

REX

REX EVOLUTION SERIES
SUPER STAR TRANSFORMERS
8 IN 1

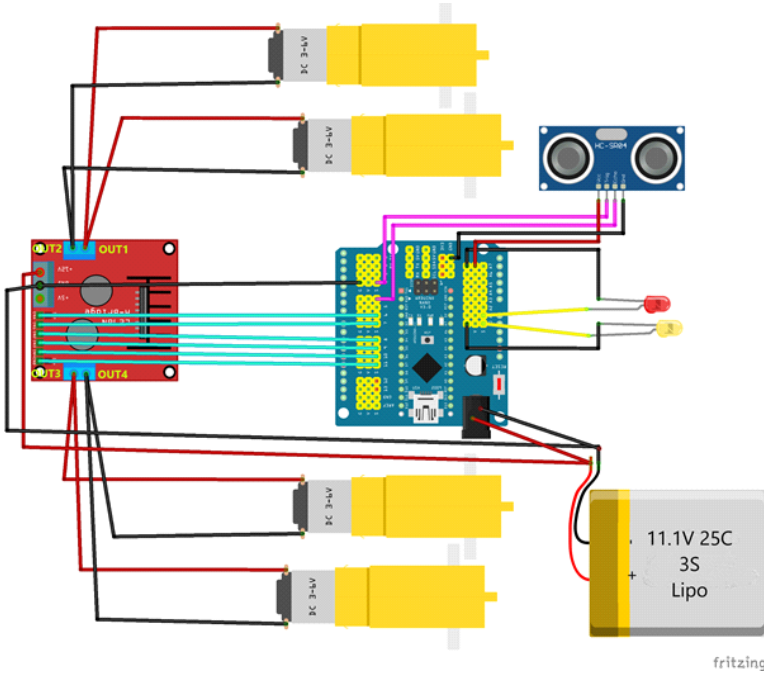
Engelden
Kaçan
Survivor
Robot

RoboSonic

Yazar: Mustafa Kemal AVCI

Bağlantı Şeması

Survivor Robotumuzun parçalarını bir araya getirip elektronik bileşenlerini monte ettikten sonra kablo bağlantılarını yaparak kodlamaya başlayalım. Aşağıdaki Engelleri aşan Survivor robotun bağlantı şemasına göre kablo bağlantılarını hazırlayalım.



Robotun sağdaki motorların + kutupları OUT1, - kutupları OUT2 klemensine takarak vidalarını sıkın. Robotun solundaki motorların ise + kutuplarını OUT3 - kutuplarını ise OUT4 klemensine takarak vidasını sıkın.

Motor sürücü kartı (L298N) üzerindeki OUT1 ve OUT2 çıkışları IN1 ve IN2 pinlerinden kontrol edilmektedir. OUT3 ve OUT4 çıkışları ise IN3 ve IN4 pinlerinden kontrol edilmektedir.

Sağ motorun dönme yönünün ayarlandığı IN1 ve IN2 pinlerini sensör shield üzerindeki 7 ve 8 nolu dijital pine bağlayalım. Sol motorun dönme yönünün ayarlandığı IN3 ve IN4 pinlerini ise 9 ve 10 nolu dijital pine bağlayalım.

Hız ayarı için kullanılacak pinlerin PWM desteği olması gerekmektedir. Arduino Nano kartındaki PWM pinleri 3,5,6,9,10 ve 11. pinlerdir.

Sağ motorun dönüş hızının ayarlandığı ENA pini 6 nolu dijital pine, sol motorun dönüş hızının ayarlandığı ENB pinini ise 11 nolu dijital pine bağlayalım.

Engelden kaçan Survivor Robot

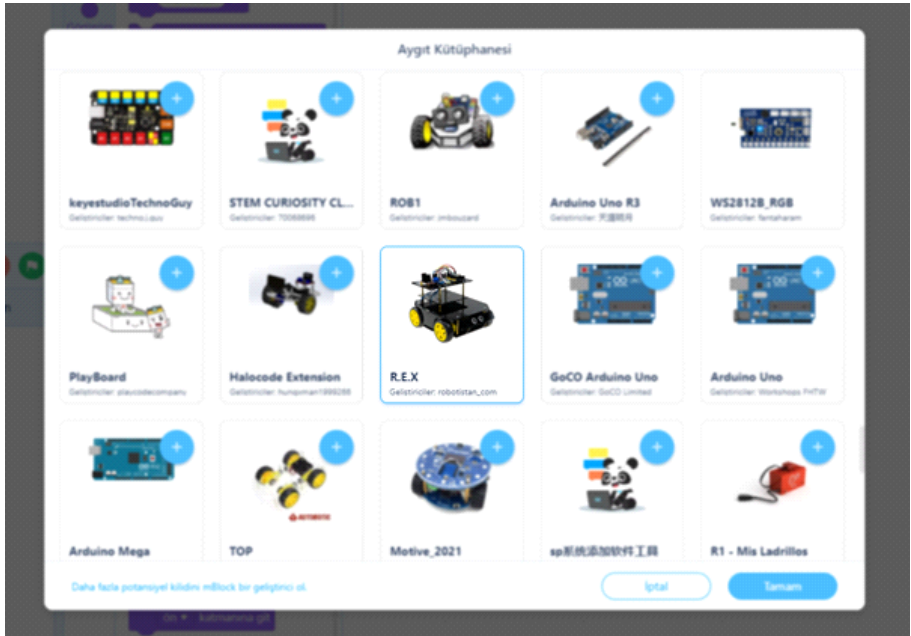
Hem engel algılama , tepki verme gibi işlemleri bildirmek hem de kodda oluşabilecek hataları denetlerken kolaylık sağlaması adına robotumuza iki adet led bağladık. Birinin uzun bacağını A0 analog pinine, diğerinin uzun bacağını ise A1 analog pinine bağlayalım. Kısa bacakları ise sensör shield üzerinde G ile gösterilen GND pinlerinden uygun olana bağlayabilirsiniz.

