

**KARADENİZ TEKNİK ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**SU TERAZİSİ, ŞAKÜL, GÖNYE**

**TASARIM PROJESİ**

**Büşra KARABELA**

**2015-2016 GÜZ DÖNEMİ**

**KARADENİZ TEKNİK ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**SU TERAZİSİ, ŞAKÜL, GÖNYE**

**TASARIM PROJESİ**

**Büşra KARABELA**

**Bu projenin teslim edilmesi ve sunulması tarafımdan uygundur.**

**Danışman : Prof. Dr. Cemal KÖSE**

**2015-2016 GÜZ DÖNEMİ**



## **IEEE Etik Kuralları** **IEEE Code of Ethics**



Mesleğime karşı şahsi sorumluluğumu kabul ederek, hizmet ettiğim toplumlara ve üyelerine en yüksek etik ve mesleki davranışta bulunmaya söz verdiğimi ve aşağıdaki etik kurallarını kabul ettiğimi ifade ederim:

1. Kamu güvenliği, sağlığı ve refahı ile uyumlu kararlar vermenin sorumluluğunu kabul etmek ve kamu veya çevreyi tehdit edebilecek faktörleri derhal açıklamak;
2. Mümkün olabilecek çıkar çatışması, ister gerçekten var olması isterse sadece algı olması, durumlarından kaçınmak. Çıkar çatışması olması durumunda, etkilenen taraflara durumu bildirmek;
3. Mevcut verilere dayalı tahminlerde ve fikir beyan etmelerde gerçekçi ve dürüst olmak;
4. Her türlü rüşveti reddetmek;
5. Mütenasip uygulamalarını ve muhtemel sonuçlarını gözeterek teknoloji anlayışını geliştirmek;
6. Teknik yeterliliklerimizi sürdürmek ve geliştirmek, yeterli eğitim veya tecrübe olması veya işin zorluk sınırları ifade edilmesi durumunda ancak başkaları için teknolojik sorumlulukları üstlenmek;
7. Teknik bir çalışma hakkında yansız bir eleştiri için uğraşmak, eleştiriye kabul etmek ve eleştiriye yapmak; hatları kabul etmek ve düzeltmek; diğer katkı sunanların emeklerini ifade etmek;
8. Bütün kişilere adilane davranmak; ırk, din, cinsiyet, yaş, milliyet, cinsi tercih, cinsiyet kimliği, veya cinsiyet ifadesi üzerinden ayrımcılık yapma durumuna girişmemek;
9. Yanlış veya kötü amaçlı eylemler sonucu kimsenin yaralanması, mülklerinin zarar görmesi, itibarlarının veya istihdamlarının zedelenmesi durumlarının oluşmasından kaçınmak;
10. Meslektaşlara ve yardımcı personele mesleki gelişimlerinde yardımcı olmak ve onları desteklemek.

IEEE Yönetim Kurulu tarafından Ağustos 1990'da onaylanmıştır.

## ÖNSÖZ

Bu kılavuzun ilk taslaklarının hazırlanmasında emeđi geçenlere, kılavuzun son halini almasında yol gösterici olan kıymetli hocam Sayın Cemal Köse'ye Őukranlarımı sunmak istiyorum. Ayrıca bu çalıřmayı destekleyen Karadeniz Teknik Üniversitesi Rektörlüğü'ne Mühendislik Fakültesi Dekanlığına ve Bilgisayar Mühendisliđi Bölüm Başkanlığına içten teşekkürlerimi sunarım.

Her Őeyden öte, eđitimim süresince bana her konuda tam destek verem aileme ve bana hayatlarıyla örnek olan tüm hocalarıma saygı ve sevgilerimi sunarım.

Büşra KARABELA

Trabzon, 2015

## İçindekiler

ÖNSÖZ.....	III
İÇİNDEKİLER.....	IV
ÖZET.....	V
SEMBOLLER VE KISALTMALAR.....	VI
1.GİRİŞ.....	1
2.MPU6050 SENSÖRÜ.....	2
2.1.İVMEÖLÇER – ACCELEROMETER.....	2
2.2.JİROSKOP - GYROSCOPE.....	5
2.3.ATALETSEL ÖLÇÜ BİRİMİ - IMU.....	7
2.4.DÖRDEY.....	8
3.ARDUINO VE MPU6050.....	9
4.PC UYGULAMASI.....	9
4.1.KÜPÜN EKTRAN GÖRÜNTÜSÜ.....	10
4.2.KONSOL EKRANI.....	11
5.KAYNAKLAR.....	12

## ÖZET

Bu projede yapı işleri, inşaat ve marangozlukta oldukça yaygın olarak kullanılan su terazisi, şakul ve gönye tek bir projede elektronik olarak gerçekleştirilmiştir. Bu üç malzemede farklı eksenlerde açı ölçtüğü için projeye özetle üç eksenli açıölçer de denilebilir. Bunun için üç eksenli açı ölçebilen MPU6050 sensörü ve bilgisayarla bağlantıyı sağlamak için Arduino UNO kullanılmıştır.

