

REX8



GitHub
rbt.ist/rexgithub



REX DOCS
rbt.ist/rexrdt



REX'i Keşfet
rbt.ist/rex8in1

BALANCEBOT KURULUM KILAVUZU



BalanceBot

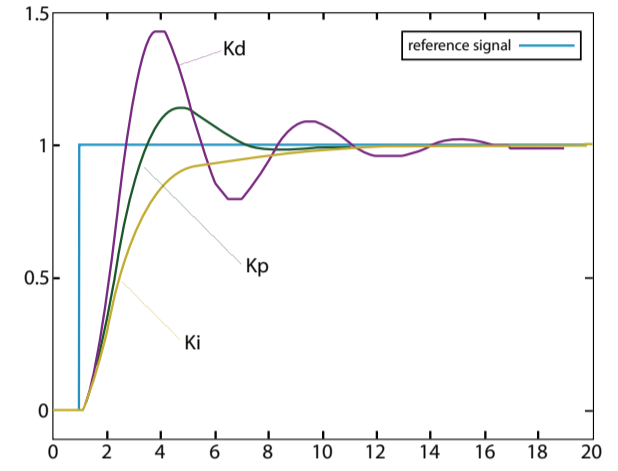
BalanceBot, REX kartı üzerinde bulunan MPU6050 sensörü sayesinde değişen çevresel faktörlere karşı dengede kalabilen REX robotudur.

BalanceBot Nasıl Dengede Kalır?

BalanceBot'un dengede kalması için çeşitli algoritmalar kullanılabilir. Biz BalanceBot'u dengede tutabilmek için PID algoritmasını kullanacağız.

PID (Proportional, Integral, Derivative) Algoritması Nasıl Çalışır?

Geri besleme ile giriş sinyaline gelen verinin, giriş sinyali ile olan farkı bulunur. Bu fark hatayı oluşturur. Hata sinyali PID denetleyicisine gönderilir ve hata sinyaline üç farklı katsayı ile üç farklı formül uygulanır. Daha sonra tekrar çıkış sinyaline gönderilir. Bu süreç, hata minimuma indirilene kadar bir döngü içerisine alınır.

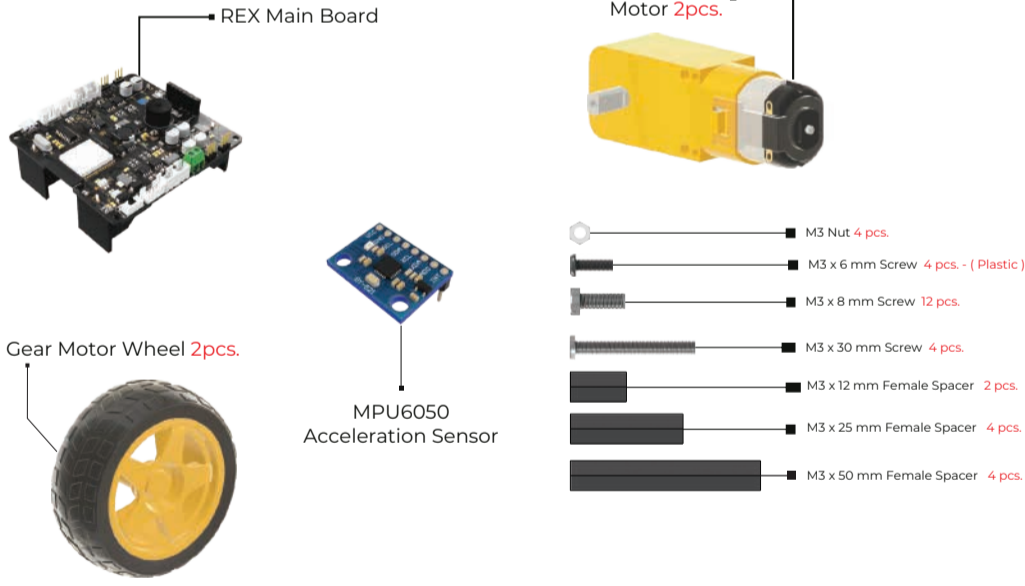


REX kartı üzerinde bulunan MPU6050 (gyro/ivme) sensörü ile çevresel faktörler hesaplanarak PID'ye gönderilir. PID, BalanceBot'un dengesini sağlamak için gerekli işlemleri gerçekleştirerek çıkış sinyali üretir.

