1.adım: File** menüsünden **import** a tıklayın ve çizdirmek istediğiniz resmi seçin.

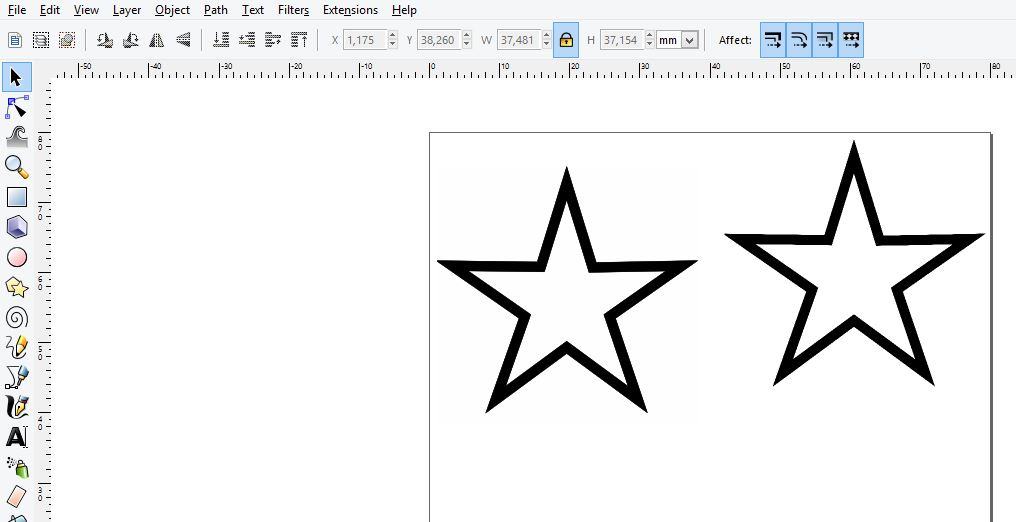
**2.adım:**Resmi çizim alanına sığacak şekilde boyutunu ayarlayın ve çizim alanına konumlandırın. Daha sonra resmi seçin ve aşağıdaki gibi **path** menüsünden **Trace Bitmap** e tıklayın.

[](https://www.robimek.com/wp-content/uploads/Inkscape_ayar_6.jpg)

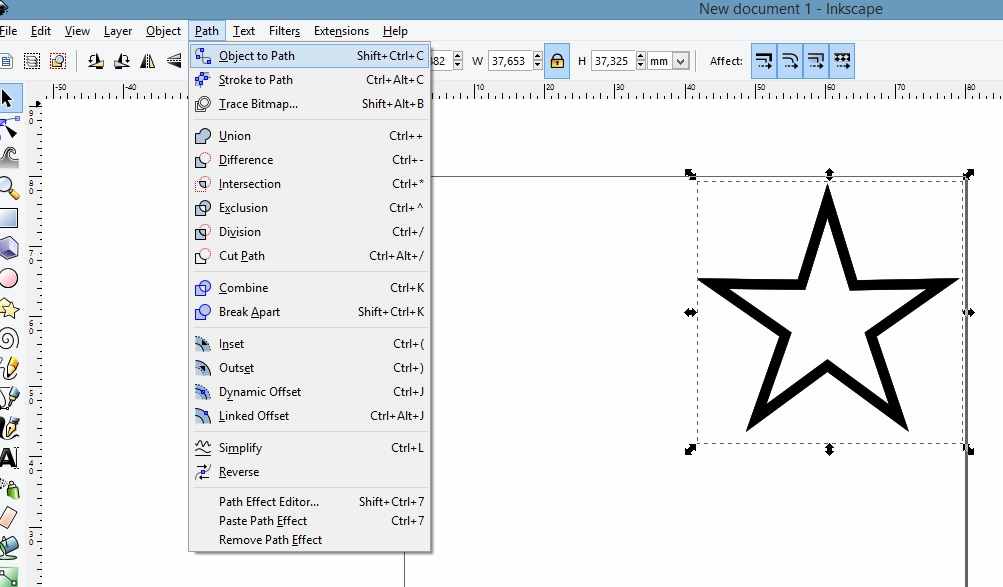
**3.adım:**Karşımıza bir pencere çıkıyor. Aşağıdaki gibi Colors seçeneğini seçin ve Update butonuna tıklayın. Ardından ok butonuna tıklayın işlem bitene kadar bekleyin. İşlem bittiğinde pencereyi kapatın.

