

**KARADENİZ TEKNİK ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
TASARIM ÇALIŞMASI**



**AKILLI BEŞİK**

**PROJE YÜRÜTÜCÜLERİ**

**GRUP: 20**

**OĞUZHAN FARIZ**

**YUSUF VANER**

**DANIŞMAN**

**DOÇ. DR. MURAT EKİNCİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ  
ANABİLİM DALI**

**2013 - 2014**

## ÖNSÖZ

Bebeklerin özellikle geceleri uykularından uyanması ebeveynler için uyku problemi oluşturmaktadır. Uyku düzensizlikleri anne ve babaların en rahatsız olduğu konulardan biridir. Bebeklerin uyku düzensizliği sebebiyle ailenin de uyku problemleri meydana gelmektedir.

Karadeniz Teknik Üniversitesi, Bilgisayar Mühendisliği Bölümün'de "Tasarım Projesi" dersi kapsamında uygun olarak " Akıllı Beşik " tasarımı yapılmıştır. Hazırladığımız "Akıllı Beşik " isimli projemizde bebeklerin beşiklerinde uyku esnasında herhangi bir sebeple uykudan kalktıklarında tekrar uykularına devam etmelerini sağlamaya çalıştık.

Lisans öğrenimi boyunca bizlere destek veren ve çalışmamız sırasında danışmanlığımızı yapan Sayın **Doç. Dr. Murat EKİNCİ** hocamıza teşekkür eder ve saygılarımızı sunarız.

Trabzon 2014

## İÇİNDEKİLER

ÖNSÖZ.....	2
İÇİNDEKİLER.....	3
ÖZET.....	4
1. SES.....	5
1.1.SES İŞLEME.....	5
1.2. ADC VE DAC.....	5
1.3.A/D ÇEVİRİCİNİN ÇALIŞMA PRENSİBİ.....	6
1.4.SESİN KULLANILDIĞI ALANLAR VE PROJELER.....	6
2. ARDUİNO UNO.....	7
2.1.ARDUİNO'NUN BİLGİSAYARA TANITILMASI.....	8
2.2.ARDUİNO GELİŞTİRME ORTAMININ TANITILMASI.....	10
2.3.GÜÇ.....	11
2.4.BELLEK.....	11
2.5. Giriş ve Çıkış.....	11
2.6. HABERLEŞME.....	12
2.7. PROGRAMLAMA.....	12
2.8. USB AŞIRI AKIM KORUMASI.....	13
2.9. FİZİKSEL ÖZELLİKLER.....	13
3.DC MOTOR.....	13
3.1.ENDÜVİ.....	13
3.2.GÖVDE.....	14
3.3.DC MOTORUN ÇALIŞMA ESASLARI.....	14
3.4. DC MOTORLARIN MEKANİK ÖZELLİKLERİ.....	15
3.5.DC MOTORUN HIZ DENETİMİ.....	15
4.AKUSTİK SES SENSÖRÜ (LM393) .....	16
5.UYGULAMA ADIMLARI.....	18
6.SONUÇ VE ÖNERİLER.....	22
KAYNAKLAR.....	23

## ÖZET

Araştırmalara göre bebekler anne karnında yaşarken duyduğu sesler ve annenin hareketleri nasılsa yeni doğan bebekler ilk 3 ayında alışık olduğu ortamı aramaktadırlar. Bu yüzden anne karnında ortalama yaşadığı 9 aylık sürede nasıl bir ortamda bulduysa doğduktan sonra da bu ortam sağlanırsa bebeklerin uyku düzensizlikleri minimuma ineceği kanısına varılmış.

Bu tasarım projesindeki amacımız bebeklerin uykularının bozulmaması ve uyku düzenlerini kazanmalarına yardımcı olmaktır. Bu amaçla donanım arayüzü için AtMega328 mikrodenetleyicisine sahip Arduino uno R3 kullanılmıştır. Ortamdaki sesi algılamak için kullanılan ses modülü ile ses şiddeti ölçülerek belirlenen eşik değerinin üzerinde ise Arduino üzerinden beşiğe bağlantılı olan dc motora akım verilerek bebeğin ses şiddetine göre hızlı veya yavaş olarak sallanması sağlanmaktadır.

## 1. SES

Ses, atmosferde canlıların işitme organları tarafından algılanabilen periyodik basınç değişimleridir. Fiziksel boyutta ses, hava katı sıvı veya gaz ortamlarda oluşan basit bir mekanik düzensizliktir. Bir maddedeki moleküllerin titreşmesi sonucunda oluşur.

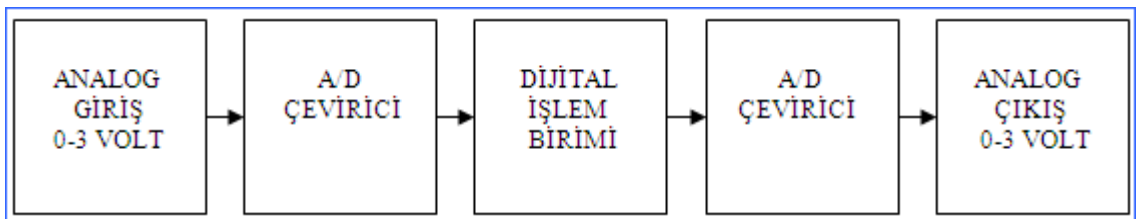
Ses sinyali, ortamdan alınmış olan ses girdisinin örneklenip quantalanarak elektronik ortamlarda iletilebilmesi veya saklanabilmesi için elektromanyetik enerjiye çevrilmiş halidir. Literatürde Audio Frequency (AF) olarak geçer. Ortamdan alınan ses girdileri analog türünde sinyallerden oluştuğu için bu sinyallerin işlenmesi analog/dijital çeviriciler yardımı ile yapılmaktadır.