## SEMBOLLER VE KISALTMALAR

### KISALTMA

### AÇIKLAMA

ACC

Accelerometer

GYR

Gyroscope

DMP

Digital Motion Process

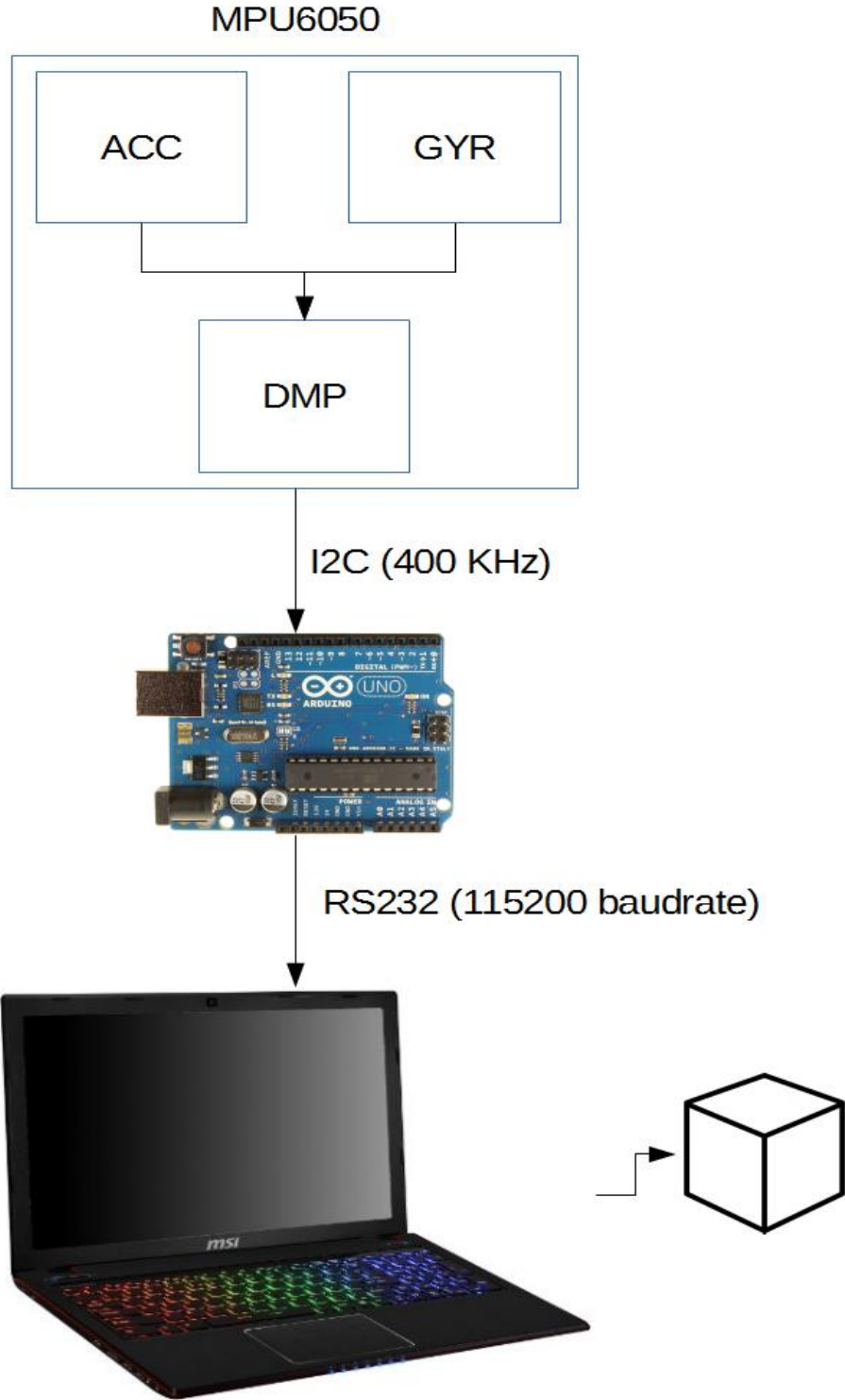
IMU

Inertial Measurement Unit

DOF

Degrees of Freedom

# 1. GİRİŞ





## 2. MPU6050 SENSÖRÜ

MPU6050 sensörü, çeşitli hobi ve robotik projelerinde özellikle denge robotlarında ve hava araçlarında sıklıkla kullanılan üzerinde 3 eksenli bir gyro ve 3 eksenli bir ivmeölçer bulunduran 6 eksenli bir IMU sensör kartıdır. Kart 12C protokolünü desteklediğinden üzerindeki pinler standart 12C pinleridir ve Arduino'nun standart 12C pinlerine takılmalıdır.

