REX

REX EVOLUTION SERIES
SUPER STAR TRANSFORMERS
8 IN 1

MonsterBot

Monster Robot Application Calculating Slope

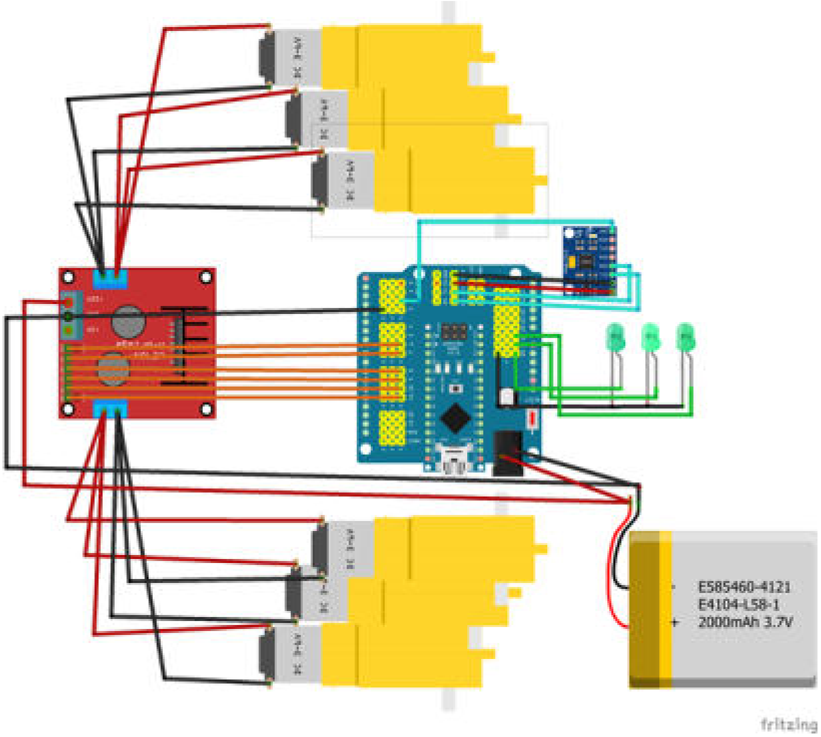
Author:**Mustafa Kemal AVCI**

When climbing a ramp it may be necessary to give more power to the motors than on level ground. In this case, Monster can decide for himself. Likewise, with the power on the flat ground, the robot may land faster than expected, and the room may cause the robot to hit the ground quickly.

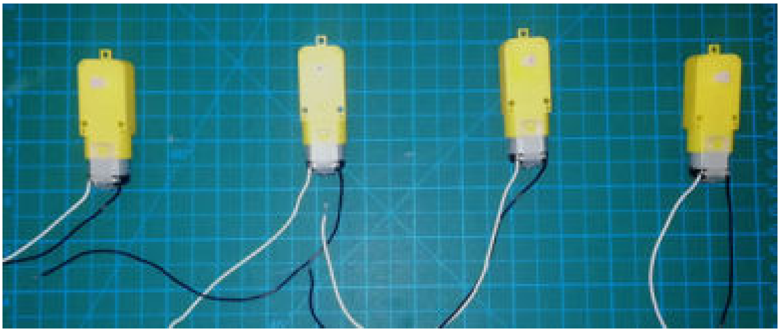
In this application, the monster robot recognizes the ramp and gives more power to the motors, detects the descent and reduces the speed of the motors.

After completing the installation of the Monster robot, let's fix the MPU6050 Gyro sensor close to the front of the robot with its integrated upwards and make the cable connections like that. Positioning the gyro sensor forward is necessary to detect slope changes more quickly.

Connection Diagram



Connect the (+) poles of the 3 motors on the right side to Out 1 (-) the poles to Out 2, the (+) poles of the 3 motors on the left side to Out 3 and the (-) poles with the Out 4. We will control the Out1, out2, out3 and out4 outputs on the motor driver (L298N) with the IN1, IN2, IN3 and IN4 pins, respectively. Our L298N motor driver card can deliver 2A current per channel. If we don't push our motors too hard, our robot will be able to use all its wheels. For directional control of right motors, connect IN1 and IN2 pins to digital pins D7 and D8 on the sensor shield, respectively. Connect IN3 and IN4 pins to D9 and D10 digital pins for left motor direction control. Connect ENA pin to D6 pin for right motor speed control and ENB pin to D11 pin for left motor speed control.

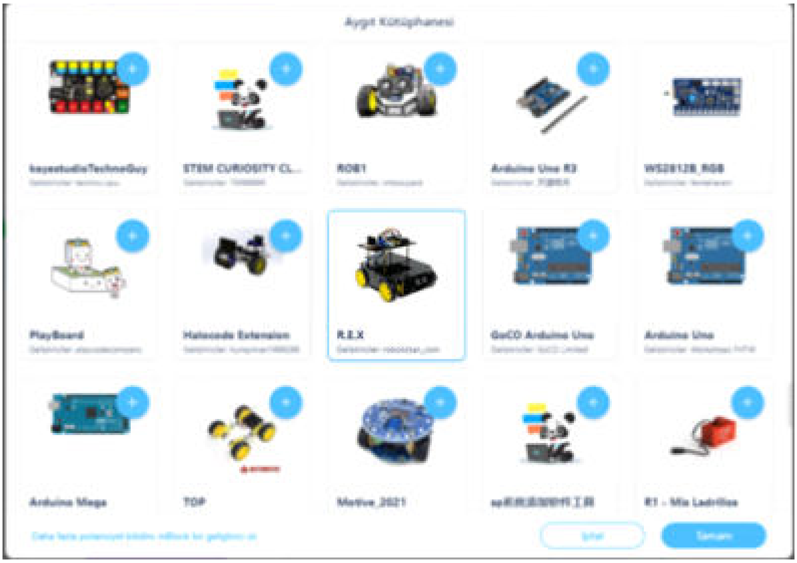


You should solder the motors with 25-30 cm cable as in the photo. We will use the white leads (+) and the black leads (-) as poles. In this case, when we connect the (+) black cable (-) pole to the white wires of the motors, the wheels rotate clockwise.