[](https://www.robimek.com/wp-content/uploads/Inkscape_ayar_7.jpg)

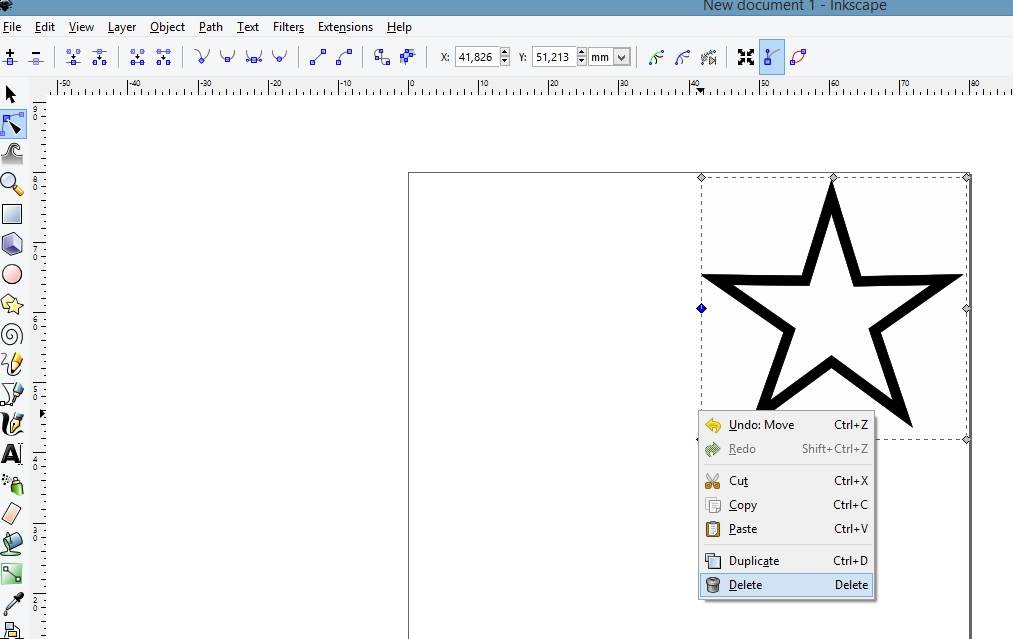
**4.adım:**Resmi farenin sol tuşuna basılı tutarak yana kaydırın.Aşağıdaki gibi iki tane resim oluştuğunu göreceksiniz. Bunlardan yana kaydırdığınız resim vektörel forma dönüşmüş olan resimdir. Diğer resmi silin ve yana kaydırmış olduğunuz resmi çizim alanına yerleştirin.

[](https://www.robimek.com/wp-content/uploads/Inkscape_ayar_8.jpg)

**5.adım:**Resmi seçin ve aşağıdaki gibi **path** menüsünden **opject to path** a tıklayın.

[](https://www.robimek.com/wp-content/uploads/Inkscape_ayar_9.jpg)