Robotun önüne çıkan engelleri algılayabilmesi için ultrasonik mesafe sensörü kullanacağız. Ultrasonic sensör ses dalgaları yayar ve geri yansıyan ses dalgasının geri geliş süresinden ses hızını hesaplayarak bize bildirir.Sensörün adı HC-SR 04. Bu sensör 4 cm ile 4m arasında ölçümler yapabilir. Trig pinini 3 nolu dijital pine Echo pinini ise 4 nolu dijital pine bağlayalım. GND pinini shield üzerinde G ile gösterilen bir pine, VCC pinini ise V ile gösterilen 5V pinlerinden birine bağlayalım.

7-12V aralığındaki pil gücünüzü (Lipo Pil veya 6x AA kalem pil) motor sürücünüze ve Arduino Nano sensör shield a aktarmalısınız. Bunun için pilinizi güç çıkışlarını 2 li hale getirmelisiniz. + ve - kutuplarını birer tanesini motor sürücü üzerinde şemada gösterilen klamenslere takmalısınız. Diğerini ise shield ın güç girişine jag olarak takabilir hale getirmelisiniz. Son olarak da shield üzerindeki bir G pininden yani GND pininden motor sürücü üzerindeki GND klamensine bir kablo ile ortak hat kurmalısınız.

Kodlama

mBlock yazılımını çalıştırdıktan sonra aygıt kütüphanesine girerek R.E.X aygıtını çalışma alanımıza ekleyelim.



Engelden kaçan Survivor Robot

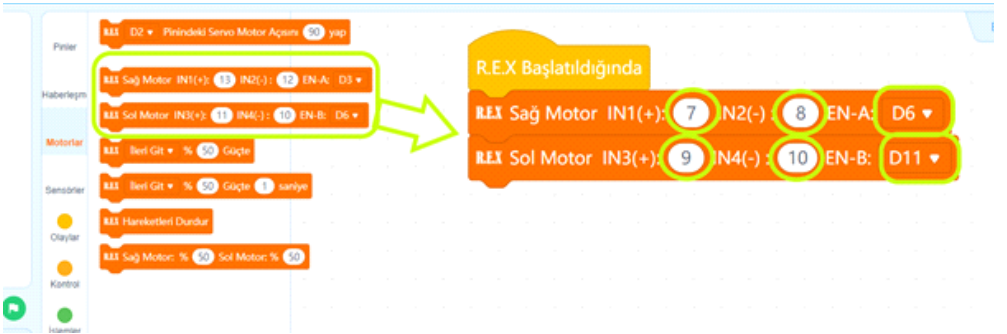
Algoritmamız şu şekilde olacak.

1. Başla
2. Motor pinlerini tanımla
3. Mesafe değişkenini tanımla
4. Sürekli ultrasonik mesafe sensörünü oku ve değerini mesafe değişkenine ata
5. Mesafe değişkeni değeri 30 cm'den az ve 2 cm'den fazla ise sağa veya sola dön. (Sağa veya sol dönme durumu rastgele robot tarafından seçilecek)
6. Mesafe 30 cm'küçük değilse ise ileri yönde hareket ettir.
7. Bitir.



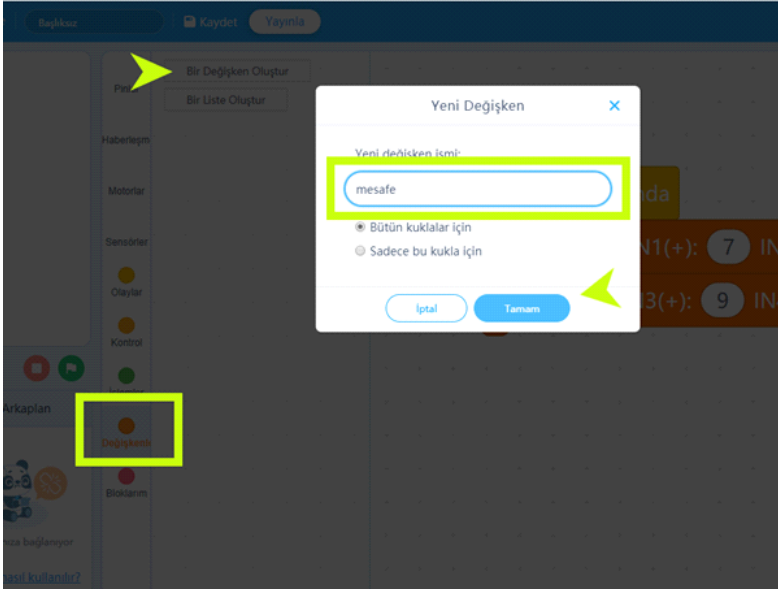
Olay bloklarından “R.E.X Başlatıldığında” bloğunu kodlama alanına sürükleyip bırakın.