### 1.1 SES İŞLEME

Ortamdan alınan analog ses sinyallerinin işlenmesi işlemi ADC (Analog Digital Converter) yardımı ile sağlanır. Bu işlem sonucunda bu veriler ile çeşitli sorgular yardımı ile başka bir mekanik düzeneğin kontrolü sağlanmaktadır.

### 1.2 ADC VE DAC

Dış dünyanın daha çok analog değerlerden oluşmasına karşılık , bilgi işleyen cihazlar (dijital sistemler, mikroişlemciler, bilgisayarlar) dijitaldir. Elde edilen bilginin tekrar dış dünyaya aktarılması da (örneğin görüntülenmesi) analog veya dijital biçimde olabilir. Bütün bu nedenlerle analog değerlerin dijitalle, dijital değerlerin de analog değerlere çevrilmesi gerekir.



**Şekil 1. ADC ve DAC İşlemleri Blok Diagramları**

Yukarıdaki şekilde analog bir değer dijital olarak çevrilip, işlendikten sonra tekrar analog değere çevrilmesi sürecinin blok diyagramı görülmektedir. Girişteki gerilim bir transduser yardımı ile elektriksel büyüklüğe çevrilmiş bir fiziksel büyüklüğü temsil etmektedir. Bu gerilim daha sonra Analog/Dijital Çevirici vasıtası ile dijital olarak çevrilir ve dijital olarak işlenir.

Daha sonra elde edilen sonuç Dijital/Analog Çevirici vasıtası ile tekrar analog bilgiye çevrilir ve çıkışa aktarılır. Çıkışta kullanılan eleman ise elektriksel büyüklüğü (gerilim) fiziksel büyüklüğe (ses, ısı, ağırlık vs) çevrilir. Örneğin hoparlör elektriksel büyüklüğü sese çeviren bir aygıttır.

### 1.3.A/D ÇEVİRİCİNİN ÇALIŞMA PRENSİBİ

A/D çeviricilerde giriş sinyali analog olarak değişen bir gerilimdir. Yukarıdaki blok diyagramda ki giriş gerilimi 0 volt ile 3 volt arasında değişmekte ve bu arada bulunan herhangi bir değeri alabilmektedir. Çıkışta ise girişteki analog değere göre bir binary sayı elde edilmektedir. Yandaki tabloda, çeşitli analog giriş gerilimlerine göre, devre çıkışında elde edilen binary sayılar görülmüyor

Analog Giriş (Volt)	Dijital Çıkış			
	D	C	B	A
0	0	0	0	0
0.2	0	0	0	1
0.4	0	0	1	0
0.6	0	0	1	1
0.8	0	1	0	0
1.0	0	1	0	1
1.2	0	1	1	0
1.4	0	1	1	1
1.6	1	0	0	0
1.8	1	0	0	1
2.0	1	0	1	0
2.2	1	0	1	1
2.4	1	1	0	0
2.6	1	1	0	1
2.8	1	1	1	0
3.0	1	1	1	1

### 1.4.SESİN KULLANILDIĞI ALANLAR VE PROJELER

Ses titreşimi elektrikli haberleşme araçlarında iletilebilmesi için, mikrofon gibi elektronik cihazlar tarafından elektromanyetik dalgalara çevrilerek iletilir. Bu veriler sesi oluşturarak hoparlör tarafından kulaklarımızın duymasını sağlayarak haberleşme gerçekleşir.

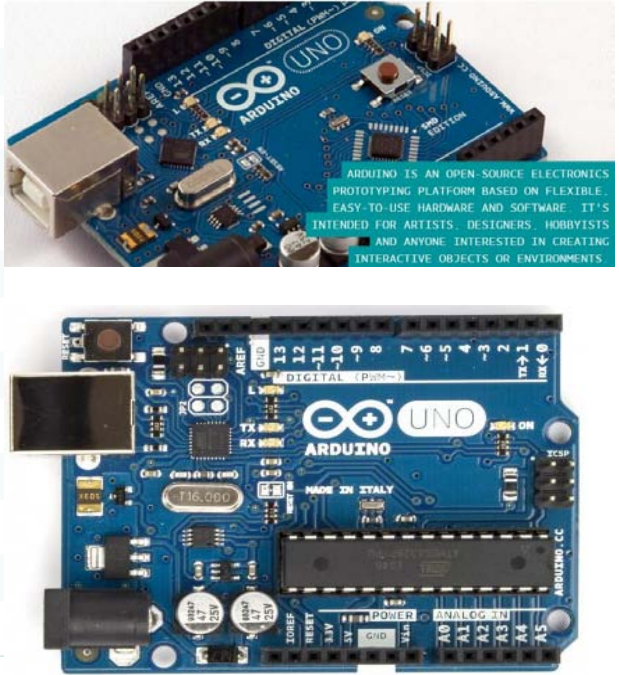
Ses işlemeden insan kulağının işitmediği titreşimlerden günlük yaşamda pek çok alanda yararlanır. Petrol arama yer kabuğu inceleme tıp endüstri ve haberleşme bu alanlardan bazılarıdır. Örneğin doktorlar insan vücudunu dinlemek için stetoskop kullanarak kalp ve akciğer hastalıklarını teşhis eder. Tıp alanında kullanım alanından başka bir örnek ise "Ultrason" dur. Ultrason da seslerin yansıtıldığı yerler organ ve dokulardır. Tıbbın her alanında ultrason ile görüntüleme yararlanılmaktadır.