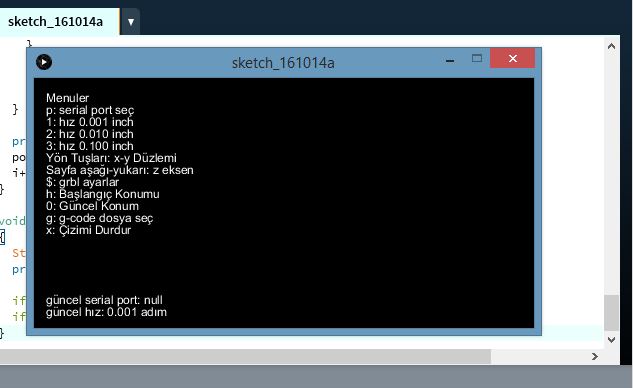
**6.adım:**Resmi vektörel yaptık fakat dış çerçeve kısmındaki kalıntıları silmemiz gerekiyor. Bunun için sol taraftaki araçlardan fare okunun altında olan **Edit Paths by nodes**aracını seçin ve aşağıdaki gibi resim üzerine tıklayın. Kesikli çizgi üzerine gelin ve farenin sağ tuşuna basarak delete yi seçin.

[](https://www.robimek.com/wp-content/uploads/Inkscape_ayar_10.jpg)

**7.adım:**File menüsünden farklı kaydet e tıklayın ve gcode u seçerek kaydedin.

**Gcode’ u Cnc’ ye Göndererek Çizim Yaptırma İşlemleri:**

Gcode çıktısını aldığımız çizimi cnc ye nasıl göndereceğimize bakalım. Öncelikle bütün bağlantıları tekrar kontrol ediniz ve doğru olduğundan emin olunuz. Daha sonra arduinoyu bilgisayara bağlayın. Processing programını açın ve processing kodunu ekrana aktararak programı çalıştırın. Karşımıza menülerin olduğu aşağıdaki şekildeki gibi bir ekran gelecektir.

[](https://www.robimek.com/wp-content/uploads/cnc-kontrol-panel.jpg)

İlk önce p tuşuna basın ve arduinonun bağlı olduğu serial portu seçin. Ardından G tuşuna basarak çizdirmek istediğiniz gcode dosyasını seçin. Dosyayı seçtiğinizde çizim başlayacaktır. Ayrıca cnc çizim yaparken X tuşuna basarsanız çizimi durdurabiliriz. ve H tuşuna basarak da kalemi başlangıç noktasına alırsınız. Diğer bir özellik ise çizim hızını 1, 2 veya 3 tuşlarına basarak hangi hızda olmasını istiyorsunuz ayarlayabilirsiniz.

**Arduino Kod:**

/\*   
 Gctrl.pde kullanarak bu Sketch'e GCODE gönder <https://github.com/damellis/gctrl>  
 InDesign için MakerBot Unicorn eklentisi ile SVG'yi GCODE'ye dönüştürme Burada bulabilirsiniz <https://github.com/martymcguire/inkscape-unicorn>  
 Arduino kodu Bu Mini CNC Plotter dayalı: <https://github.com/adidax/mini_cnc_plotter_firmware>  
\*/

#include <Servo.h>  
#include <AFMotor.h>

#define LINE\_BUFFER\_LENGTH 512

char STEP = MICROSTEP ;

// Yukarı ve Aşağı   
const int penZUp = 115;  
const int penZDown = 83; //kalemin ucunun kağıt ile mesafesine göre ayarlayın

// Servo on PWM pin 10  
const int penServoPin =10 ;

// Should be right for DVD steppers, but is not too important here  
const int stepsPerRevolution = 48;

// create servo object to control a servo   
Servo penServo;

// Initialize steppers for X- and Y-axis using this Arduino pins for the L293D H-bridge  
AF\_Stepper myStepperY(stepsPerRevolution,1);              
AF\_Stepper myStepperX(stepsPerRevolution,2);

/\* Structures, global variables    \*/  
struct point {   
  float x;   
  float y;   
  float z;   
};

// Current position of plothead  
struct point actuatorPos;

//  Drawing settings, should be OK  
float StepInc = 1;  
int StepDelay = 0;  
int LineDelay =0;  
int penDelay = 50;