“Motorlar” kategorisindeki Sağ ve Sol Motor kontrol pinlerini tanımladığımız blokları sırasıyla görseldeki gibi “R.E.X Başlatıldığında” bloğunun altına sürükleyip bırakın. Devre şemasındaki pin tanımlamalarını gerekli yerlere görseldeki gibi yazınız.



Ultrasonik mesafe sensörünün okuyacağı mesafe değerini saklamak için mesafe değişkeni oluşturacağız. Bunun için “Değişkenler” kategorisine girip “Bir Değişken oluştur” butonuna tıklıyoruz. Açılan pencerede değişken adını mesafe olarak yazıp tamam butonuna basıyoruz. Böylelikle değişkenimizi tanımlamış ve ilk değerini 0 olarak belirlemiş olduk.

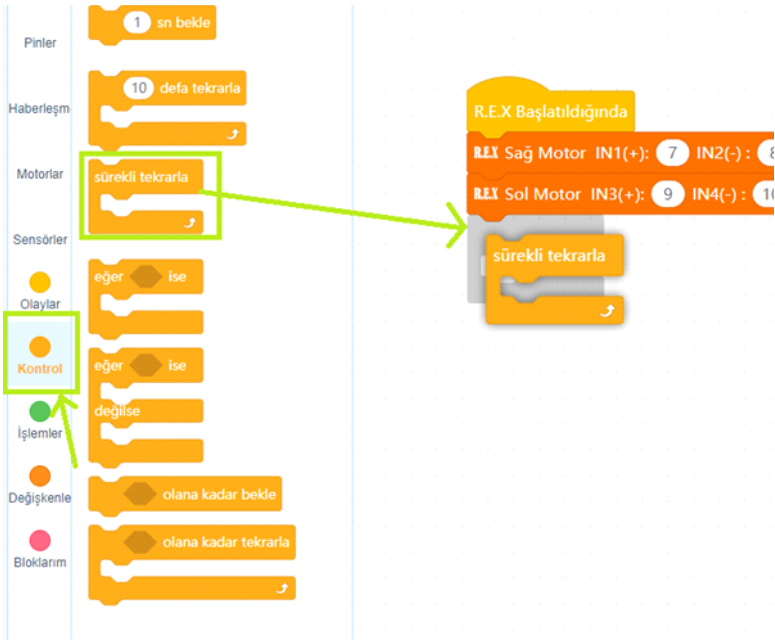
Engelden kaçan Survivor Robot



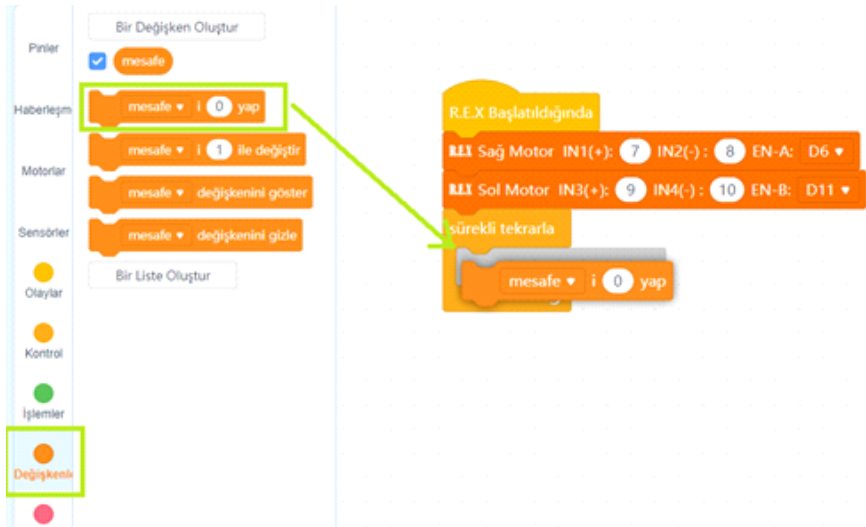
değişken adını “mesafe” olarak yazıp tamam butonuna bastığımızda, “Değişkenler” kategorisi blokları şu şekilde görünmelidir.

Sürekli olarak mesafe okunup mesafenin durumuna göre motorlara yön ve hız verileceği için motor tanımlama bloklarının hemen altına “Kontrol” blokları kategorisinden “Sürekli tekrarla” döngüsü bloğunu yerleştiriyoruz.