## 2.ARDUINO UNO

Arduino aynı zamanda mikro denetleyici cihazın adı olarak da kullanılmaktadır. Baş tasarımcılarının (Massimo Banzi ve David Cuartielles) İtalyan olmaları nedeniyle cihazın adı da doğal olarak İtalyancadan seçilmiş... Kelime "Sıkı arkadaş" anlamına gelen bir erkek ismidir. Wikipedia kaynağına göre Arduino'ya ilham veren Wiring platformu, Ivrea Tasarım Enstitüsü'nde Hernando Barragan tarafından geliştirilmiş. Ivrea'lı Arduin ise bu enstitünün bulunduğu kasabaya ait tarihi bir karakterdir.

Arduino Uno ATmega328 tabanlı bir mikroişlemci kartıdır. Bu mikroişlemci kartının teknik özellikleri aşağıdaki gibidir;

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB
Flash Memory for Bootloader	0.5 KB
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

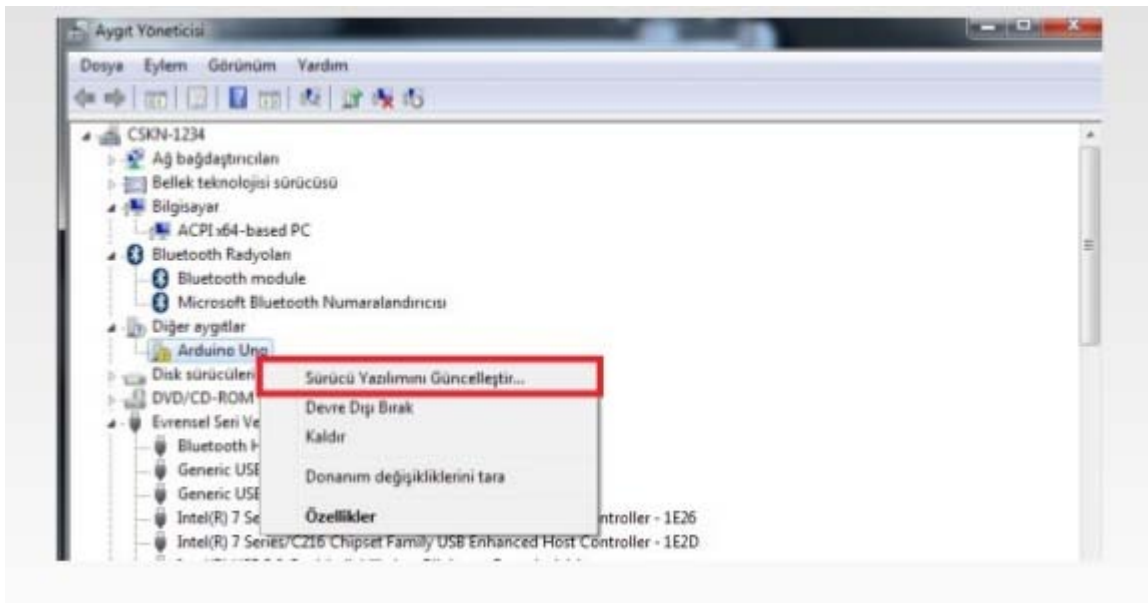


## 2.1.ARDUİNO'NUN BİLGİSAYARA TANITILMASI

İlk olarak arduino'yu USB üzerinden bilgisayara bağlıyoruz. Sürücüyü doğal olarak bilgisayarımız ilk başta tanımıyor çünkü arduino'yu kullanabilmek için ilk olarak bilgisayarımıza tanıtmamız gerekiyor.

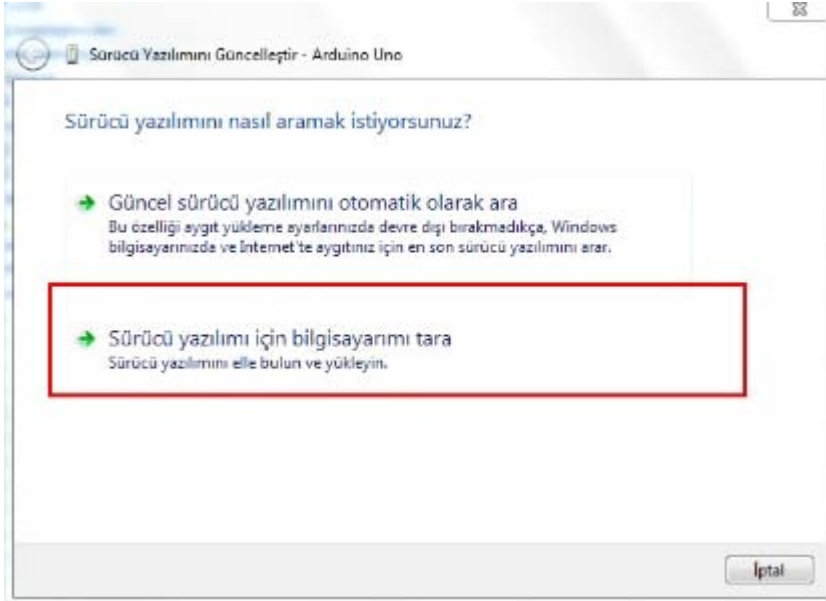


Bunu için Denetim masasından Aygıt yöneticisini açıyoruz ve diğer aygıtlar altında tanıtılmamış olan Arduino UNO'yu görünüyor ve yazıcıyı güncelleştir diyoruz.