// 1 milimetre gitmek için motor adımları.   
// 100 adım atmak için test çizimini kullanın. Çizginin uzunluğunu ölçün.   
// mm başına adımları hesaplayın. Buraya girin   
float StepsPerMillimeterX = 800.0; // 28 BYJ-48 Redüktörlü Step Motor için olması gereken değer   
float StepsPerMillimeterY = 800.0; // ikisi içinde 800 olmalı

// çizim yapacağınız eksenlere göre xmax ve ymax değerleri ayarlanmalı  
// aşağıdaki değerler DVD sürücü ile yapılan mini cnc yazıcı için  
float Xmin = 0;  
float Xmax = 40;  
float Ymin = 0;  
float Ymax = 40;  
float Zmin = 0;  
float Zmax = 1;

float Xpos = Xmin;  
float Ypos = Ymin;  
float Zpos = Zmax;

// Hata ayıklama çıktısını almak için true olarak ayarlayın.   
boolean verbose = false;

//  Needs to interpret   
//  G1 for moving  
//  G4 P300 (wait 150ms)  
//  M300 S30 (pen down)  
//  M300 S50 (pen up)  
//  Discard anything with a (  
//  Discard any other command!

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
 \* void setup() - Initialisations  
 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
void setup() {  
  //  Setup  
    
  Serial.begin( 9600 );  
    
  penServo.attach(penServoPin);  
  penServo.write(penZUp);  
  delay(100);

  // Gerekiyorsa azaltın 28 BYJ-48 Redüktörlü Step Motor için myStepperX.setSpeed(10); değeri kullandım.

  myStepperX.setSpeed(600);  
  myStepperY.setSpeed(600);    
    
  //  Set & move to initial default position  
  // TBD

  //  Notifications!!!  
  Serial.println("Mini CNC Plotter alive and kicking!");  
  Serial.print("X range is from ");   
  Serial.print(Xmin);   
  Serial.print(" to ");   
  Serial.print(Xmax);   
  Serial.println(" mm.");   
  Serial.print("Y range is from ");   
  Serial.print(Ymin);   
  Serial.print(" to ");   
  Serial.print(Ymax);   
  Serial.println(" mm.");   
}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
 \* void loop() - Main loop  
 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
void loop()   
{  
    
  delay(100);  
  char line[ LINE\_BUFFER\_LENGTH ];  
  char c;  
  int lineIndex;  
  bool lineIsComment, lineSemiColon;

  lineIndex = 0;  
  lineSemiColon = false;  
  lineIsComment = false;

  while (1) {

    // Serial reception - Mostly from Grbl, added semicolon support  
    while ( Serial.available()>0 ) {  
      c = Serial.read();  
      if (( c == '\n') || (c == '\r') ) {             // End of line reached  
        if ( lineIndex > 0 ) {                        // Line is complete. Then execute!  
          line[ lineIndex ] = '\0';                   // Terminate string  
          if (verbose) {   
            Serial.print( "Received : ");   
            Serial.println( line );   
          }  
          processIncomingLine( line, lineIndex );  
          lineIndex = 0;  
        }   
        else {   
          // Empty or comment line. Skip block.  
        }  
        lineIsComment = false;  
        lineSemiColon = false;  
        Serial.println("ok");      
      }   
      else {  
        if ( (lineIsComment) || (lineSemiColon) ) {   // Throw away all comment characters  
          if ( c == ')' )  lineIsComment = false;     // End of comment. Resume line.  
        }   
        else {  
          if ( c <= ' ' ) {                           // Throw away whitepace and control characters  
          }   
          else if ( c == '/' ) {                    // Block delete not supported. Ignore character.  
          }   
          else if ( c == '(' ) {                    // Enable comments flag and ignore all characters until ')' or EOL.  
            lineIsComment = true;  
          }   
          else if ( c == ';' ) {  
            lineSemiColon = true;  
          }   
          else if ( lineIndex >= LINE\_BUFFER\_LENGTH-1 ) {  
            Serial.println( "ERROR - lineBuffer overflow" );  
            lineIsComment = false;  
            lineSemiColon = false;  
          }   
          else if ( c >= 'a' && c <= 'z' ) {        // Upcase lowercase  
            line[ lineIndex++ ] = c-'a'+'A';  
          }   
          else {  
            line[ lineIndex++ ] = c;  
          }  
        }  
      }  
    }  
  }  
}

void processIncomingLine( char\* line, int charNB ) {  
  int currentIndex = 0;  
  char buffer[ 64 ];                                 // Hope that 64 is enough for 1 parameter  
  struct point newPos;

  newPos.x = 0.0;  
  newPos.y = 0.0;