Kart üzerinde voltaj regülatörü bulunduğundan 3 ile 5V arası bir besleme voltajı ile çalıştırılabilir. Her ekseninde 16 bitlik bir çözünürlükte çıkış verebilmektedir. Pinler arası boşluk standart olarak ayarlandığı için breadboard veya farklı devre kartlarında kullanılabilir.

### 2.1. İVMEÖLÇER - ACCELEROMETER

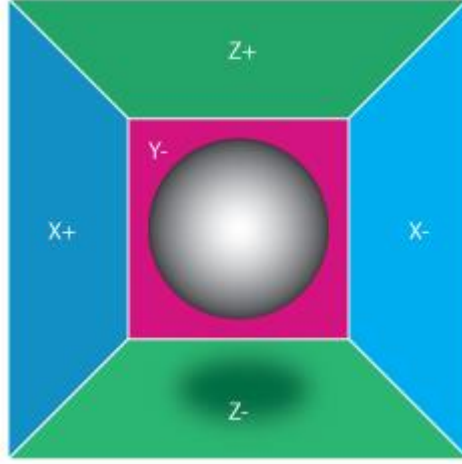
Accelerometerlar üzerlerine düşen statik (yer çekimi) ve dinamik (aniden hızlanma veya durma) ivmeyi ölçmektedirler. Sensörden aldığımız değer  $m/s^2$  veya yerçekimi türünden ifade edilebilir. Uygulamalarda genelde yerçekimi türünden ifade edilmektedir. Eğer uzayda veya herhangi bir çekim alanının kapsamında değilseniz sensör üzerine  $1g$ 'lık bir çekim alanı etki etmektedir. Buda  $9.8 m/s^2$ 'dir ve bulunulan noktaya göre değişiklik gösterebilmektedir.

Sensör sürekli olarak yerçekimi etkisi altında kaldığından eğimölçer (yeni nesil akıllı cep telefonlarında kullanılmaktadır ve telefonun yatay ve dikey durumuna göre ekran değişmektedir) veya hareket algılayıcı (wii gibi ürünlerde elinizi salladığınızda oyundaki karakterlerde benzeri bir hareket yapar) olarak kullanılabilir. Ölçü skalası olarak  $\pm 1$ ,  $\pm 2$ ,  $\pm 4$ ... gibi ifade edilmektedir ve bir, iki, üç değerler ile ekseninde ölçüm yapabilen türleri vardır.

#### *Accelerometer Çalışması:*

İlk olarak uzayda olduğumuzu düşünelim. Yerçekimi ve ağırlığımız  $0$ 'dır. Önümüzde bir kutu ve kutunun ortasında da bir küre olduğunu düşünelim. Herhangi bir çekim etkisi olmadığından küre herhangi bir yüzeye temas etmeden hareketsiz bir şekilde durmaktadır.

Kürenin hareketini görebilmek için kutunu  $+Y$  yönünde kalan yüzeyini kesip atalım.