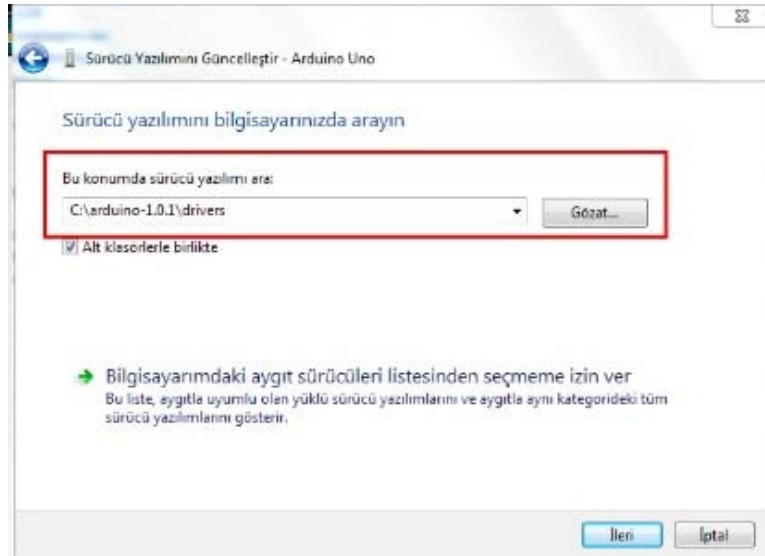




Sürüler indirdiğimiz zip dosyası içinde drivers klasörü altında bulunmaktadır bunu için bilgisayarı tara seçeneğiyle ilerliyoruz.



Bilgisayarımızdaki drivers klasörünün bulunduğu yolu gösteriyoruz.



Sonuç olarak Arduino UNO kartımızı bilgisayarımıza tanıtmış olduk .



## 2.2.ARDUİNO IDE

Geliştirme ortamı ile programımızı yazıp derleyip kartımıza yükleyebiliyoruz. Bunun için Arduino geliştirme ortamında kısa yollar bulunmaktadır. Örneğin derlediğimiz kodlarımızı upload butonu ile arduino kartımıza yükleyebiliriz. Arduino geliştiricilerinin yazdığı bir çok programa geliştirme editörü üzerinden ulaşabiliriz. Bunu için File – Examples yolunu takip edebiliriz.

Arduino Uno 'da koşulacak olan kodlar için Arduino'nun web sayfasından indirdiğimiz Arduino IDE kullandık. Bu kodların Arduino'ya aktarımı, üzerinde bulunan usb girişi ile bilgisayara bağlantısını sağlamaktadır. Arduino ile ilgili yazılmış bir örnek aşağıda gösterilmiştir.



```
sketch_may21b $
int ledPin = 13;
int beklemesuresi=100;
void setup()
{
  pinMode(ledPin, OUTPUT);    // 12. pini çıkış pini olarak setliyoruz.
}

void loop()
{
  digitalWrite(ledPin, HIGH); // Led e 5 V verilir ve yanar
  delay(beklemesuresi);      //delay fonksiyonu aldığı değere göre bekler
  digitalWrite(ledPin, LOW);
  delay(beklemesuresi);
  digitalWrite(ledPin, HIGH);
}
```

Arduino IDE de yazılan kodlar genel yapısı ile C diline benzemektedir. C ve C++ dillerinin sözdizimini kullanmaktadır. Arduino programlama da `setup()` ve `loop()` fonksiyonları mutlaka olmak zorundadır. ilk olarak `setup()` fonksiyonu çalışır daha sonra `loop()` fonksiyonu çalışır ve bu arduino'nun çalışması süresince devam eder.

### 2.3.GÜÇ

Arduino Uno USB bağlantısı veya harici güç kaynağı ile güç alabilir. Güç kaynağı otomatik olarak seçilir. Harici (non - USB) güç bir AC - DC adaptörü veya pil ile gelebilir.

### 2.4.BELLEK

Bu kart ATmega328 ve önceden yüklü olarak bootloader ile tam montajlı ve test edilmiş olarak gelmektedir. ATmega328, bootloader kullandığı 0.5KB dahil olmak üzere 32KB belleğe sahiptir. 2KB' ına sadece SRAM, 1KB' ına sadece EEPROM sahiptir.

### 2.5. GİRİŞ VE ÇIKIŞ

Uno üzerindeki 14 dijital pinden her biri `pinmode()`, `digitalWrite()` ve `digitalRead()` fonksiyonları kullanılarak giriş yada çıkış olarak kullanılabilir. Bu pinler 5V ile çalıştırılabilir. Her bir pin max 40 mA alır ve 20 – 50 kOhms olan iç pull – up direncine sahiptir. Ek olarak bazı pinler özelleştirilmiş fonksiyonlara sahiptir:

#### **Seri: 0 (RX) ve 1 (TX)**

RX ve TX TTL seri bilgilerini almak için kullanılmaktadır. Bu pinler ATmega8U2 deki USBto-TTL seri çipinin yerini tutması için bağlanmıştır.