Engelden kaçan Survivor Robot



mesafe değişkeninin değerini sürekli olarak sensörden okunan değeri atayacağız. Program akışında mesafe değeri ne zaman lazım olursa o değişkeni kullanacağız. “Değişkenler” kategorisindeki “mesafe ' i 0 yap” bloğunu görseldeki gibi sürekli tekrarla bloğunun içine yerleştirelim.



Engelden kaçan Survivor Robot

Mesafe ölçme sensörü (HC-SR04) ölçümü bize cm'nin yüzde biri hassasiyetinde vermektedir. Örneğin 36,42 cm. "Sensörler" kategorisinden mesafe ölçer bloğunu çalışma alanımıza aşağıdaki görseldeki gibi taşıyalım. Devre şemamızda da belirlediğimiz gibi Trig pinini bağladığımız pin numarasını 3, Echo pinini bağladığımız pin numarasını 4 olarak değiştirelim.

Scratch Bloklar Paneli (Sensörler):

- Mesafe Ölçer >> Trig: 3 Echo: 4
- Immortal Eksen Olfaktörleri Hesapla
- AZ offset değeri
- GK offset değeri
- GY offset değeri
- GZ offset değeri
- PID Ayarla -> Kp: 50 Ki: 15 Kd: 1
- Sağ Motor Çarpımı: 0.6 Sol Motor Çarpımı: 0.6
- IN1: 11 IN2: 12 IN3: 11 IN4: 11
- ENA: 1 ENB: 4
- Kalite Et -> AZ: 1201 GK: 143
- Dengelece Kal

Scratch Bloklar Alanı:

- 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
- sürekli tekrarla
- mesafe i 0 yap
- Mesafe Ölçer >> Trig: 3 Echo: 4 (cm)

Ondaklıkları ifade ettiğimiz bu sayının karşımıza tam sayı olarak çıkması robotun bize daha stabil ölçümler yapmasını sağlar. Bunun için "Yuvarla" bloğunu kullanacağız. "İşlemler" kategorisinden yuvarla bloğunu alıp çalışma alanımıza aşağıdaki gibi taşıyoruz.

Scratch Bloklar Paneli (İşlemler):

- Yuvarla
- mod
- rin: mutlak değer

Scratch Bloklar Alanı:

- 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
- sürekli tekrarla
- mesafe i 0 yap
- Yuvarla
- Mesafe Ölçer >> Trig: 3 Echo: 4 (cm)

Engelden kaçan Survivor Robot

mesafe ölçer bloğunu öncelikle yuvarla bloğunun içine görseldeki gibi sürükleyip bırakalım. Mesafe Ölçer bloğu yuvarla bloğunun içine oturacak ve birlikte hareket edebilir hale gelecektir. Daha sonra yuvarla bloğunu yeşil kısımdan tutarak mesafe'i 0 yap bloğunun değer alanına sürükleyip bırakalım. Böylece mesafe değişkenimizin değerinin mesafe sensöründen okunan değerin yuvarlanmış hali yapmış oluyoruz.

The screenshot shows the Scratch code editor with a library on the left and a script area on the right. The library includes categories like Sensörler, Olaylar, Kontrol, İşlemler, Değişkenler, and Blokaram. The script area contains a 'Sürekli Tekrarla' block with a 'mesafe' variable set to 0. A 'yuvarla' block is being dragged from the 'İşlemler' category into the 'mesafe' block's value field. A red arrow labeled '2' points to the 'yuvarla' block, and another red arrow labeled '1' points to the 'Mesafe Ölçer >> Trig: 3 Echo: 4 (cm)' block.

Artık matematiksel sınama yapacağımız blokları hazırlayabiliriz. Mesafe değişkeni değeri belirli bir aralıktaysa motorların engelden kaçabilmesi için sağa veya solda dönmesi davranışı sergilemesini belirli bir aralıktan değilse ileri gitme davranışını sergilemesini istiyoruz. Yani sınavadığımız koşul doğru ise bir komut çalıştırılacak, yanlışsa da farklı bir komut çalıştırılacaksa "Kontrol" blokları kategorisinde "Eğer ise Değilse" bloğunu mesafe değişkeni bloğunun altına sürükleyip bırakın.

The screenshot shows the Scratch code editor with the 'Eğer ise Değilse' block added to the script area. The 'Eğer ise' block is highlighted with a green box. The 'mesafe' variable is now set to 'yuvarla Mesafe Ölçer >> Trig: 3 Echo: 4 (cm)'. The 'Eğer ise Değilse' block is positioned below the 'mesafe' block, with the 'Eğer ise' block containing a 'Sürekli Tekrarla' block and the 'Değilse' block containing a 'Sürekli Tekrarla' block.

Yapacağımız matematiksel sınama şöyle olacak; mesafe değişkeni değeri 2 cm'den büyükse ve 30 cm'den küçükse. Bu sınama mesafe değerinin 2 ile 30 cm arasında olup olmadığını kontrol et anlamındadır. Bunun kod bloklarıyla ifade edebilmek için "işlemler" kategorisinden "ve" operatör bloğunu görseldeki gibi eğer ise değilse bloğunun koşul alanına taşıyalım.