02

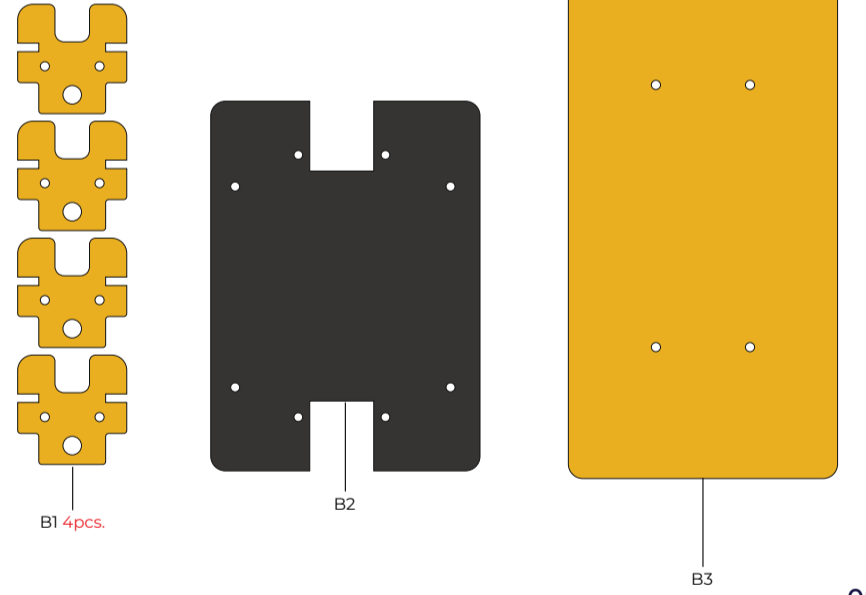
03

BalanceBot için Gerekli Komponentler

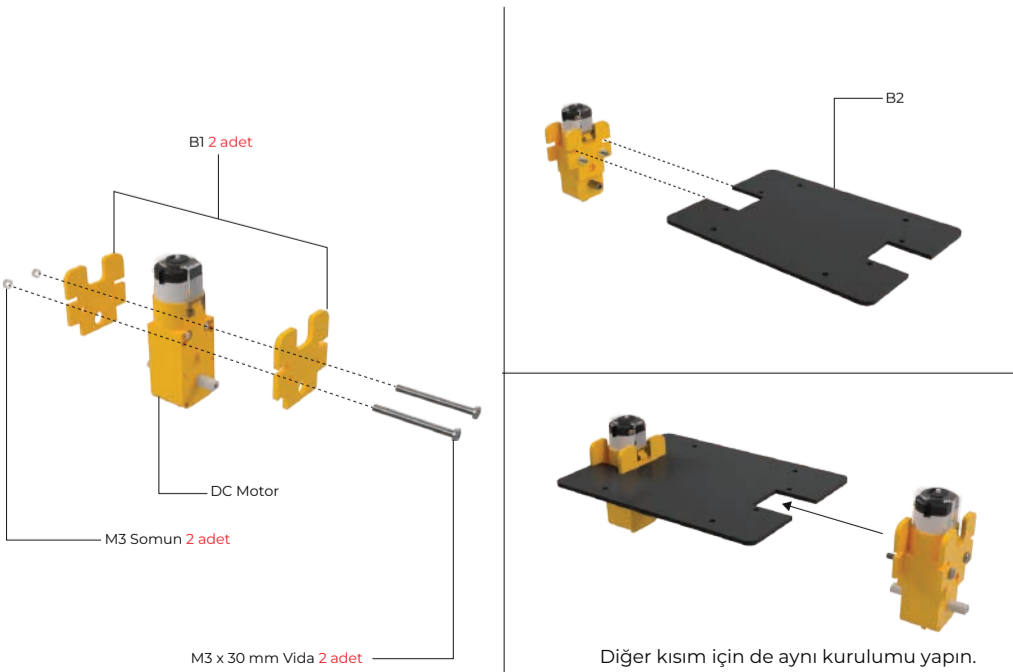


04

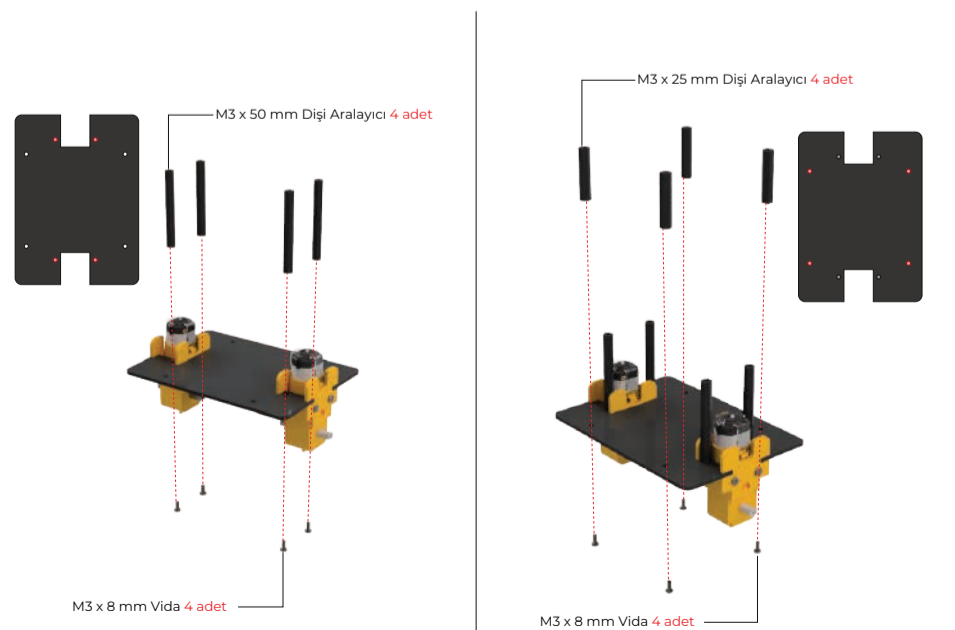
Pleksi Parçalar