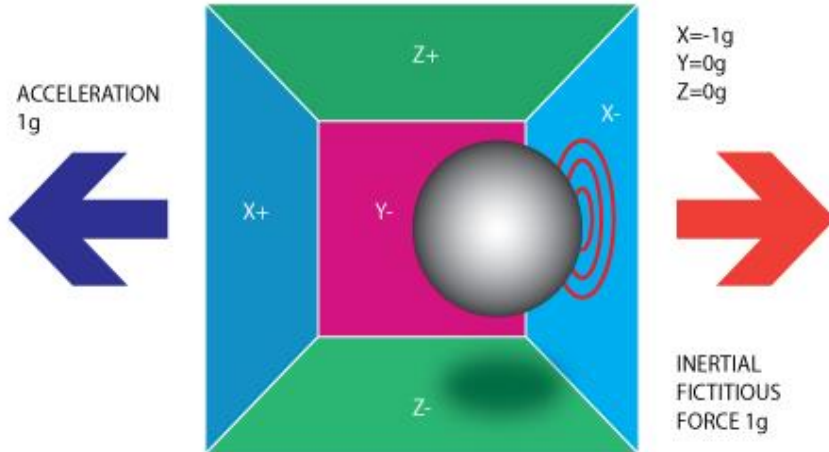
$$x = 0g$$

*still weightless state*

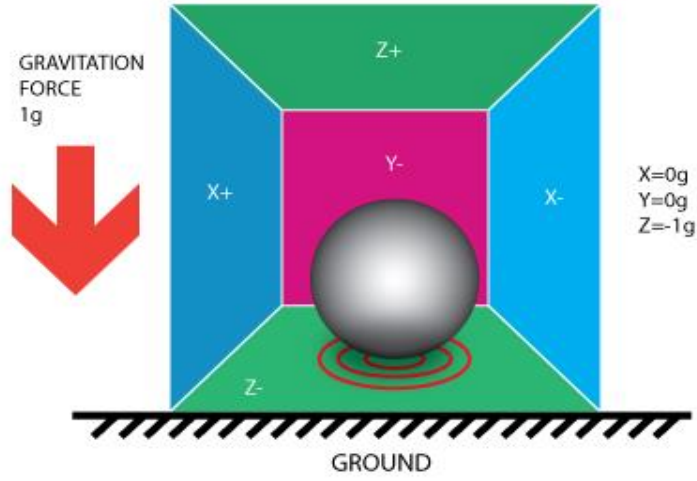
$$y = 0g$$

$$z = 0g$$

Kutuyu elimizde tutup +X yönünde 1g kuvveti ile hızlandırdığımızda küre kutunun –X yüzeyine eylemsizlikten dolayı 1g lik kuvvet uygulayacaktır.



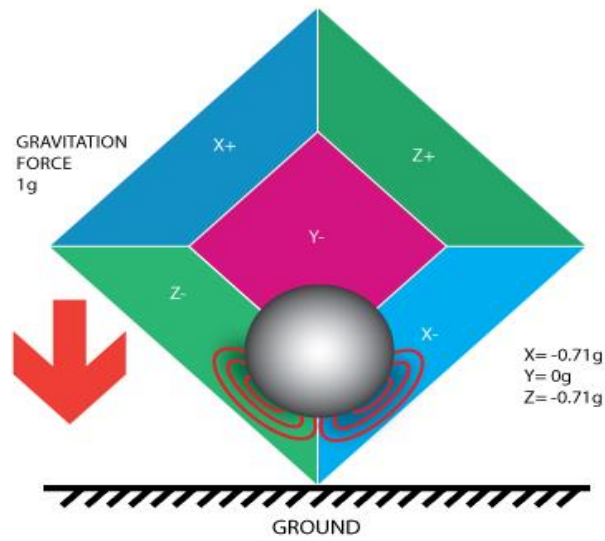
Kutuyu alıp dünyaya dönelim. Kutuyu yere koyduğumuzda dünyanın 1g'lik yerçekiminden dolayı küre –Z yüzeyine 1g kuvvet uygular.



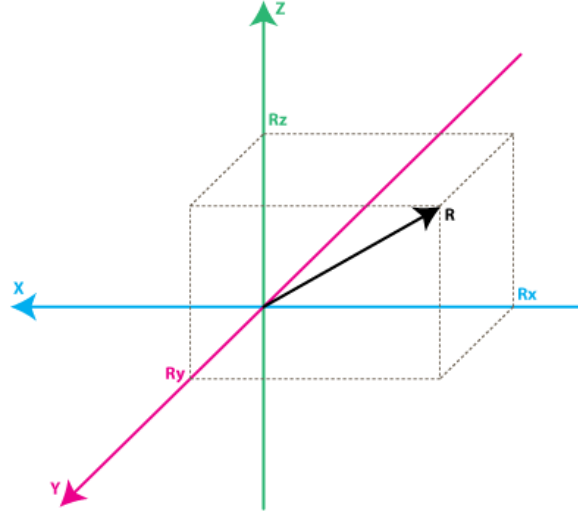
İvmeölçerlerde benzer şekilde çalışmaktadır. Yüzeyleri basınca veya manyetik alana tepki verecek şekilde yapılmaktadır ve ivmeölçerlerde bu tepkiyi ölçerek bize bir değer vermektedir.

Sensörün yer yüzü ile yaptığı açı değiştiğinde sensörün eksenlerine uygulanan kuvvet değişecektir ve bizde yeni değerleri okuyarak yeryüzü ile olan açıyı trigonometri ile hesaplarız.

Örneğin kutuyu 45 derece sağa çevirirsek bu durumda kürenin  $-X$  ve  $-Z$  yüzeylerine kök içinde  $\frac{1}{2}$ 'lik bir kuvvet uygulanır. O da  $0,707g$ 'ye eşittir.



Koordinat sistemi modeline gezersek;



Burada R vektörü ivmeölçerimiz üzerine düşen kuvvet vektörü olsun. Bu kuvvet demin anlatıldığı gibi yerçekimi ve sensörün hareketi sonucu yerçekimi kuvveti ile eylemsizlik kuvvetinin bileşkesi olabilir. R vektörünün 3 bileşeni vardır ve  $R = [R_x, R_y, R_z]$  şeklinde ifade edilir. Pisagor teoreminden R vektörünü bu bileşenlerden şu şekilde hesaplanır;