#### **Harici Kesmeler: 2 ve 3**

Bu pinler düşük seviyedeki bir kesmeyi, alçalan yada yükselen kenarlı bir kesmeyi ve değerdeki bir değişimi tetiklemesi için konfigure edilmiştir. `attachInterrupt()` fonksiyonu ile daha fazla detay öğrenilebilir.

### **PWM: 3, 5, 6, 9, 10, AND 11**

analogWrite() fonksiyonuyla birlikte 8 bitlik PWM çıkışı sağlar.

### **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK)**

Bu pinler SPI kütüphanesini kullanarak SPI iletişimi desteklemektedir.

### **KONTROL LED 13**

Dijital 13 pinine bağlı bir led mimarisi mevcuttur. Pin yüksek seviyeye çekildiğinde led açıktır yani yanar, düşük seviyeye çekildiğinde ise led kapanır. Uno A0...A5 olarak etiketlenmiş 6 adet analog girişe sahiptir. Her biri 10 adet bite sahiptir. Varsayılan olarak toprak 5V ölçülmektedir. analogReference() fonksiyonuyla beraber kullanılan AREF pininin en üst seviyesini değiştirmek mümkündür. Ayrıca bazı pinler özel olarak işlevsellendirilmiştir:

### **RESET**

Mikrodenetleyiciye reset atmak için bu pin low seviyeye getirilir. Genelde boardın üzerinde bulunan bir grup shield resetleme butonu eklemek için kullanılmaktadır.

## **2.6. HABERLEŞME**

Arduino Uno bir bilgisayarla, diğer bir Arduino ile ya da diğer mikrodenetleyiciler ile haberleşmek için bir takım özelliklere sahiptir. ATmega328, pin 0 ( RX ) ve pin 1 ( TX ) dijital pinleri üzerinde kullanılabilir olan UART TTL ( 5V ) seri haberleşme sağlar. Board üzerindeki bir ATmega16U2 USB üzerinden seri haberleşme sağlar ve bilgisayarda sanal bir com port gibi görünmektedir.

## **2.7. PROGRAMLAMA**

Arduino Uno, Arduino yazılımı kullanılarak programlanabilmektedir. Arduino Uno üzerinde bir programcı olmadan da yeni kod yüklemeyi sağlayan bir bootloader gelmektedir. Bootloader orijinal STK500 protokolünü kullanarak haberleşmektedir.

## **OTOMATİK ( YAZILIM ) RESET**

Yüklemeden önce reset butonuna fiziksel bir dokunuştan ziyade, Arduino Uno bağlandığı bilgisayar üzerinden yazılım koşularak resetlemenin çeşitli yolları olacak şekilde dizayn edilmiştir.

## **2.8 USB AŞIRI AKIM KORUMASI**

Arduino Uno, gelen aşırı akımdan bilgisayarınızdaki USB portu koruyan, resetlenebilen bir polyfuse' e sahiptir. Çoğu bilgisayar kendi iç koruma sağlamasına rağmen, sigorta ekstra bir koruma katmanı sağlar. Eğer USB bağlantı noktasına 500 mA' den fazla akım uygulanırsa, fuse, aşırı yük kaldırılincaya kadar otomatik olarak bağlantıyı koparacaktır.

## **2.9 FİZİKSEL ÖZELLİKLER**

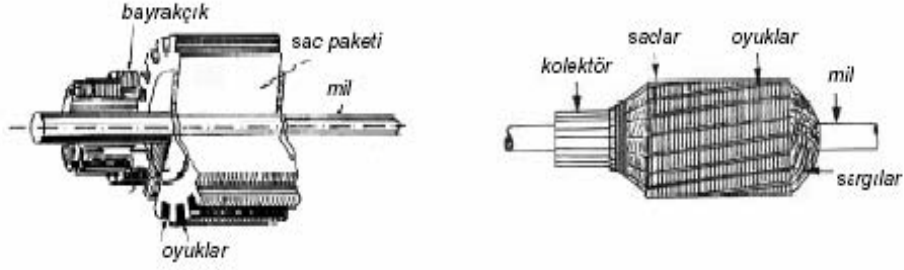
Uno PCB' nin maksimum uzunluğu 2.7 ve genişliği ise 2.1 inçdir. Uno yüzey ve duruma bağlı olarak board üzerinde 4 adet vida çukuruna izin vermektedir. Dijital pin 7 ve pin 8 arasındaki mesafe 160 mil ( 0.16'' ) dir. Diğer pinlerin 100 mil aralığı içinde 2 kat mevcuttur.

## **3.DC MOTOR**

Doğru akım motorları, doğru akım elektrik enerjisini dairesel mekanik enerjiye dönüştüren elektrik makineleridir. Yapıları DC generatörlere çok benzer.

### **3.1.ENDÜVİ**

Endüvi; DC dinamo, DC motor ve AC seri motorun, dönen kısmıdır. Bu eleman, 0,30,7 mm kalınlığında çelik saclardan yapılmış silindirik gövde üzerine açılmış ve oluklara yerleştirilmiş sargılardan oluşmuştur. Endüvi sargılarının uçları, bakır dilimlerinden yapılmış olan ve üzerine fırçaların temas ettiği kısma (kolektöre) bağlanmıştır. Şekilde endüvinin yapısı verilmiştir.