Let's connect the long legs of the leds, which we will use to learn the degree of slope, to the A0, A1 and A2 pins, respectively, and the short legs to any of the GND lines.
VCC pin of MPU 6050 Gyro sensor must be connected to 5V, GND pin to G pin, SDA pin to SDA and SCL pin to SCL pin on the shield as shown in the circuit diagram. In addition, we must connect the INT pin to pin 2 on the shield.

Coding the Robot

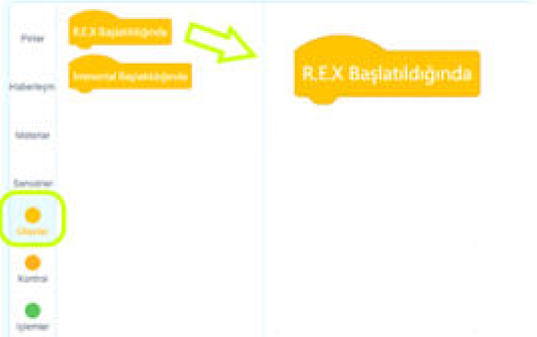
Let's start the mBlock 5 software and add R.E.X from the device library and move on to the coding stage.



Algoritmamız şu şekilde olacak;

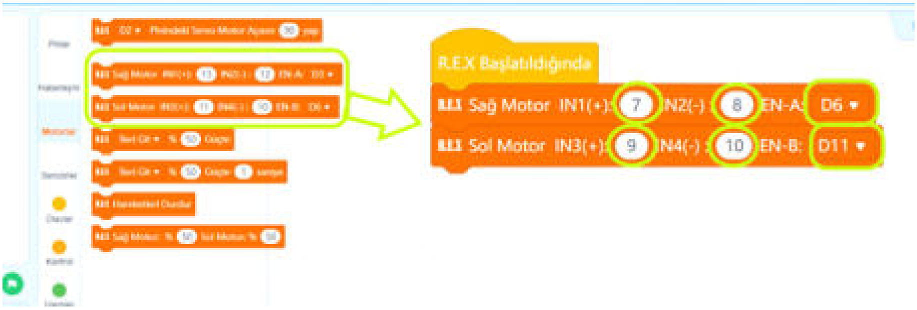
1. Start
2. Identify engines
3. Start accelerometer
4. Create the speed variable
5. Read the accelerometer
6. If the X-axis angular acceleration value is in the range of 90-110, set the speed to 40 and turn on the A0 led.
7. If the X-axis angular acceleration value is between 110-130, set the speed to 50 and turn on the A1 led.
8. If the X-axis angular acceleration value is greater than 130, set the speed to 60 and turn on the A2 led.
9. If the X-axis angular acceleration value is between 90 - 70, set the speed to 35 and turn on the A1 led.
10. If the X-axis angular acceleration value is less than 70, set the speed to 30 and turn on the A2 led.
11. Go ahead at speed
12. Go to step 5
13. Stop

Drag and drop the When REX Starts block from the event blocks to the coding area.



R.E.X Başlatıldığında

Drag and drop the blocks in which we defined the Right and Left Motor control pins in the "Motors" category, under the "When REX Starts" block, respectively, as in the image. Write the pin definitions in the circuit diagram to the necessary places as in the image.