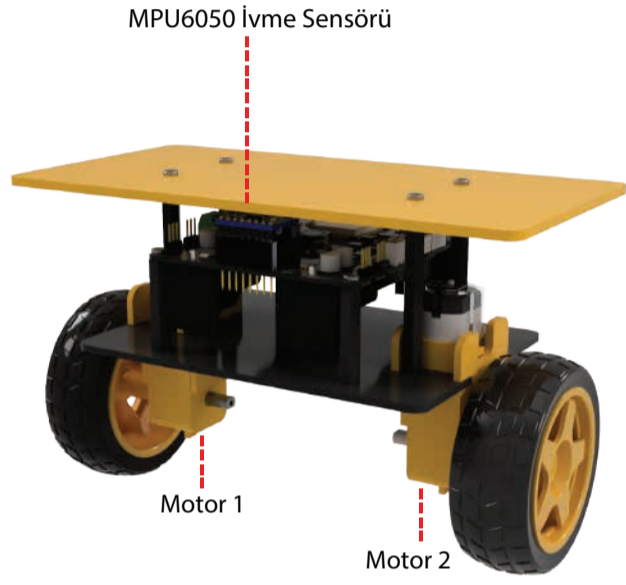
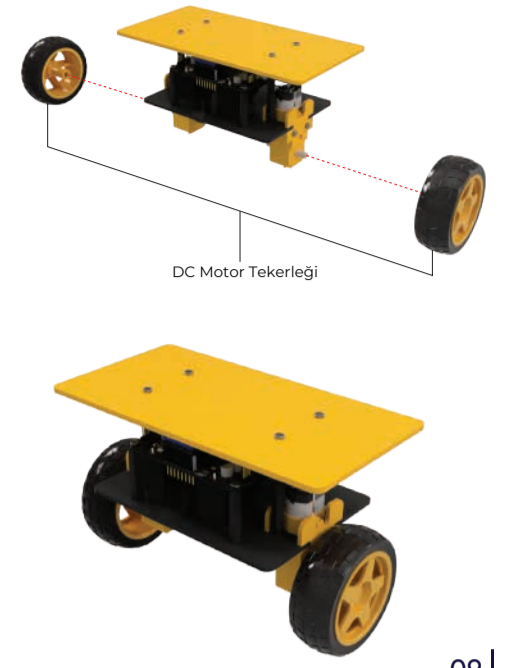
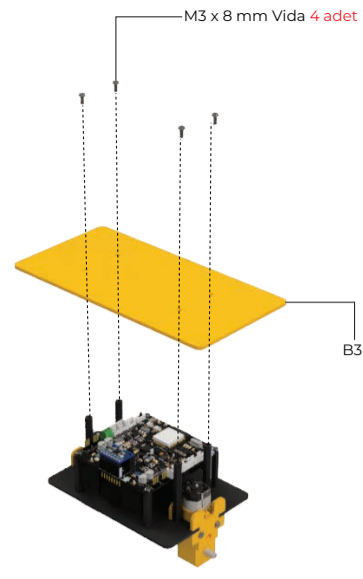
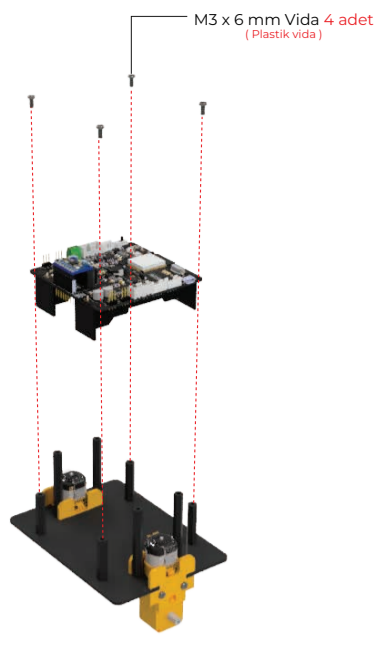
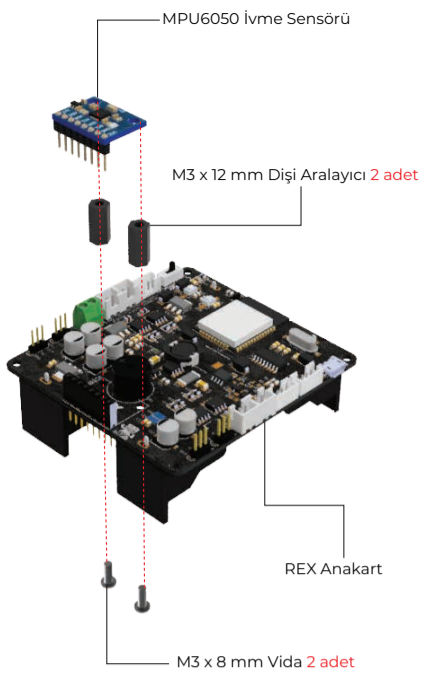
05