Engelden kaçan Survivor Robot

Scratch environment showing the initial code for the Survivor Robot. The code starts with 'R.E.X Başlatıldığında' (When R.E.X is started) and includes two 'R.E.X' blocks for the right and left motors. A 'sürekli tekrarla' (Repeat Forever) loop contains a 'mesafe' (distance) block set to 'yuvarla' (round) and 'Mesafe Ölçer >> Trig: 3 Echo: 4'. Below this is an 'eğer' (if) block with a 've' (and) condition, and a 'değilse' (otherwise) block.

iki ayrı koşulumuz vardı. mesafenin 2 den büyük olması ve mesafenin 30 dan küçük olması. iki koşulumuzu aşağıdaki gibi karşılaştırma operatörlerinden küçüktür ve büyüktür operatörlerini ve bloğunun içine sürükleyip bırakalım.

Scratch environment showing the updated code for the Survivor Robot. The code is similar to the previous one, but the 'eğer' (if) block now has two conditions: '> 50' and '< 50'. The 'değilse' (otherwise) block is also present.

büyüktür ve küçüktür karşılaştırma operatörünün içindeki boş alanlara mesafe değişkenimizi yerleştireceğiz. Bunun için değişkenler kategorisindeki mesafe değişkenini değer bloğunu sürükleyip aşağıdaki görselde gösterilen yerlere bırakalım. 50 sayılarını ise 2 ve 30 ile değiştirelim.

Engelden kaçan Survivor Robot

Bir Değişken Oluştur

mesafe

Haberleşim

mesafe i 10 yap

Motorlar

mesafe i 10 de değiştir

mesafe de değişkenini göster

Sensörler

mesafe de değişkenini gütle

Bir Liste Oluştur

R.E.X Başlatıldığında

BELI Sağ Motor IN1(+): 7 IN2(-): 8 EN-A: D6

BELI Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11

sürekli tekrarla

mesafe i yuvarla Mesafe Ölçer >> Trig: 3 Echo: 4

eğer mesafe > 2 ve mesafe < 30 ise

değilse

mesafe değerimiz 2 ile 30 arasında ise robotumuz rastgele bir tarafa dönme işlemi gerçekleştirecek, değilse ileri yönde gitmeye devam edecektir. Ta ki önüne 30 cmden yakında duran bir engel çıkana kadar. İleri git %40 güçte bloğunu görseldeki gibi “değilse” komut alanına sürükleyip bırakalım.

D2 Pinindeki Servo Motor Açsın

Sağ Motor IN1(+): 13 IN2(-): 12 EN-A: D6

Sol Motor IN3(+): 11 IN4(-): 10 EN-B: D11

İleri Git % 50 Güçte

İleri Git % 50 Güçte

Hareketleri Durdur

Sağ Motor: % 50 Sol Motor: % 50

R.E.X Başlatıldığında

BELI Sağ Motor IN1(+): 7 IN2(-): 8 EN-A: D6

BELI Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11

sürekli tekrarla

mesafe i yuvarla Mesafe Ölçer >> Trig: 3 Echo: 4

eğer mesafe > 2 ve mesafe < 30 ise

değilse

İleri Git % 40 Güçte

Robotumuz ileri gittiği sırada Ön ledi açsın ve arka ledi kapatsın. Böylelikle mesafe sensöründen okunan değer motorlarda bir sorun olduğunda hangi aralıktta olduğunu anlayabiliriz. Bunun için aşağıdaki görseldeki gibi “Pinler” kategorisindeki “A0 pinini 180 yap”, “A1 pinini 0 yap” bloklarını değilse komut alanına sürükleyip bırakalım.

Engelden kaçan Survivor Robot

Pinler

- Sayısal D2 pinini YÜKSEK
- PWM D3 pinini 0 olarak ayarla
- A0 pinini 0 olarak ayarla
- Sayısal D2 pininin durumu
- Analog A0 pininin değeri

Haberleşim

Motorlar

Sensörler

- Olaylar
- Kontrol
- İşlemler
- Değişkenler
- Bloktaım

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

sürekli tekrarla

mesafe i yuvarla Mesafe Ölçer >> Trig: 3 Echo: 4

eğer mesafe > 2 ve mesafe < 30 ise

değilse

R.E.X İleri Git % 40 Güçte

A0 pinini 180 olarak ayarla

A1 pinini 0 olarak ayarla

Kodlarımızın son bölümünde ise engel algılandığında robotumuzun ne yapacağını kodlayacağız. Eğer ise değilse bloğumuzun eğer ise komut alanına gerekli kodlarımızı ekleyeceğiz. Öncelikle rastgele bir yöne yani sağa veya sola dönüş işlemi belirleyeceğiz. Bunun içinde “taraf” isimli bir değişken oluşturalım. Aşağıdaki görseldeki gibi değişkeni oluşturuyoruz.