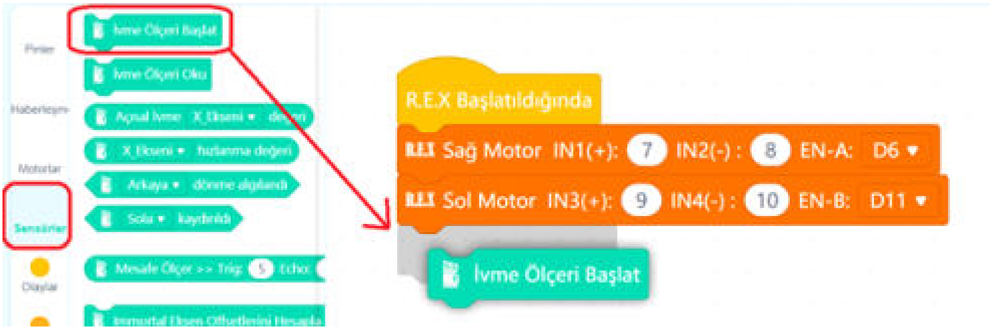


R.E.X Başlatıldığında

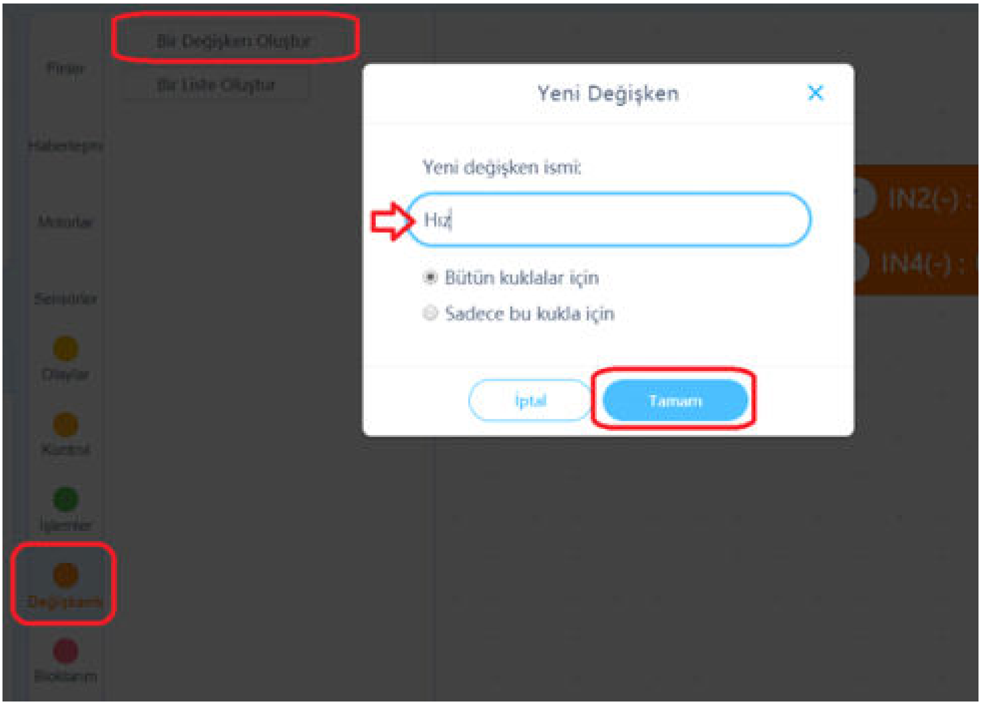
R.E.X Sağ Motor IN1(+): 7 IN2(-): 8 EN-A: D6

R.E.X Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11

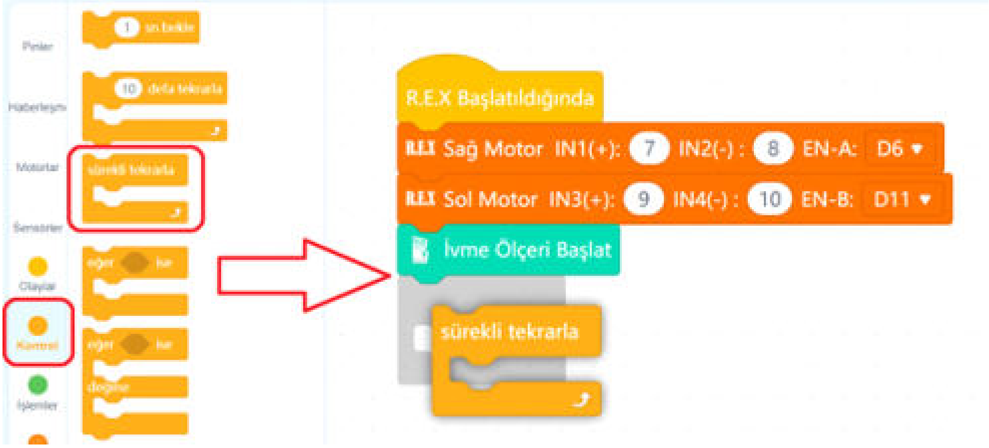
To define the acceleration sensor that will calculate the slope, let's drag and drop the Accelerometer start block in the Sensors category as in the image below.


Pinler
Haberleşm
Motorlar
Sensörler
Olaylar
İvme Ölçeri Başlat
İvme Ölçeri Oku
Açısal İvme X_Ekseni ▾ değeri
X_Ekseni ▾ hızlanma değeri
Arkaya ▾ dönme algılandı
Sola ▾ kaydırıldı
Mesafe Ölçer >> Trig: 5 Echo:
Immortal Eksen Offsetlerini Hesapla
R.E.X Başlatıldığında
Sağ Motor IN1(+): 7 IN2(-) : 8 EN-A: D6 ▾
Sol Motor IN3(+): 9 IN4(-) : 10 EN-B: D11 ▾
İvme Ölçeri Başlat

After defining the accelerometer, we must create the variable that we will carry the speeds of the motors. In the Variables category, click the Create a variable button, type the name of the variable as in the image below, and press the OK button.