06

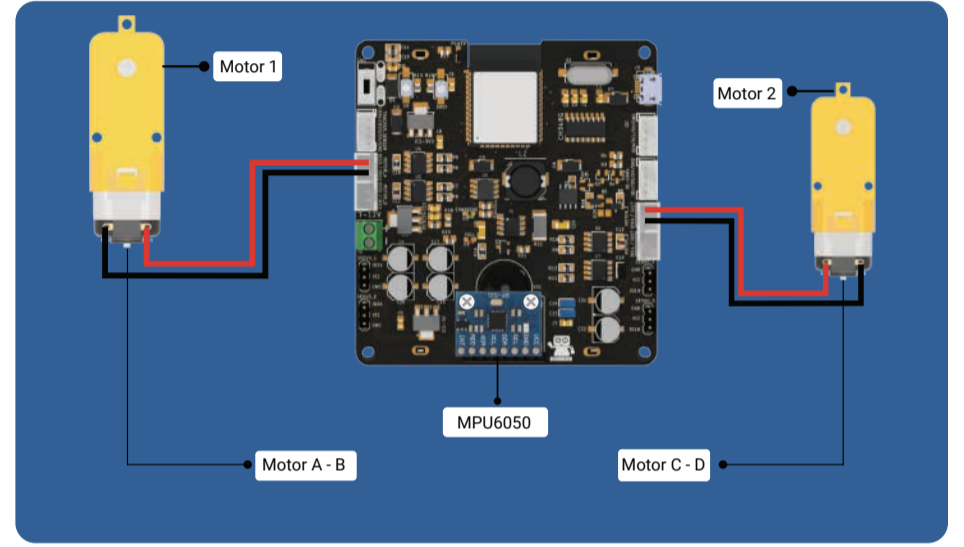


07



Devre Şeması

Pleksi parçaları kurduktan sonra, devre kurulumunu aşağıdaki şemada gösterildiği şekilde yapabilirsiniz.



Arduino Kodu

```

1 //***REX Mini Balance Bot***
2 //Check the web site for Robots http://rxt-rbt.meathedevs.io/en/latest/
3
4 #include "I2Cdev.h"
5 #include "PID_v1.h"
6 #include "MPU6050_6Axis_MotionApps20.h"
7 #include "Wire.h"
8
9 #define INTERRUPT_PIN 13
10
11 #define Motor_A1 16
12 #define Motor_A2 17
13 #define Motor_C1 23
14 #define Motor_C2 15
15
16 MPU6050 mpu;
17
18 bool dmpReady = false; // set true if DMP init was successful
19
20 uint8_t mpuIntStatus; // holds actual interrupt status byte from MPU
21 uint8_t devStatus; // return status after each device operation (0 = success, != 0 = error)
22 uint16_t packetSize; // expected DMP packet size (default is 42 bytes)
23 uint16_t fifoCount; // count of all bytes currently in FIFO
24 uint8_t fifoBuffer[64]; // FIFO storage buffer
25
26
27 // orientation/motion vars
28 Quaternion q; // [w, x, y, z] quaternion container
29 VectorFloat gravity; // [x, y, z] gravity vector
30 float ypr[3]; // [yaw, pitch, roll] yaw/pitch/roll container and gravity vector
31
32 //.....set following 4 values for your robot....
33 double setpoint = 178; //set the value when the bot is perpendicular to ground using serial monitor.(input value)
34 double kp = 10; //set this value first
35 double kd = 0.20; //set this value second
36 double ki = 250; //finally set this value
37
38
39 double input, output;
40 PID pid(input, &output, &setpoint, kp, ki, kd, DIRECT);
41
42
43 volatile bool mpuInterrupt = false; // indicates whether MPU interrupt pin has gone high

```

<http://rbt.ist/balancebotcode>



Kodun tamamına ve gerekli olan kütüphanelere gitmek için QR kodu okutunuz.