$$R^2 = R_x^2 + R_y^2 + R_z^2$$

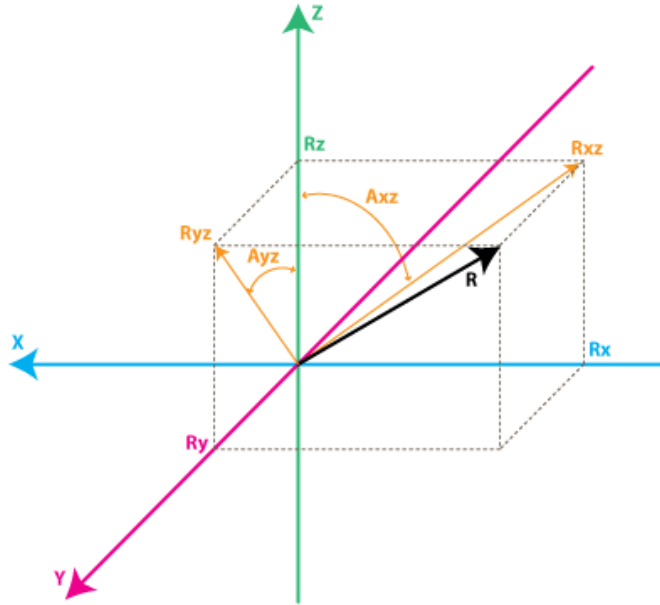
$R_x$ ,  $R_y$  ve  $R_z$  bileşenlerini bildiğimiz takdirde şekilden de görüldüğü gibi trigonometrik fonksiyonlar yardımı ile R vektörünün X ve Y eksenleri ile yaptığı açıları hesaplayıp sensörümüzün yeryüzüne göre konumunu bulabiliriz. R vektörünün bileşenlerini de bize ivmeölçer g kuvveti türünden vermektedir.

## 2.2. JİROSKOP - GYROSCOPE

Gyroscopes'lar basitçe bir tekerleğin ekseni etrafında hızlıca döndürülmesi sonucu ortaya çıkar. Tekerleğin etrafındaki çembere dik açıyla kenetlenmiş başka bir çember ve bu çemberlere dik açıyla tutturulmuş başka bir çember jiroskobu modeller. Jiroskobun öne çıkan

iki özelliği vardır. Yatay eksende dönmekte olan jiroskoba birden yatay eksen doğrultusunda bir kuvvet uyguladığımızda yatay eksen etrafında dönmek yerine eksen etrafında dönmeye başlar. Diğer bir özelliği ise jiroskobun dönmeye başladığı eksenin jiroskobun durduğu yüzeye açıyla oynatılırsa oynatılsın jiroskobun dönüş eksenini sabit kalır. Bu özelliğinden dolayı uyduların sürekli olarak dünyaya dönük kalması, uçaklarda ve çeşitli araçlarda yapay ufuk oluşturulması ve otopilot gibi uygulamalarda kullanılmaktadır.

Piyasada bütünleşmiş devre olarak satılan modelleri rahatlıkla bulunmaktadır. İvmeölçerlerde olduğu gibi bir, iki veya üç eksenle ölçüm yapabilen modelleri vardır ve saniyede dönüş hızı ölçümüne göre değerlendirilmektedirler. Bu hızların üzerindeki dönüşler sonucu sensör çıkışları anlamsız olabilmektedir.



İki eksenli (X,Y) gyro kullandığımızı düşünelim. Burada R vektörünün XZ uzayındaki izdüşümü  $R_{xz}$ , YZ uzayındaki izdüşümü  $R_{yz}$  vektörü ile ifade edilmektedir. Bu vektörleri Pisagor teoreminden

$$R_{xz}^2 = R_x^2 + R_z^2$$

$$R_{yz}^2 = R_y^2 + R_z^2 \text{ olarak hesaplayabiliriz.}$$

Vektörlerin Z eksenini ile yapmış olduğu açılar ise  $A_{xz}$  ve  $A_{yz}$ 'dir. Sistemi Y eksenini etrafında döndürdüğümüzde  $A_{xy}$  açısı, X eksenini etrafında döndürdüğümüzde ise  $A_{yz}$  açısı değişecektir. Gyroscopun dönüş hızını ölçtüğünü söylemiştik. Dönüş hızını zaman ile çarparsak dönüş açısını elde etmiş oluruz.  $t_0$  anındaki açımızın  $A_{xz0}$  olduğunu ve  $t_1$  anındaki açımızın  $A_{xz1}$  olduğunu düşünelim. O halde dönüş açımız  $(A_{xz1} - A_{xz0}) = \text{Rate}_{Axz} * (t_1 - t_0)$  ile ifade edilir.

Sensörden alacağımız değerleri dönüş hızına çevirmek içinse aşağıdaki formülleri kullanabiliriz.