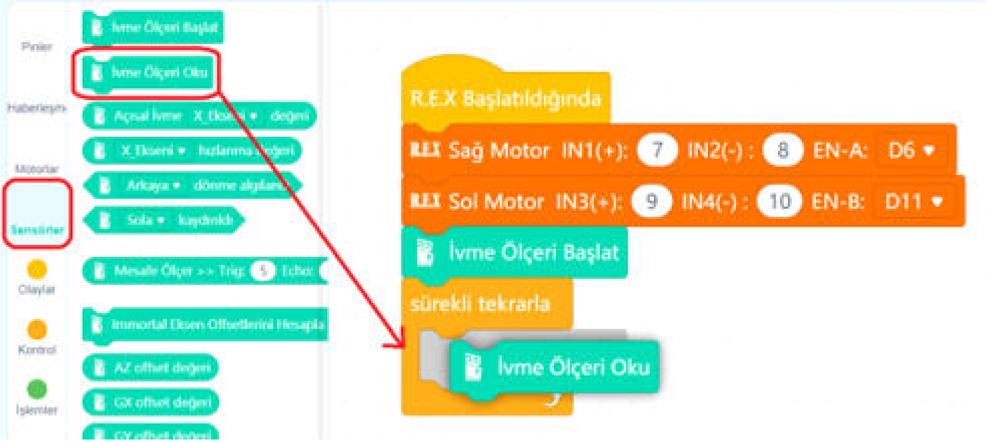


Our acceleration sensor will be read continuously and the value read will be subjected to a mathematical test. We will perform these operations in an infinite loop. For this, let's take the repeat block from the control blocks and put it at the bottom of our code blocks.



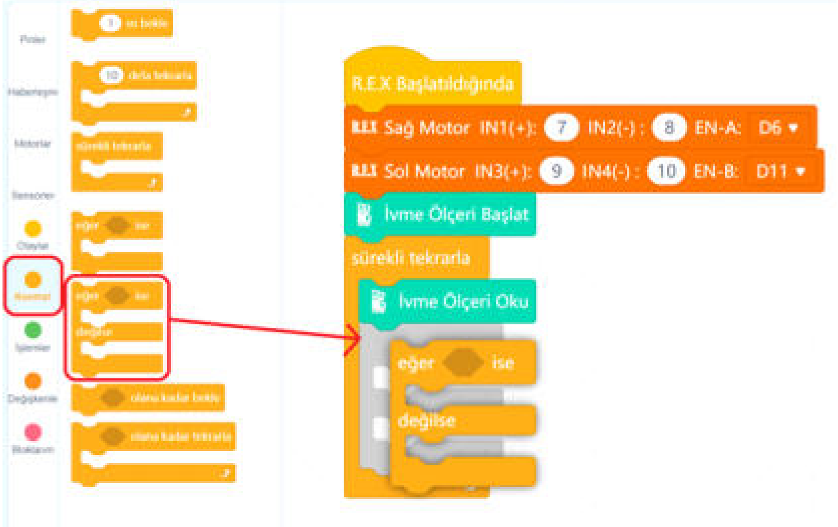
```
R.E.X Başlatıldığında
  R.E.X Sağ Motor IN1(+): 7 IN2(-) : 8 EN-A: D6
  R.E.X Sol Motor IN3(+): 9 IN4(-) : 10 EN-B: D11
  İvme Ölçeri Başlat
  sürekli tekrarla
```

We get the angular acceleration values on the axes by reading the accelerometer. Because in the next step, we will determine our velocity value by comparing this angular acceleration value.



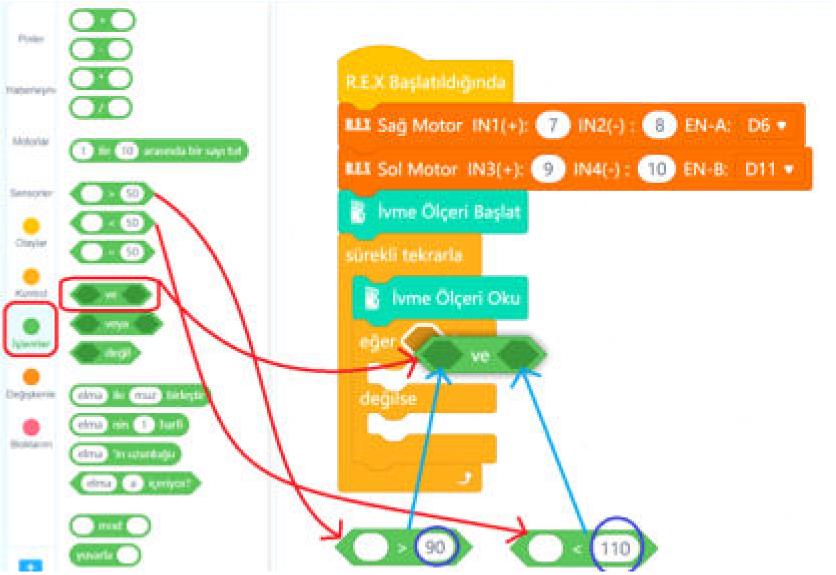
```
R.E.X Başlatıldığında
  R.E.X Sağ Motor IN1(+): 7 IN2(-) : 8 EN-A: D6
  R.E.X Sol Motor IN3(+): 9 IN4(-) : 10 EN-B: D11
  İvme Ölçeri Başlat
  sürekli tekrarla
    İvme Ölçeri Oku
```

We will adjust the velocity variable by comparing the Angular acceleration values we have obtained. Since both conditions will not be fulfilled at the same time, we will check each condition if the previous condition is not met. This will also ensure that the software of our robot works more healthily. Let's place the If If not block in the control blocks under the Accelerometer arrow block.