Bir Değişken Oluştur

Pinler

- mesafe

Haberleşim

mesafe i 0 yap

mesafe i 1 ile değ

Motorlar

mesafe değişkenini o

Sensörler

mesafe değişkenini o

Bir Liste Oluştur

Olaylar

Kontrol

İşlemler

Değişkenler

Yeni Değişken

Yeni değişken ismi:

taraf

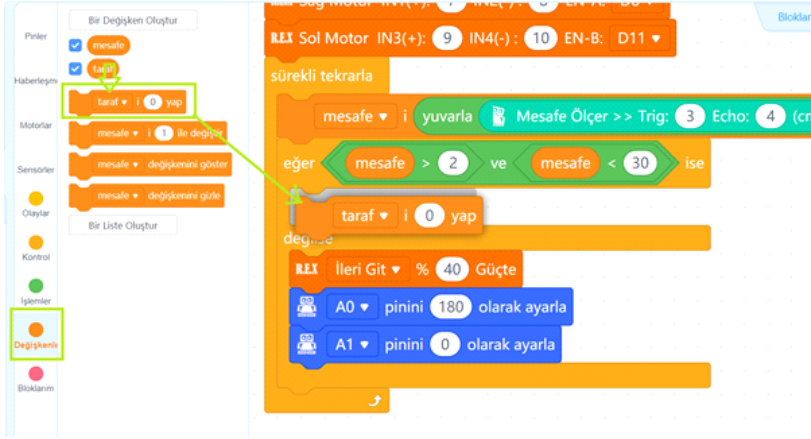
Bütün kuklalar için

Sadece bu kukla için

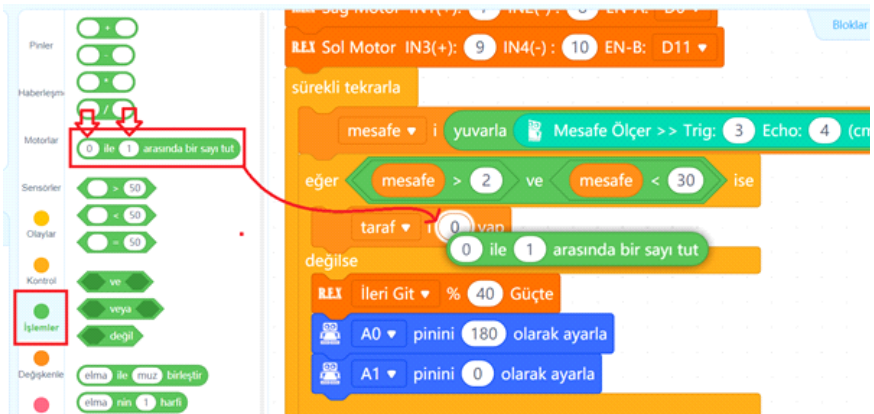
İptal Tamam

Engelden kaçan Survivor Robot

taraf değişkeninin değerini engel her algılandığında (mesafe değeri 2 ile 30 arasında her ölçüldüğünde) rastgele olarak 0 ya da 1 olarak belirleyeceğiz. taraf değişkeni 0 ise robotumuzu sağa doğru 1.2 sn 0 değilse sola doğru 1.2 sn dönmesini sağlayacağız. taraf değişkenini rastgele 0 ya da 1 değeri alabilmesi için öncelikle aşağıdaki gibi taraf'ı 0 yap bloğunu eğer ise komut alanına sürükleyip bırakalım.



taraf değişkeninin rastgele değer alabilmesi için "işlemler" kategorisindeki "1 ile 10 arasında bir sayı tut" bloğunu 0 ile 1 arasında olacak şekilde değiştirip, "taraf'ı 0 yap" bloğunun değer alanına aşağıdaki gibi sürükleyip bırakalım.



Dönme işlemini gerçekleştirmek için taraf değişkeninin değerinin 0 veya 1 olma durumunu sorgulamamız gerekiyor. taraf değişkeni 0 ise sağa dönme komutu 0 değilse (1 ise) sola dönme komutu vereceğiz. Bunun için taraf değişkeninin değerinin rastgele ayarlandığı bloğun hemen altına görseldeki gibi "kontrol" kategorisindeki "eğer ise değilse" bloğunu yerleştirelim.

Engelden kaçan Survivor Robot

The screenshot shows the Scratch environment. On the left, the 'Kontrol' category is highlighted in the sidebar. The main workspace contains a code block for the 'Sol Motor' (Left Motor) with the following structure:

- Block: REX Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11
- Block: sürekli tekrarla (forever loop)
- Block: mesafe Ölçer >> Trig: 3 Echo: 4 (cm)
- Block: eğer (if) with conditions: mesafe > 2 ve mesafe < 30
- Block: taraf (direction) i 0 ile 1 arasında bir sayı tut yap
- Block: eğer ise (if-then)
- Block: değilse (else)
- Block: değilse (else)
- Block: REX İleri Git % 40 Güçte
- Block: A0 pinini 180 olarak ayarla

Şimdi robotumuzun dönme işleminin koşulunu yerleştirmeliyiz. Eğer ise değilse bloğunun koşul alanına taraf değişkeni 0 a eşitse koşulunu ifade edebilmek için aşağıdaki görseldeki gibi "işlemler" kategorisindeki "eşittir" operatör bloğunu sürükleyip bırakalım.