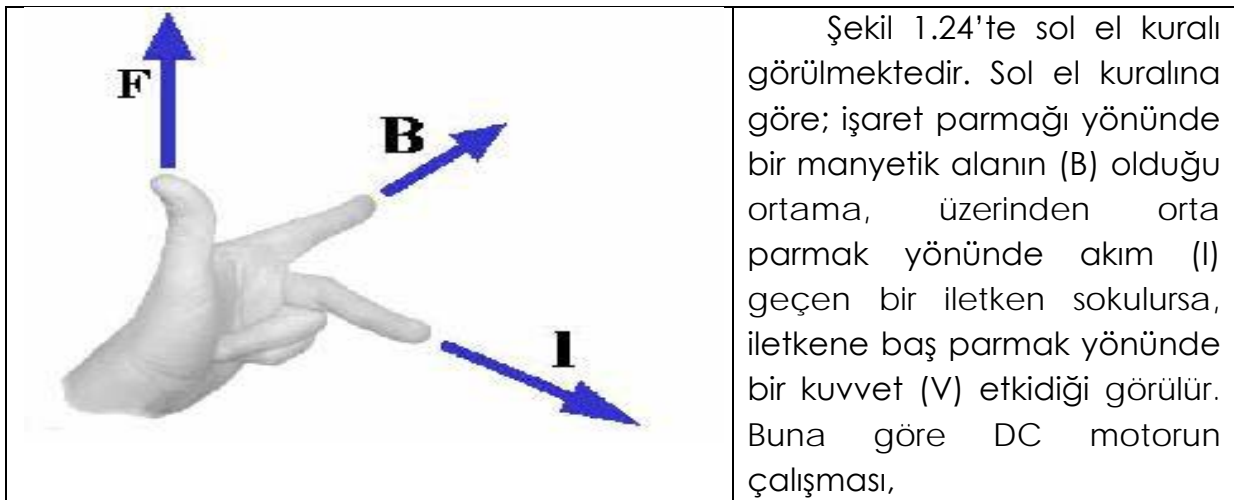
### 3.2.GÖVDE

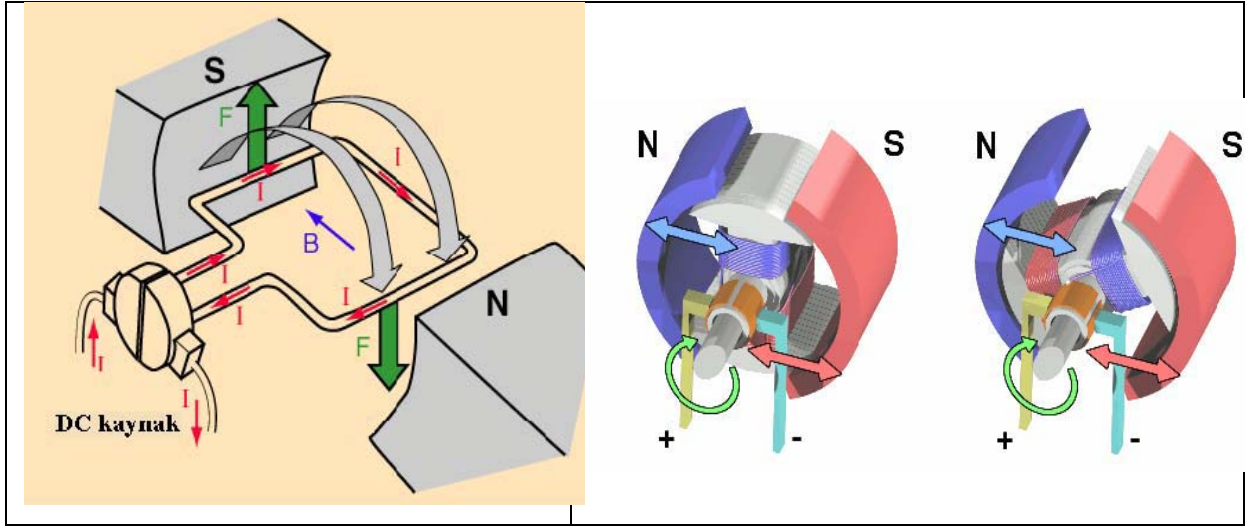
DC ya da AC ile çalışan makinelerde N-S kutuplarının oluşturulması için yapılmış olan sargıların yerleştirildiği kısma indüktör denir. Küçük makinelerin indüktörleri doğal mıknatıstan yapılırken, yüksek güçlü makine indüktörleri



### 3.3.DC MOTORUN ÇALIŞMA ESASLARI

İçerisinden akım geçen iletken, manyetik alana sokulursa iletkene bir kuvvet etkir.