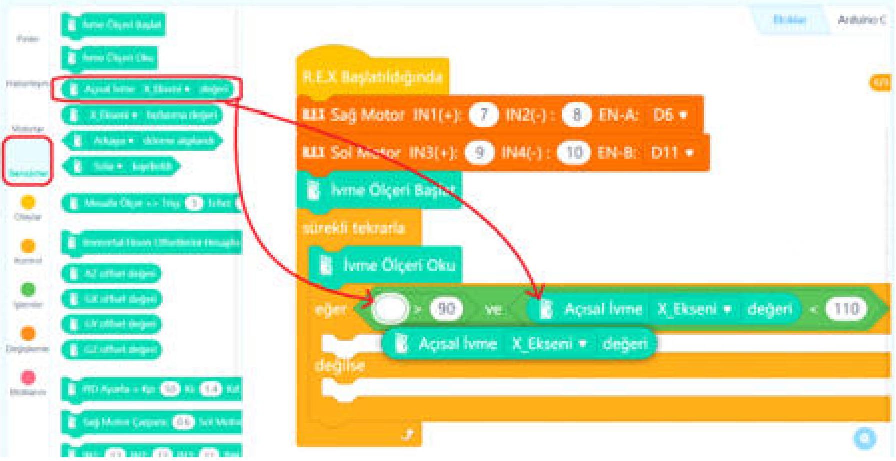
```
R.E.X Başlatıldığında
REX Sağ Motor IN1(+): 7 IN2(-) : 8 EN-A: D6 ▾
REX Sol Motor IN3(+): 9 IN4(-) : 10 EN-B: D11 ▾
İvme Ölçeri Başlat
sürekli tekrarla
    İvme Ölçeri Oku
    eğer ⬡ ise
    değilse
```

Our first condition is that the X-axis accelerometer value is between 90 and 110. In this case, the robot will be perceived as going on flat ground and the motors will be operated at 40 power. Place the comparison operators correctly in the appropriate place in your code as in the image below.

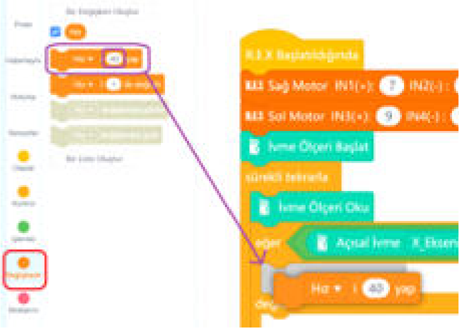


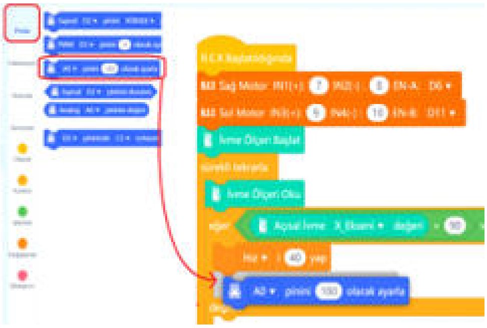
```
R.E.X Başlatıldığında
REX Sağ Motor IN1(+): 7 IN2(-) : 8 EN-A: D6 ▾
REX Sol Motor IN3(+): 9 IN4(-) : 10 EN-B: D11 ▾
İvme Ölçeri Başlat
sürekli tekrarla
    İvme Ölçeri Oku
    eğer ⬡ ve ⬡
    değilse

( ) > 90
( ) < 110
```

Next, let's drag and drop the X-axis Angular Acceleration value block from the Sensors category to the comparison area of the greater and lesser operators.

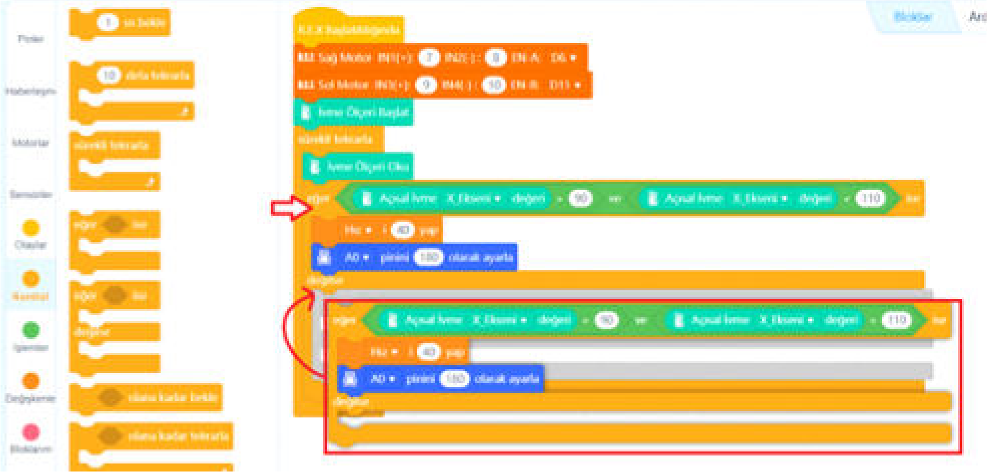

R.E.X Başlatıldığında
Sağ Motor IN1(+): 7 IN2(-): 8 EN-A: D6
Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11
İvme Ölçeri Başlat
sürekli tekrarla
İvme Ölçeri Oku
eğer > 90 ve Açısal İvme X_Ekseni değeri < 110
Açısal İvme X_Ekseni değeri
değilse
Arduino C

When this condition is met, our Speed variable value will be 40 according to the 6th step of the algorithm and we will turn on the LED on the A0 pin. Let's drag and drop the Set the Speed value to 40 block in the Variables category and the Set A0 pin to 180 block in the Pins category to the Commands that will run if the condition is true.