  //  Needs to interpret   
  //  G1 for moving  
  //  G4 P300 (wait 150ms)  
  //  G1 X60 Y30  
  //  G1 X30 Y50  
  //  M300 S30 (pen down)  
  //  M300 S50 (pen up)  
  //  Discard anything with a (  
  //  Discard any other command!

  while( currentIndex < charNB ) {  
    switch ( line[ currentIndex++ ] ) {              // Select command, if any  
    case 'U':  
      penUp();   
      break;  
    case 'D':  
      penDown();   
      break;  
    case 'G':  
      buffer[0] = line[ currentIndex++ ];          // /!\ Dirty - Only works with 2 digit commands  
      //      buffer[1] = line[ currentIndex++ ];  
      //      buffer[2] = '\0';  
      buffer[1] = '\0';

      switch ( atoi( buffer ) ){                   // Select G command  
      case 0:                                   // G00 & G01 - Movement or fast movement. Same here  
      case 1:  
        // /!\ Dirty - Suppose that X is before Y  
        char\* indexX = strchr( line+currentIndex, 'X' );  // Get X/Y position in the string (if any)  
        char\* indexY = strchr( line+currentIndex, 'Y' );  
        if ( indexY <= 0 ) {  
          newPos.x = atof( indexX + 1);   
          newPos.y = actuatorPos.y;  
        }   
        else if ( indexX <= 0 ) {  
          newPos.y = atof( indexY + 1);  
          newPos.x = actuatorPos.x;  
        }   
        else {  
          newPos.y = atof( indexY + 1);  
          indexY = '\0';  
          newPos.x = atof( indexX + 1);  
        }  
        drawLine(newPos.x, newPos.y );  
        //        Serial.println("ok");  
        actuatorPos.x = newPos.x;  
        actuatorPos.y = newPos.y;  
        break;  
      }  
      break;  
    case 'M':  
      buffer[0] = line[ currentIndex++ ];        // /!\ Dirty - Only works with 3 digit commands  
      buffer[1] = line[ currentIndex++ ];  
      buffer[2] = line[ currentIndex++ ];  
      buffer[3] = '\0';  
      switch ( atoi( buffer ) ){  
      case 300:  
        {  
          char\* indexS = strchr( line+currentIndex, 'S' );  
          float Spos = atof( indexS + 1);  
          //         Serial.println("ok");  
          if (Spos == 30) {   
            penDown();   
          }  
          if (Spos == 50) {   
            penUp();   
          }  
          break;  
        }  
      case 114:                                // M114 - Repport position  
        Serial.print( "Absolute position : X = " );  
        Serial.print( actuatorPos.x );  
        Serial.print( "  -  Y = " );  
        Serial.println( actuatorPos.y );  
        break;  
      default:  
        Serial.print( "Command not recognized : M");  
        Serial.println( buffer );  
      }  
    }  
  }  
}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
 \* Draw a line from (x0;y0) to (x1;y1).  
 \* int (x1;y1) : Starting coordinates  
 \* int (x2;y2) : Ending coordinates  
 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
void drawLine(float x1, float y1) {

  if (verbose)  
  {  
    Serial.print("fx1, fy1: ");  
    Serial.print(x1);  
    Serial.print(",");  
    Serial.print(y1);  
    Serial.println("");  
  }