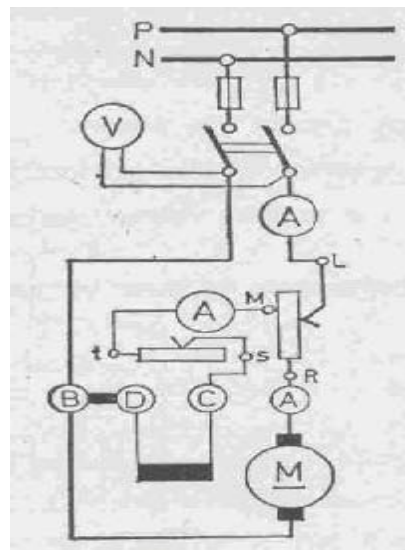


### 3.4. DC MOTORLARIN MEKANİK ÖZELLİKLERİ

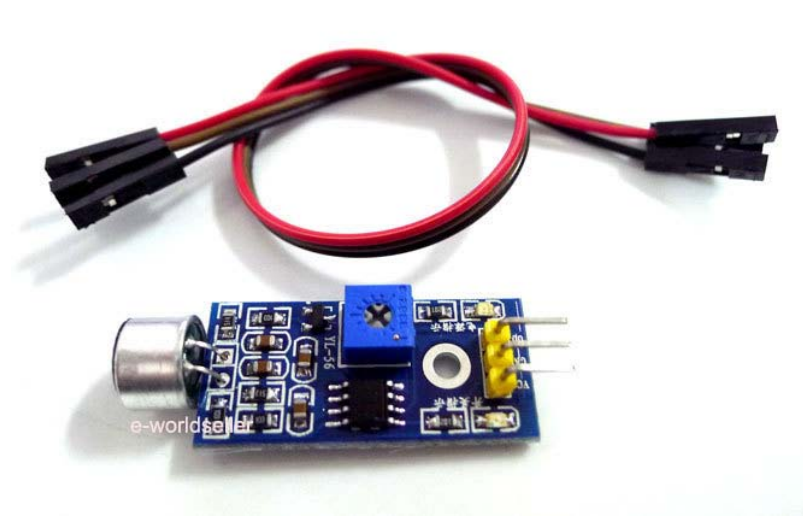
Endüvi iletkenlerinden akım geçtiğinde, iletkenler manyetik alan tarafından alanın dışına doğru itilir. Böylece endüvi çevresinde endüviyi döndüren bir kuvvet veya bir moment meydana gelir. DC motorunun endüvisinde meydana gelen moment, kutupların manyetik alanı ve endüviden geçen akımla doğru orantılıdır.

### 3.5. DC MOTORUN HIZ DENETİMİ

DC motorlarında devir sayısı, kutup gerilimi ve kutup manyetik akışına bağlıdır. Uyarım direnci ayarlanarak devir ayarlanır. Şekil 1.31'de DC şönt motorun hız denetim devresi verilmiştir.



#### 4.AKUSTİK SES SENSÖRÜ - LM393



- Ses algılama modül iki çıkış:
  1. AO, analog çıkış, gerçek- zamanlı çıkış voltaj sinyali mikrofon görevi görür
  2. Ses belirli bir eşik üzerine çıktığı zaman sinyal üretir

Modülü özellikleri:

kullanımı 5v dc güç kaynağı

analog çıkış

güç gösterge ışığı

- Dijital çıkış:

```
Int led=13;// led arayüzü tanımlandı
Int buttonpin=3// d0 sensörü arayüzü tanımlandı
Int val;// sayısal değişken val
Geçersiz kurulum() geçersiz kurulum()
{
Pinmode( led, çıkış);//tanımlamak olarak açtı çıkış arayüzü
Pinmode( buttonpin, giriş);//tanımlamak d0 olarak sensör çıkış arayüzü
}
Geçersiz döngü() geçersiz döngü()
{
Val=digitalread( buttonpin);//dijital arayüz atanacak değeri 3 okumak için val
```



```
If( val==high)/Işık sensörü algılar bir sinyal kesilir, ışığı yanıp sönüyor
{
Digitalwrite( led, yüksek)
}
başka
{
Digitalwrite( led, düşük)
}
}
```

- 2. analog çıkış:

```
Int sensorpin = a5;//seçin giriş pim potansiyometre
Int ledpin = 13;//seçin pin led
Int sensorvalue = 0;//değişken mağaza değeri gelen sensörü
```

```
if(value>=treshold)
{
Pinmode( ledpin, çıkış);
Seri. Başlar(9600);
}

else ()
{
Sensorvalue = analogread( sensorpin);
Digitalwrite( ledpin, yüksek);
delay( sensorvalue);
Digitalwrite( ledpin, düşük);
delay( sensorvalue);
Seri. Println( sensorvalue, Aralık);
}
```

## 5.UYGULAMA ADIMLARI

Projede gerçekleştirilen ortamda bulunan ses şiddetinin analog olarak girişini alarak kod içerisinde map edilerek bu değerleri volume değişkenine atanan sesin hangi seviyede olduğu belirlendi. Bu işlemler aşağıdaki kodlar ile gerçekleştirildi

```
volume = analogRead(A0);
```

```
Serial.print("Ses =");
```

```
Serial.println(volume);
```

```
delay(1000);
```

Atanan volume değerlerini kullanıcı tarafından hangi seviyede olduğunu gözlemlenmesi için 3 seviyeli olarak, kırmızı-sarı-yeşil ledler aktif edildi. Hangi seviyede olduğunun tespiti yapıldıktan sonra o seviyeye ait olan işlemler aktif hale getirildi.

```
.
```

```
.
```

```
.
```

```
int lowLed = 13 ; //kırmızı led ses yüksek
```

```
int highLed = 12 ; // sarı led ses düşük
```

```
int zeroLed = 11 ; //yeşil led ses yok
```

```
if(volume>=highThreshold)
```

```
{
```

```
digitalWrite(highLed, HIGH); // kırmızı led yak
```

```
for(int i=0;i<15;i++)
```

```
{
```

```
.
```

```
.
```

```
.
```

```
else if(volume>=lowThreshold)
{
    digitalWrite(lowLed, HIGH); // sarı led yak

    for(int i=0;i<5;i++)
    {
        .
        .
        .
    }
}
```

```
else if(volume>=zeroThreshold)
{
    digitalWrite(zeroLed, HIGH); // yeşil led yak

    for(int i=0;i<5;i++)
    {
```

Sesin şiddetinin seviyesine göre karşılaştırmalar gerçekleştirilerek dc motora aktarılacak olan akım seviyesi değiştirilerek hız kontrolünü rael time olarak aktarıldı. Bu sayede beşiğin belirli aralıklarda hız kontrolü gerçekleştirildi.