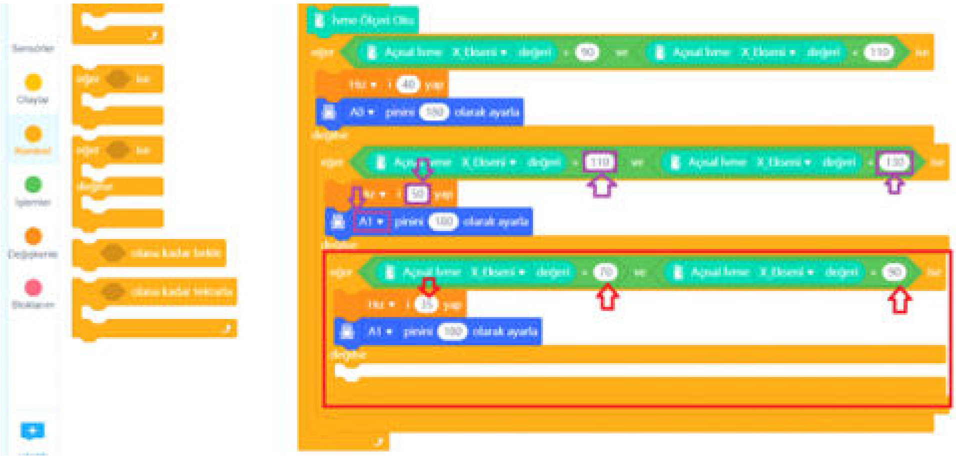



R.E.X Başlatıldığında
Sağ Motor IN1(+): 7 IN2(-): 8 EN-A: D6
Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11
İvme Ölçeri Başlat
sürekli tekrarla
İvme Ölçeri Oku
eğer Açısal İvme X_Ekseni değeri 90
Hız 40 yap
A0 pinini 180 olarak ayarla

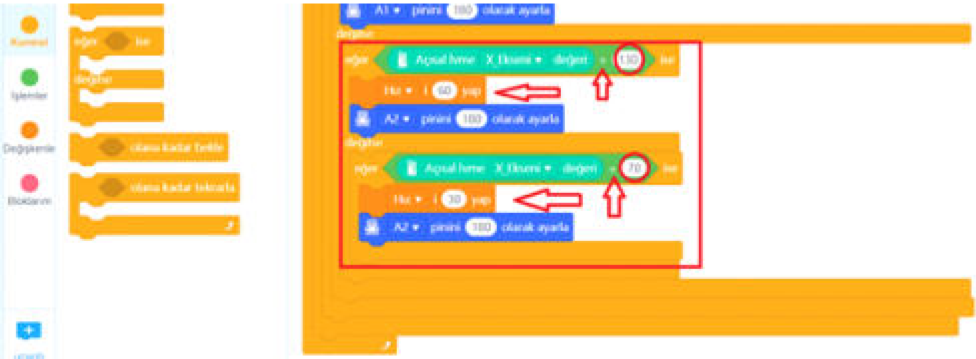
When the X-axis angular acceleration value is between 110 and 130, we make a speed of 50 and turn on the led connected to A1 by RIGHT-CLICKING on an orange part of the If If If Not block, copy it as in the image and insert it in the area if it is not.



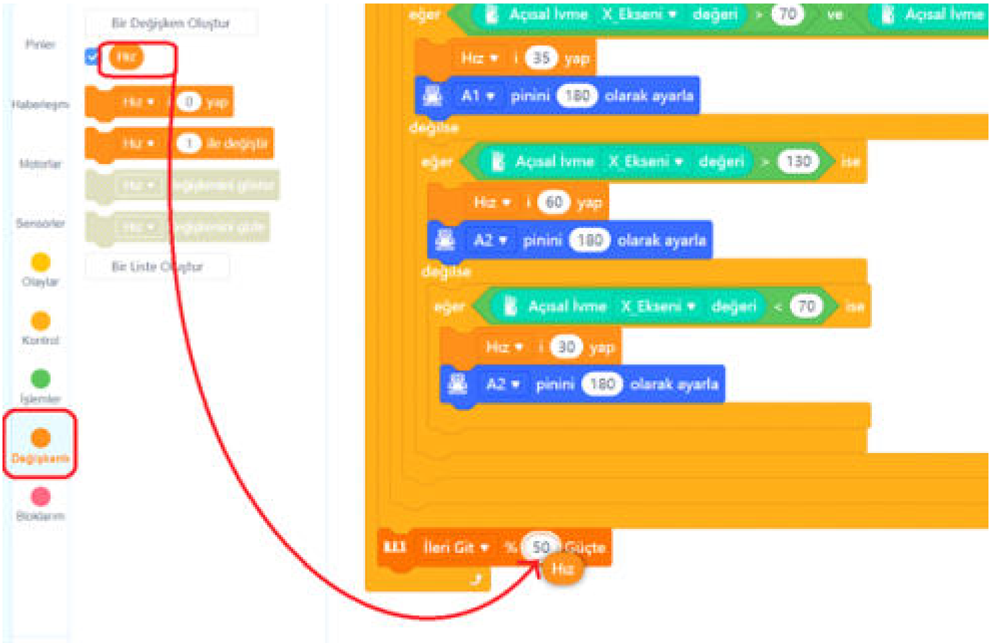
We replace 90 in the greater and lesser operator blocks with 110 and 110 with 130. Then, with the same technique, let's copy our condition that the X-axis angular acceleration value is greater than 70 and less than 90, if not, let's insert it into our field and write the necessary numerical changes as in the image below.