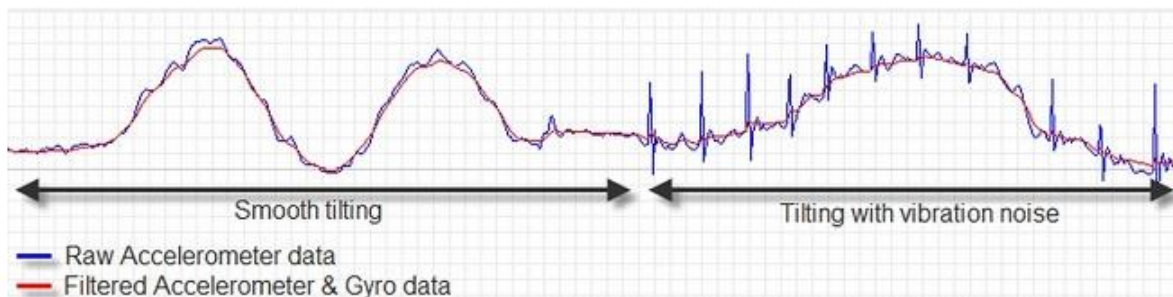
$$\text{Rate}_{Axz} = (\text{AdcGyroXZ} * V_{\text{ref}} / 1023 - V_{\text{zeroRate}}) / \text{Sensitivity}$$

$$\text{Rate}_{Ayz} = (\text{AdcGyroYZ} * V_{\text{ref}} / 1023 - V_{\text{zeroRate}}) / \text{Sensitivity}$$

### 2.3. Ataletsel Ölçüm Birimi - IMU (Inertial Measurement Unit)

Gyroscope ve accelerometer tek başlarına yeterince güvenli bilgi vermezler. Bu yüzden bu iki sensörü birleştirerek yönelim, hız, pozisyon gibi bilgileri tek bir üniteden alabiliriz. Bu üniteye IMU (Inertial Measurement Unit) denilmektedir. Serbestlik derecesi DOF (Degrees of Freedom) ile ifade edilmektedir. Örneğin 2 eksen gyro ve 3 eksen ivmeölçeriniz varsa 5 DOF IMU elde etmiş olursunuz.

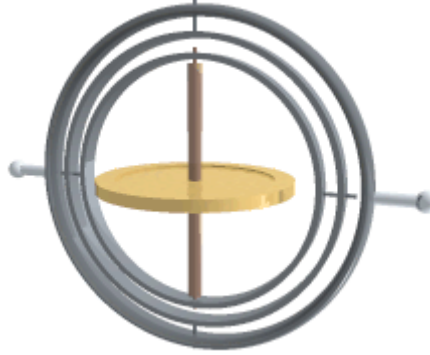
Gyro ve accelerometer bias drift adı verilen bir kayma yaparlar ve bundan dolayı hassas açı ölçümünde tek başlarına kullanılmazlar. Ayrıca accelerometerlar kuvvete karşı çok duyarlı olduğundan en ufak titreşimlerde çok yüksek gürültüler oluşturmaktadırlar. Gyrolar bu kuvvetlerden etkilenmezler. Şekilde görüldüğü gibi gyrolar ivmeölçer çıkışlarını filtreleyerek daha doğru ölçüm yapılmasını sağlarlar.



Filtreleme için çeşitli algoritmalar kullanılmaktadır. En yaygın olarak kullanılanlardan bir tanesi kalman filtresidir. Sistemin bir önceki çıkışları ile yeni ölçümlerden yeni çıkışları tahmin edecek şekilde çalışmaktadır.

## 2.4. DÖRDEY

Uçuş dinamiğinde hesaplama yapılırken dördeyler kullanılır. Bunun sebebi Euler açılarıyla hesaplama yaparken arctan fonksiyonunun tanımsız bölgesinden kaynaklanan gimbal lock sorunudur. Bu sorun eksenlerin 180 dereceden fazla dönmesinden kaynaklanan eksenlerin tanımsız bölgeye girip kilitlenmesidir.



Dördeyler ile hesaplamada kullanılan karmaşık sayılar ve ek bir değişken ile bu sorun aşılmıştır. Kullanıcının ekranda okunabilir değer gösterebilmek için bu değerler kolayca dereceye çevrilebilmektedir.

## 3. ARDUINO VE MPU5060

MPU6050 içerisinde dahili hesaplama birimi içermektedir. DMP olarak isimlendiren bu birimde sensörden alınan veriler yine MPU6050 yongası içerisinde matematiksel hesaplamalardan geçirilip sonuç değerleri hesaplanabilir. Bu bağlamda euler açıları ve quaternion açıları MPU6050 yongasında hesaplanır.

MPU6050 hareket sensörü Arduino ile I2C protokolü ile haberleşir. Hesaplanan roll, pitch, yaw değerleri birer kaydedici içerisinde tutulur. Bunlar Arduino ile I2C kanalı üzerinden 400Khz hızında alınır.

Arduino tarafından taranan veriler RS232 kanalı üzerinden bilgisayara aktarılmak üzere geçici bellekte tutulur. Mpu6050'ye yapılan her bir I2C sorgusu sonrasında bilgisayar programının okuyabileceği biçimde formatlanıp 115200 baudrate hızında serikanala basılır.