  //  Bring instructions within limits  
  if (x1 >= Xmax) {   
    x1 = Xmax;   
  }  
  if (x1 <= Xmin) {   
    x1 = Xmin;   
  }  
  if (y1 >= Ymax) {   
    y1 = Ymax;   
  }  
  if (y1 <= Ymin) {   
    y1 = Ymin;   
  }

  if (verbose)  
  {  
    Serial.print("Xpos, Ypos: ");  
    Serial.print(Xpos);  
    Serial.print(",");  
    Serial.print(Ypos);  
    Serial.println("");  
  }

  if (verbose)  
  {  
    Serial.print("x1, y1: ");  
    Serial.print(x1);  
    Serial.print(",");  
    Serial.print(y1);  
    Serial.println("");  
  }

  //  Convert coordinates to steps  
  x1 = (int)(x1\*StepsPerMillimeterX);  
  y1 = (int)(y1\*StepsPerMillimeterY);  
  float x0 = Xpos;  
  float y0 = Ypos;

  //  Let's find out the change for the coordinates  
  long dx = abs(x1-x0);  
  long dy = abs(y1-y0);  
  int sx = x0<x1 ? StepInc : -StepInc;  
  int sy = y0<y1 ? StepInc : -StepInc;

  long i;  
  long over = 0;

  if (dx > dy) {  
    for (i=0; i<dx; ++i) {  
      myStepperX.onestep(sx,STEP);  
      over+=dy;  
      if (over>=dx) {  
        over-=dx;  
        myStepperY.onestep(sy,STEP);  
      }  
    delay(StepDelay);  
    }  
  }  
  else {  
    for (i=0; i<dy; ++i) {  
      myStepperY.onestep(sy,STEP);  
      over+=dx;  
      if (over>=dy) {  
        over-=dy;  
        myStepperX.onestep(sx,STEP);  
      }  
      delay(StepDelay);  
    }      
  }

  if (verbose)  
  {  
    Serial.print("dx, dy:");  
    Serial.print(dx);  
    Serial.print(",");  
    Serial.print(dy);  
    Serial.println("");  
  }

  if (verbose)  
  {  
    Serial.print("Going to (");  
    Serial.print(x0);  
    Serial.print(",");  
    Serial.print(y0);  
    Serial.println(")");  
  }

  //  Delay before any next lines are submitted  
  delay(LineDelay);  
  //  Update the positions  
  Xpos = x1;  
  Ypos = y1;  
}

//  Raises pen  
void penUp() {   
  penServo.write(penZUp);   
  delay(penDelay);   
  Zpos=Zmax;   
  digitalWrite(15, LOW);  
    digitalWrite(16, HIGH);  
  if (verbose) {   
    Serial.println("Pen up!");   
      
  }   
}  
//  Lowers pen  
void penDown() {   
  penServo.write(penZDown);   
  delay(penDelay);   
  Zpos=Zmin;   
  digitalWrite(15, HIGH);  
    digitalWrite(16, LOW);  
  if (verbose) {   
    Serial.println("Pen down.");   
  }   
}