Sensörler
Olaylar
Kontrol
İşlemler
Değişkenler
Bloklarım
eğer ise
eğer ise değilse
olana kadar bekle
olana kadar tekrarla
İvme Ölçeri Oku
eğer Açısal İvme X Ekseni değeri > 90 ve Açısal İvme X Ekseni değeri < 110 ise
Hız = 40 yap
A0 pinini 180 olarak ayarla
değilse
eğer Açısal İvme X Ekseni değeri > 110 ve Açısal İvme X Ekseni değeri < 130 ise
Hız = 50 yap
A1 pinini 180 olarak ayarla
değilse
eğer Açısal İvme X Ekseni değeri > 70 ve Açısal İvme X Ekseni değeri < 90 ise
Hız = 35 yap
A1 pinini 180 olarak ayarla
değilse

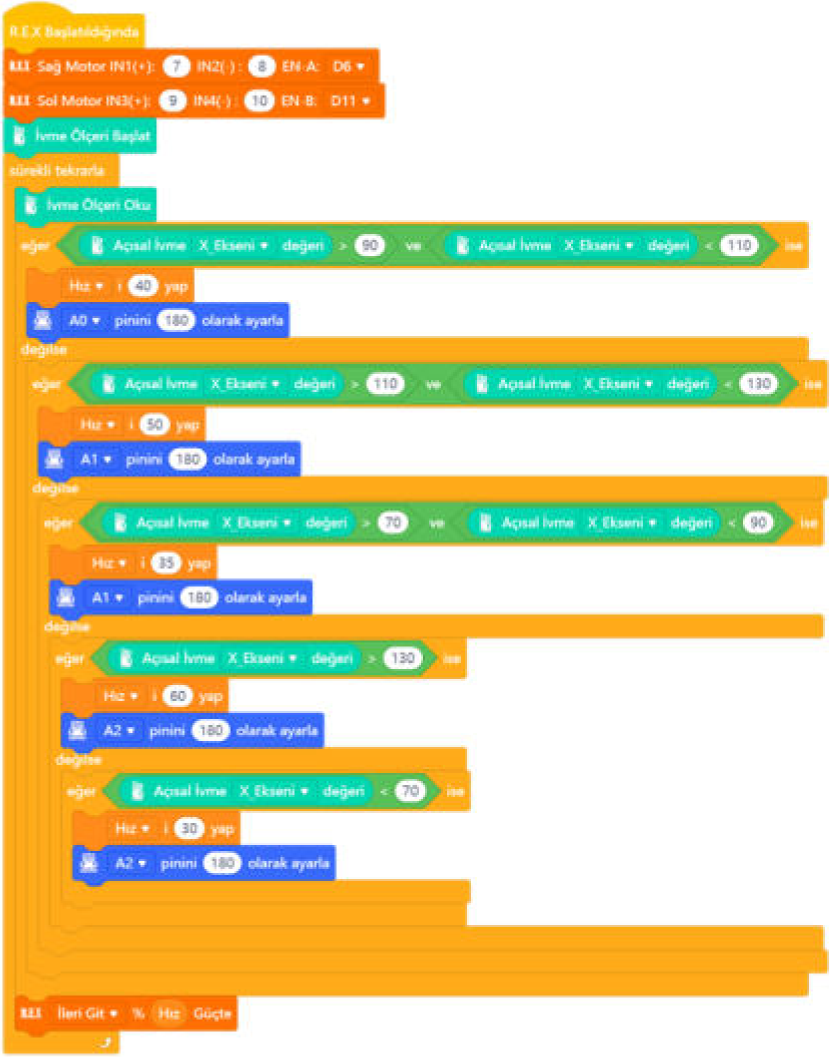
Now, when the X axis angular acceleration value is greater than 130, we will make the speed 60 and turn on the A2 led, otherwise, if the X axis angular acceleration value is less than 70, we will make the speed 30 and turn on the A2 led. Let's prepare the necessary commands by taking the necessary operators from the Operations blocks into the If If Not Block as in the image below.


Kontrol
İşlemler
Değişkenler
Bloklarım
eğer ise
değilse
olana kadar bekle
olana kadar tekrarla
A1 ▾ pinini 180 olarak ayarla
değilse
eğer Açısal İvme X Ekseni ▾ değeri > 130 ise
Hız ▾ i 60 yap
A2 ▾ pinini 180 olarak ayarla
değilse
eğer Açısal İvme X Ekseni ▾ değeri < 70 ise
Hız ▾ i 30 yap
A2 ▾ pinini 180 olarak ayarla

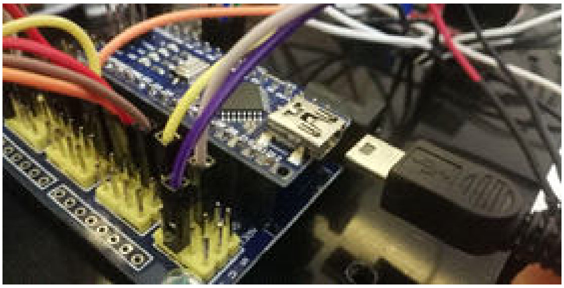
We set the value of the speed variable with the codes we have prepared so far. Finally, we will use this speed variable for our robot to go forward. As in the image below, let's drag and drop the Speed variable value block into the Repeat continuously block, instead of the number 50 in the Go Forward at 50% power block to the bottom.



The final version of the code blocks is as in the image below.