The screenshot shows the Scratch environment. On the left, the 'İşlemler' category is highlighted in the sidebar. The main workspace contains a code block for the 'Sol Motor' (Left Motor) with the following structure:

- Block: REX Sol Motor IN3(+): 9 IN4(-): 10 EN-B: D11
- Block: sürekli tekrarla (forever loop)
- Block: mesafe Ölçer >> Trig: 3 Echo: 4 (cm)
- Block: eğer (if) with conditions: mesafe > 2 ve mesafe < 30
- Block: taraf (direction) i 0 ile 1 arasında bir sayı tut yap
- Block: eğer ise (if-then) with condition: 0 = 0
- Block: değilse (else)
- Block: değilse (else)
- Block: REX İleri Git % 40 Güçte

Değişkenler kategorisinden taraf değişkeni değer bloğunun eşittir operatör bloğunun değişken alanına sürükleyip bırakalım.

Engelden kaçan Survivor Robot

The image shows the Scratch environment. On the left, the 'Değişkenler' (Variables) category is selected, showing a variable named 'mesafe' (distance) and another named 'taraf' (direction). The 'Motorlar' (Motors) category is also visible, showing a 'R.E.X Sol Motor' block with 'IN3(+): 9' and 'IN4(-): 10' pins. The 'Eğer ise değilse' (If-then-else) block is also visible, with 'taraf' variable being compared to 0.

Robotumuz, taraf değişkeni değeri 0 ise 1.2 sn boyunca sağa dönmesi, değilse (taraf değişkeni değeri 1 ise) 1.2 sn boyunca sola dönme işlemini gerçekleştirmesi için Motorlar kategorisindeki görseldeki gibi blokları sürükleyip “Eğer ise değilse” bloğundaki ilgili yerlere yerleştirelim.

The image shows the Scratch environment. On the left, the 'Motorlar' (Motors) category is selected, showing a 'R.E.X Sol Motor' block with 'IN3(+): 9' and 'IN4(-): 10' pins. The 'Eğer ise değilse' (If-then-else) block is also visible, with 'taraf' variable being compared to 0. The 'R.E.X Sol Motor' block is set to 'İleri Git' (Move Forward) with a power of 40% and a duration of 1.2 seconds.

Engelden kaçan Survivor Robot

Son olarak dönme işlemi gerçekleştirirken Arka ledin açılması ve ön ledin kapanmasını sağlayalım. Bunun için Pinler kategorisindeki A0 pinini 0 yap bloğunu kullanarak Eğer ise değilse bloğunun hemen öncesine sürükleyip bırakalım.

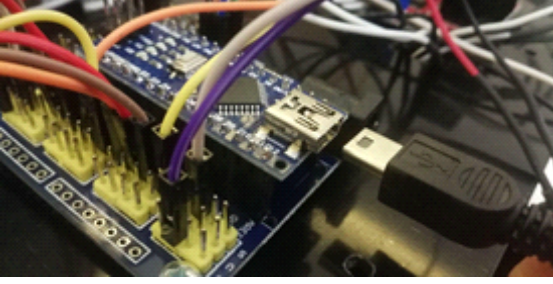
A0 ve A1 pinleri analog pinlerdir aynı zamanda PWM gibi 0-255 değer aralığında pinlere gerilim gönderebilirler. Çoğu led 3V-2V civarında maksimum parlaklığa eriştiği için A0 ve A1 pinlerine 5V anlamına gelen 255 değeri vermektense, 180 değeri ile 3V civarında gerilim verilmesini sağlayarak ledlerin ömrünü kısaltmamış olduk.



Kodlarımızın bitmiş hali aşağıdaki görseldeki gibi olmalı.