Formatlı serikanal girdisi :

\$q.w,q.x,q.y,q.z

#### 4. PC UYGULAMASI

Hesaplanan verilerin görselleştirilmesi için Visual Studio 2015 IDE'si ve C# programlama dili kullanılmıştır. Pc uygulaması 115200 baudrate ile periyodik serikanal taraması yaparak, Arduino'dan hesaplanan quaternion açılarını almaktadır.

Serikanaldan gelen formatlı verinin önce ayıklanması gerekmektedir. Serikanaldan veri sıralı gelmeyeceğinden başlangıç karakteri olan '\$' okununca veriye senkron olunur. Peşisıra gelen 4 quaternion değeri ',' karakteri ile ayrılmıştır.

Görselleştirme için ekrana Form\_3Dcuboid kütüphanesi kullanılarak küp çizdirilmiştir. Bu kütüphane sensör kutusunu temsil edecek şekilde çizdirilmiş ve kutu eksenleriyle aynı olacak şekilde kapak resimleriyle süslenmiştir.

Her bir serikanal sonrası yapılan ayıklama işlemi sonrasında quaternion açılarının küpü döndürebilmesi için rotasyon matrisine çevrilmesi gerekmektedir :

```
public static float[] ConvertToRotationMatrix()  
{ float num = 2f * q[0] * q[0] - 1f + 2f * q[1] * q[1];
```



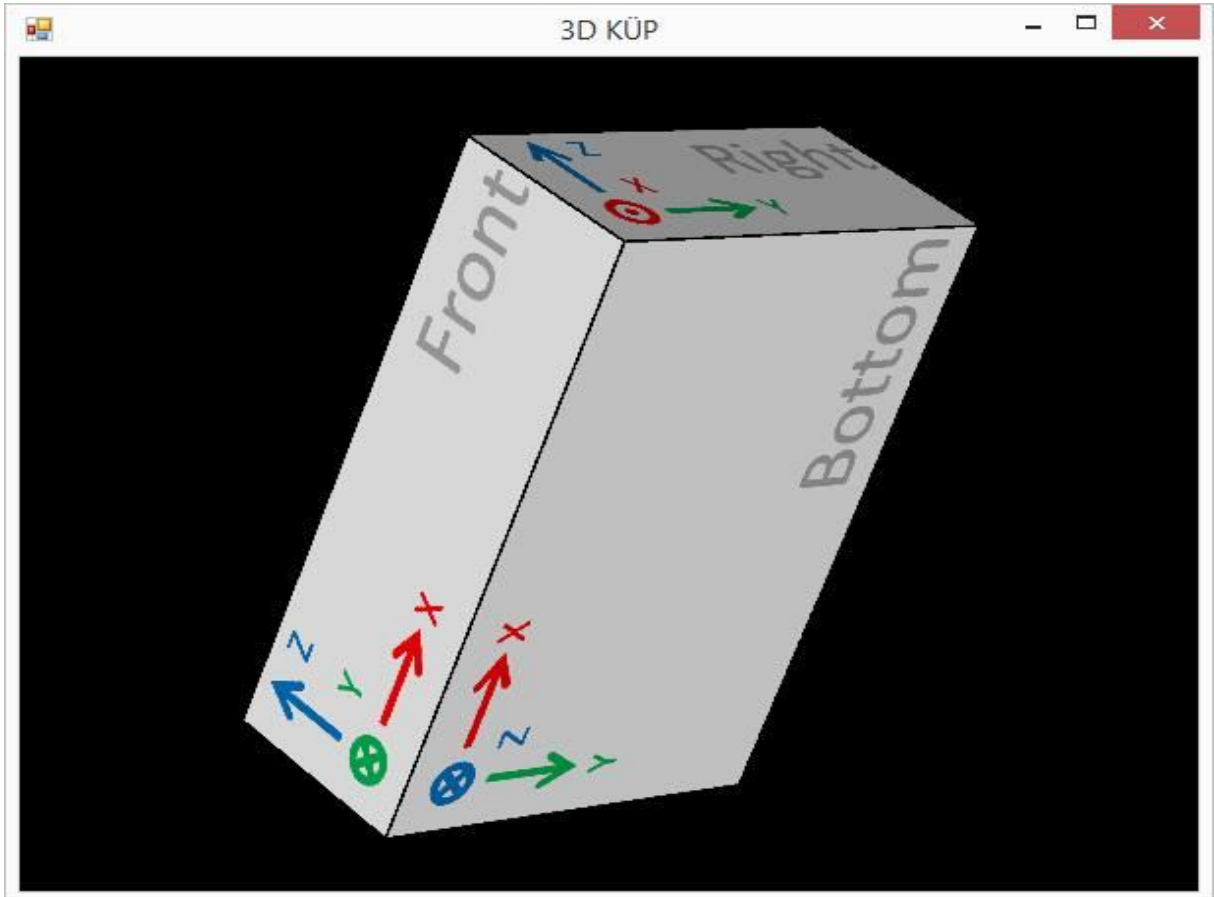
```

float num2 = 2f * (q[1] * q[2] + q[0] * q[3]);
float num3 = 2f * (q[1] * q[3] - q[0] * q[2]);
float num4 = 2f * (q[1] * q[2] - q[0] * q[3]);
float num5 = 2f * q[0] * q[0] - 1f + 2f * q[2] * q[2];
float num6 = 2f * (q[2] * q[3] + q[0] * q[1]);
float num7 = 2f * (q[1] * q[3] + q[0] * q[2]);
float num8 = 2f * (q[2] * q[3] - q[0] * q[1]);
float num9 = 2f * q[0] * q[0] - 1f + 2f * q[3] * q[3];
return new float[] { num, num2, num3,
                    num4, num5, num6,
                    num7, num8, num9 }; }