Ses sensöründen alınarak volume değişkenine atılan değere göre dc motora gönderilecek olan akım değeri aşağıdaki kod ile gerçekleştirildi.

```
.
.
.

digitalWrite(highLed, HIGH); // kırmızı led yak
for(int i=0;i<15;i++)
{
    potansDegeri = 1023; //maksimum hız
    gerilimDegeri= (potansDegeri/1023.0)*5.0;
```

```
sonuc=map(potansDegeri,0,1023,0,255);
```

```
digitalWrite(lowLed, HIGH); // sarı led yak
```

```
for(int i=0;i<5;i++)  
{  
    potansDegeri 830; //yavaş seviyede hız  
    gerilimDegeri= (potansDegeri/1023.0)*5.0;  
    sonuc=map(potansDegeri,0,1023,0,255);
```

.  
. .  
. .

```
digitalWrite(zeroLed, HIGH); // yeşil led yak
```

```
for(int i=0;i<5;i++)  
{  
    potansDegeri = 0; //stop  
    gerilimDegeri= (potansDegeri/1023.0)*5.0;  
    sonuc=map(potansDegeri,0,1023,0,255);
```

.  
. .  
. .

Ses sensörü ile aldığımız ses değişkeninin ve dc motorun hız değerini bilgisayar ekranında kontrolünü sağlamak için serial port üzerinden görüntülendi.

```
Serial.print("Ses =");  
Serial.println(volume);
```

```
Serial.print("Potans Degeri =");  
Serial.println(potansDegeri);
```

Algılanan ses seviyesine göre aktif edilecek olan dc motorun beşiği sallarken tekrar tekrar hız değişimi olmasını engellenmesi için eşik değeri ile kıyaslandıktan sonra aşağıdaki gibi döngüye girilerek belirli bir süre aynı hızda sallanması sağlandı. Belirli süre sonunda bebeğin uyuyup uyumadığının tekrar kontrolü gerçekleştirilerek beşiğin tekrar aktif edilmesi değerlendirildi.

```
digitalWrite(highLed, HIGH); // kırmızı led yak
for(int i=0;i<15;i++)
{
    potansDegeri = /* analogRead(potansAnalogGiris) */ 1023;
    gerilimDegeri= (potansDegeri/1023.0)*5.0;
    sonuc=map(potansDegeri,0,1023,0,255);

    Serial.print("Ses =");
    Serial.println(volume);

    Serial.print("Potans Degeri=");
    Serial.println(potansDegeri);

    /* Serial.print("Gerilim =");
    Serial.println(gerilimDegeri);

    Serial.print("Sonuc =");
    Serial.println(sonuc);
    */
    delay(1000);

    analogWrite(motor,sonuc);
}
}
```

## 6.SONUÇ VE ÖNERİLER

Bu bitirme çalışmasında ATmega328 mikrodenetleyicisi kullanılarak mekanik hareketler yapabilen beşik üzerinde bir tasarım gerçekleştirildi.

Akustik ses sensörü ile üzerinde bulunan mikrofon sayesinde alınan analog girişi direkt olarak çıkış verebilse de mikrofon hassasiyeti çok düşük olduğundan dolayı, ses şiddet seviyesine göre ayarlanabilen sistemde çeşitli sorunlar yaşandı.

Gerçekleştirdiğimiz bu projede daha kapsamlı bir şekilde uygulanabilirliği yüksek bir proje haline getirilmek istersek, projemize eklenebilecek özellikler şu şekilde sıralanabilir;

- Belirli uyarılara göre (ağlama sesi, hareket, gürültü gibi..) sistemdeki GSM modülü aktif olacak ve ebeveynin cep telefonuna SMS ile bilgi verebilecek.
- İletişim modülü; uyarı belirli bir eşik değerinin üzerinde olduğunda aktif olacak, uyarının tespit edilmediği durumlarda kullanıcı kontrolüne ihtiyaç duyulmadan otomatik olarak devre dışı tutulacaktır. (GSM modülü kullanılmak istenmezse Bluetooth veya WiFi ile iletişim gerçekleştirilebilir.)
- Uyarı seviyesine göre (ağlama sesi) istenirse beşik ayarlanabilir bir kademedede otomatik olarak sallanabilecek
- Ebeveyn tercihinine göre otomatik ninni, müzik veya benzeri bir ses çalınabilecek
- Ebeveyn tercihinine göre led ile aydınlatma aktif olabilecek (İstendiğinde bebeğin uyumasına yardımcı tavan aydınlatması, ve bebek uyku halinde iken bakımının yapılabilmesini sağlayabilecek gece lambası)
- Hareket, gürültü anında ebeveynin telefonuna anlık görüntü ve sesini aktarılacak.

## KAYNAKLAR

<http://arduino.cc/en/Main/arduinoBoardUno>

<http://www.atmel.com/Images/doc8161.pdf>

<http://www.instructables.com>

<http://en.wikipedia.org/wiki/Arduino>

<http://arduinoturkiye.com>