**Processing Yazılımı:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145 | import java.awt.event.KeyEvent;  import javax.swing.JOptionPane;  import processing.serial.\*;    Serial port = null;    // select and modify the appropriate line for your operating system  // leave as null to use interactive port (press 'p' in the program)  String portname = null;  //String portname = Serial.list()[0]; // Mac OS X  //String portname = "/dev/ttyUSB0"; // Linux  //String portname = "COM6"; // Windows    boolean streaming = false;  float speed = 0.001;  String[] gcode;  int i = 0;    void openSerialPort()  {  if (portname == null) return;  if (port != null) port.stop();    port = new Serial(this, portname, 9600);    port.bufferUntil('\n');  }    void selectSerialPort()  {  String result = (String) JOptionPane.showInputDialog(frame,  "Select the serial port that corresponds to your Arduino board.",  "serial port seç",  JOptionPane.QUESTION\_MESSAGE,  null,  Serial.list(),  0);    if (result != null) {  portname = result;  openSerialPort();  }  }    void setup()  {  size(500, 250);  openSerialPort();  }    void draw()  {  background(0);  fill(255);  int y = 24, dy = 12;  text("Menuler", 12, y); y += dy;  text("p: serial port seç", 12, y); y += dy;  text("1: hız 0.001 inch", 12, y); y += dy;  text("2: hız 0.010 inch", 12, y); y += dy;  text("3: hız 0.100 inch", 12, y); y += dy;  text("Yön Tuşları: x-y Düzlemi", 12, y); y += dy;  text("Sayfa aşağı-yukarı: z eksen ", 12, y); y += dy;  text("$: grbl ayarlar", 12, y); y+= dy;  text("h: Başlangıç Konumu", 12, y); y += dy;  text("0: Güncel Konum", 12, y); y += dy;  text("g: g-code dosya seç ", 12, y); y += dy;  text("x: Çizimi Durdur", 12, y); y += dy;  y = height - dy;  text("güncel hız: " + speed + " adım", 12, y); y -= dy;  text("güncel serial port: " + portname, 12, y); y -= dy;  }    void keyPressed()  {  if (key == '1') speed = 0.001;  if (key == '2') speed = 0.01;  if (key == '3') speed = 0.1;    if (!streaming) {  if (keyCode == LEFT) port.write("G91\nG20\nG00 X-" + speed + " Y0.000 Z0.000\n");  if (keyCode == RIGHT) port.write("G91\nG20\nG00 X" + speed + " Y0.000 Z0.000\n");  if (keyCode == UP) port.write("G91\nG20\nG00 X0.000 Y" + speed + " Z0.000\n");  if (keyCode == DOWN) port.write("G91\nG20\nG00 X0.000 Y-" + speed + " Z0.000\n");  if (keyCode == KeyEvent.VK\_PAGE\_UP) port.write("G91\nG20\nG00 X0.000 Y0.000 Z" + speed + "\n");  if (keyCode == KeyEvent.VK\_PAGE\_DOWN) port.write("G91\nG20\nG00 X0.000 Y0.000 Z-" + speed + "\n");  if (key == 'h') port.write("G90\nG20\nG00 X0.000 Y0.000 Z0.000\n");  if (key == 'v') port.write("$0=75\n$1=74\n$2=75\n");  //if (key == 'v') port.write("$0=100\n$1=74\n$2=75\n");  if (key == 's') port.write("$3=10\n");  if (key == 'e') port.write("$16=1\n");  if (key == 'd') port.write("$16=0\n");  if (key == '0') openSerialPort();  if (key == 'p') selectSerialPort();  if (key == '$') port.write("$$\n");  }    if (!streaming && key == 'g') {  gcode = null; i = 0;  File file = null;  println("Loading file...");  selectInput("Select a file to process:", "fileSelected", file);  }    if (key == 'x') streaming = false;  }    void fileSelected(File selection) {  if (selection == null) {  println("Window was closed or the user hit cancel.");  } else {  println("User selected " + selection.getAbsolutePath());  gcode = loadStrings(selection.getAbsolutePath());  if (gcode == null) return;  streaming = true;  stream();  }  }    void stream()  {  if (!streaming) return;    while (true) {  if (i == gcode.length) {  streaming = false;  return;  }    if (gcode[i].trim().length() == 0) i++;  else break;  }    println(gcode[i]);  port.write(gcode[i]+ '\n');  i++;  }    void serialEvent(Serial p)  {  String s = p.readStringUntil('\n');  println(s.trim());    if (s.trim().startsWith("ok")) stream();  if (s.trim().startsWith("error")) stream(); // XXX: really?  } |

KAYNAKLAR

# [1] https://circuitdigest.com/microcontroller-projects/arduino-cnc-machine-project-code

# [2] https://www.robimek.com/cizim-yapan-mini-cnc-yapimi/