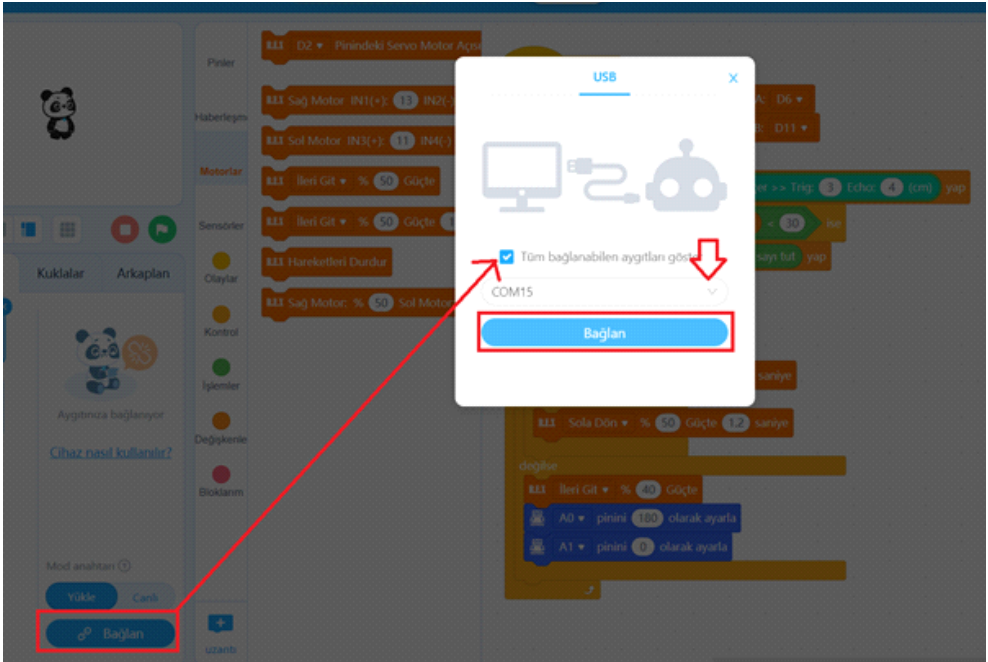


Engelden kaçan Survivor Robot

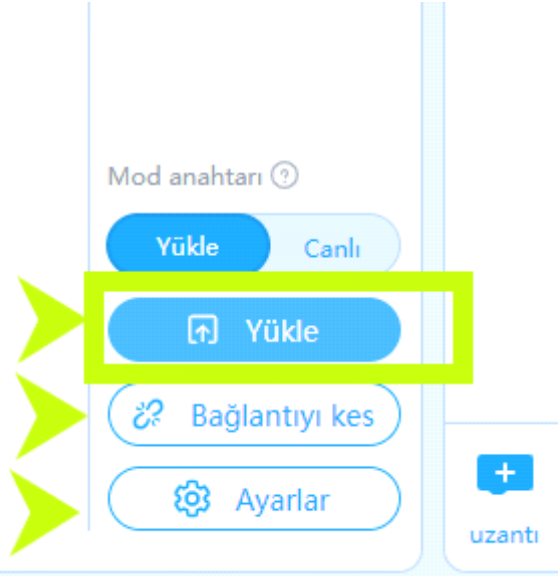


Hazırladığımız kodları artık robotumuza yükleyebiliriz. Usb kablosunun bir ucunu Arduino nano'ya diğer ucunu ise bilgisayarımıza bağlıyoruz.

Yükleme modu aktifken bağlan butonuna tıklayarak tüm bağlanılabilir aygıtları göster seçeneğini işaretliyoruz. mBlock yazılımı otomatik olarak CH340 çipli kartımızın bağlı olduğu COM port numarasını oraya getirecektir. Bağlantı sağlanamaz ise açılır listeden diğer COM port numaralarını seçebilirsiniz.



Bağlantı sağlandığında, "bağlantıyı kes", "ayarlar" ve "Yükle" butonları aktif olur. Yükle butonuna basarak kodu robotumuza yükleyelim.



Kodların Çalıştırılması

Kod yüklenir yüklenmez robotumuz sensörden gelen verileri okuyarak hareket etmeye başlayacaktır. Kodları yükledikten sonra usb kablosunu çıkarıp sensör shield ve motor sürücüyü güç sağlayan pilin kablolarını takarak ya da varsa anahtarı açarak robotun kodları işletmesini sağlayabilirsiniz. Pil gücüne göre robotunuzun dönüş sürelerini ileri gidiş hızını ve dönüş hızlarını ayarlayarak istediğiniz hassasiyeti sağlayabilirsiniz.

Olası Problemler ve Çözüm Yolları

Robotunuz sadece ileri geri gidiyor sağa sola dönmüyorsa ya da tam tersini yapıyorsa, IN1,IN2,IN3 ve IN4 pinlerinin yerlerini shield üzerinde ve motor sürücü üzerinden mutlaka kontrol edin.Yerleri karışmış olabilir.

Robotunuz İleri kodunu geri gidiyor, sağa dön kodunda sola dönüyorsa; Motor sürücü üzerindeki OUT1 ve OUT 3 ün motorların (+) kutbuna, OUT 2 ve OUT 4 ün motorların (-) kutbuna bağlı olmasını sağlayın. Ayrıca Robotun sağındaki motorların sürücü üzerinde OUT1 ve OUT2 ye solundaki motorların ise OUT3 ve OUT4 e bağladığınızdan emin olun.

Robotunuz engelle karşılaştığında dahile arka ledi yanmıyor, hep ileri gitmeye devam ediyorsa; HC-SR04 (mesafe sensörü) modülünün Trig ve Echo pinlerine bağlı jumper kabloların daha sağlam durmasını sağlayın. Sensörün duruş açısı itibarıyla bu iki pindeki jumper kablolardaki çok küçük bir hareketlilik sensörün 0(sıfır) değeri okumasına sebep olmakta o da engel algılama sisteminin çalışmamasına sebep olmaktadır.

Kılavuzun ieresinde bulunan tm kodlara,
alt tarafta bulunan QR kodu akıllı
cihazınıza okutarak, ya da kısa linki
tarayıcınızda aratarak projenin rnek
kodlarına erişebilirsiniz.



<http://rbt.ist/superstar>

 YouTube



[youtube.com/robotistan](https://www.youtube.com/robotistan)

FORUM



forum.robotistan.com

BLOG



maker.robotistan.com

Robotistan Elektronik Ticaret AŞ

Mustafa Kemal AVCI (İerik) - Fadıl PALA - Mehmet AKALI (Editr) - (Mehmet Nasır KARAER (Grafik)
info@robotistan.com - www.robotistan.com
Phone: 0850 766 0 425