```

Herbir çevrim ayrı bir thread üzerinde yapıldığından görselleştirme UI threadini zorlamayacak, akıcı bir görüntü oluşacaktır.

#### 4.1. KÜPÜN EKCRAN GÖRÜNTÜSÜ



Kutunun 3 eksene karşılık düzen kenarlarının kendi eksenleriyle paralellliğini kontrol edebilmek için Euler açılarının hesaplanması gerekmektedir. Ekran yazdırılan bu veriler (roll, pitch, yaw), kendi eksenine paralel olduğunda 0 değerini gösterecektir.

```
public static float[] ConvertToEulerAngles()
{ float radians = (float)Math.Atan2((double)(2f * (q[2] * q[3]
- q[0] * q[1])), (double)(2f * q[0] * q[0] - 1f + 2f * q[3] *
q[3]));

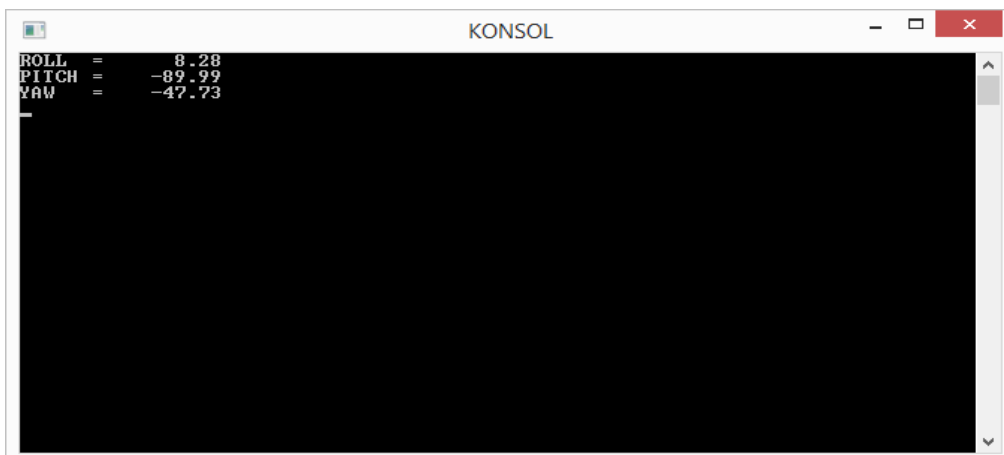
float radians2 = (float)(-(float)Math.Atan(2.0 *
(double)(q[1] * q[3] + q[0] * q[2]) / Math.Sqrt(1.0 -
Math.Pow(2.0 * (double)q[1] * (double)q[3] + 2.0 *
(double)q[0] * (double)q[2], 2.0))));

float radians3 = (float)Math.Atan2((double)(2f * (q[1] *
q[2] - q[0] * q[3])), (double)(2f * q[0] * q[0] - 1f + 2f *
q[1] * q[1]));

return new float[]
{ Rad2Deg(radians),
  Rad2Deg(radians2),
  Rad2Deg(radians3)
}
}
```

Hesaplanan değerler radyan cinsinde olup 3 boyutlu dizide geri döndürülmektedir. Ekran yazdırırken bu değerler dereceye çevrilir.

## 4.2. KONSOL EKRANI



```
KONSOL
ROLL = 8.28
PITCH = -89.99
YAW = -47.73
```

## 5. KAYNAKLAR

<http://www.barissamanci.net/Makale/26/accelerometer-gyroscope-imu-nedir/>

<https://www.arduino.cc/>

[https://www.cdiweb.com/datasheets/invensense/MPU-6050\\_DataSheet\\_V3%204.pdf](https://www.cdiweb.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf)

<https://www.arduino.cc/en/Main/ArduinoBoardUno>

<http://www.robotistan.com/mpu6050-6-eksen-ivme-ve-gyro-sensoru-6-dof-3-axis-accelerometer-and-gyros>

<https://github.com/jrowberg/i2cdevlib>

<http://playground.arduino.cc/Main/MPU-6050>