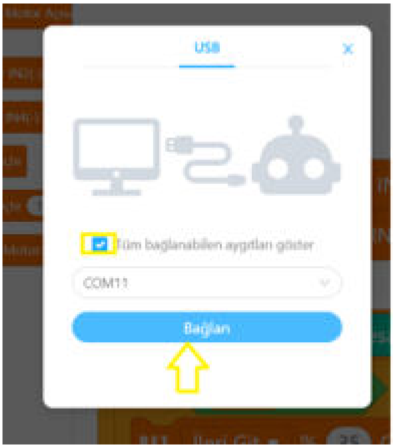


```
R.E.X Başlatıldığında
Sağ Motor IN1(+): 7 IN2(-): 8 EN-A: D6 ▾
Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11 ▾
İvme Ölçeri Başlat
sürekli tekrarla
    İvme Ölçeri Oku
    eğer < Açısal İvme X_Ekseni ▾ değeri > 90 ve Açısal İvme X_Ekseni ▾ değeri < 110 > ise
        Hız ▾ 'i 40 yap
        A0 ▾ pinini 180 olarak ayarla
    değilse
        eğer < Açısal İvme X_Ekseni ▾ değeri > 110 ve Açısal İvme X_Ekseni ▾ değeri < 130 > ise
            Hız ▾ 'i 50 yap
            A1 ▾ pinini 180 olarak ayarla
        değilse
            eğer < Açısal İvme X_Ekseni ▾ değeri > 70 ve Açısal İvme X_Ekseni ▾ değeri < 90 > ise
                Hız ▾ 'i 35 yap
                A1 ▾ pinini 180 olarak ayarla
            değilse
                eğer < Açısal İvme X_Ekseni ▾ değeri > 130 > ise
                    Hız ▾ 'i 60 yap
                    A2 ▾ pinini 180 olarak ayarla
                değilse
                    eğer < Açısal İvme X_Ekseni ▾ değeri < 70 > ise
                        Hız ▾ 'i 30 yap
                        A2 ▾ pinini 180 olarak ayarla
    İleri Git ▾ % Hız Güçte
```

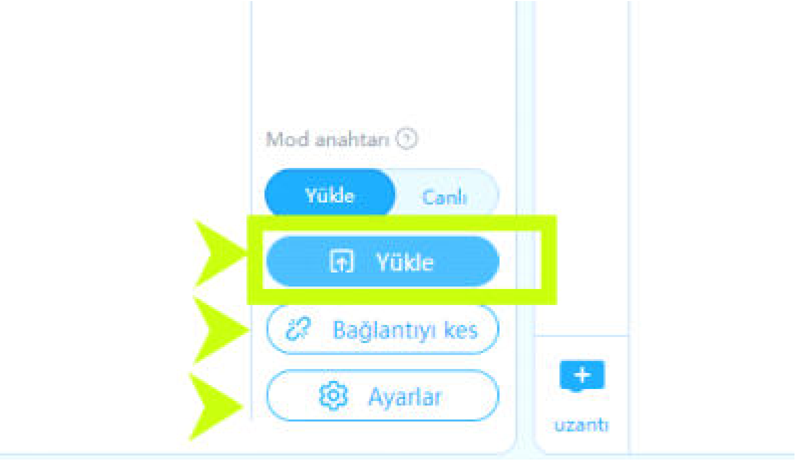


We can load our ready-made code to our robot. We connect one end of the usb cable to Arduino nano and the other end to our computer.

Click on the connect button in the installation mode and tick the show all connectable devices option. The mBlock software will automatically insert the COM port number that your CH340 chip card is connected to. If connection is not possible, you can select other COM port numbers from the drop-down list.



Once connected, “disconnect”, “settings” and “Install” buttons will be active. Let's upload the code to our robot by clicking the upload button.



Operating the Robot

After downloading the codes, connect the power. After the cables from the battery to the motor driver are made according to the connection diagram, plug the terminals from the battery into the power connection jack of the last Arduino Nano Sensor Shield to run the codes. The codes will start working immediately.

Possible Problems and Solutions

If your robot is going backwards with the Forward command;

Make sure that OUT1 and OUT 3 on the motor driver are connected to the (+) pole of the motors, and OUT 2 and OUT 4 are connected to the (-) pole of the motors. Also, make sure that the motors on the right of the robot are connected to OUT1 and OUT2 on the driver, and the motors on the left are connected to OUT3 and OUT4.

If the robot slows down while going up the ramp, and speeds up when going down the ramp;

Values from the gyro sensor are inverted means r. This is because the gyro sensor is not positioned as shown in the image. You should place the side to which the Pins of your sensor are connected, to the inside of the robot, facing the back, and with the screws to the end, and the IC to be on top.

When you energize the robot, the LEDs on the Motor driver, Shield and Arduino Nano light up but do not move at all or go at a constant speed;

Make sure that the ENA and ENB pins of the motors are connected correctly and the pin definitions are made correctly in the code blocks.

Make sure to place the forward 50% power block that we placed last, and the Speed variable block in it.

YouTube

youtube.com/robotistan

FORUM

robotistan

forum.robotistan.com

BLOG

robotistan

maker.robotistan.com

Robotistan Electronics Company

Mustafa Kemal AVCI (Content) - Fadıl PALA - Mehmet AKÇALI (Editor) - (Mehmet Nasır KARAER (Graphics) - Mert ALTUNTAS (Translation)
info@robotistan.com - www.robotistan.com
Phone: 0850 766 0 425