



**BAŞKENT ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ
EEM322 DENETİM SİSTEMLERİ
DÖNEM PROJESİ RAPORU**

15/05/2015

KETTLE KONTROL

Hazırlayanlar:



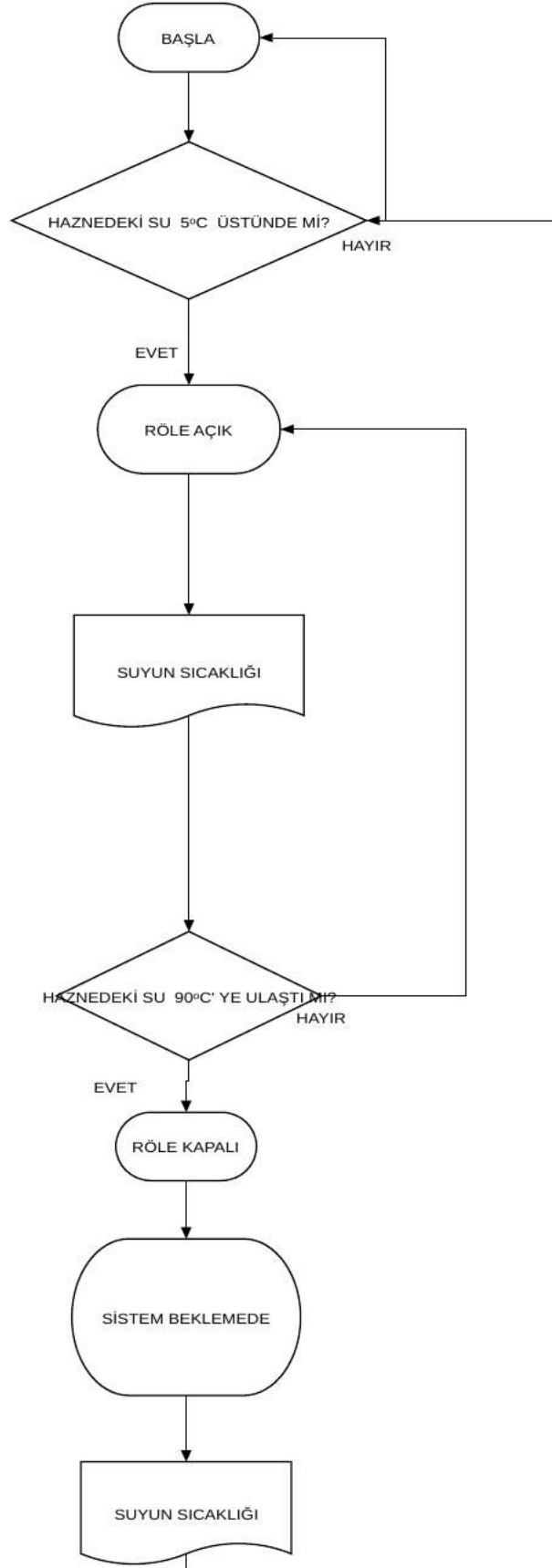
İÇİNDEKİLER

Akış Şeması	1
1.0 - Projenin Tanımı	2
1.1 - Proje Elemanları	2
1.1.0 - Arduino UNO	2
1.1.1 - Rezistans	2
1.1.2 - Sıcaklık Sensörü	2
1.1.3 - Röle	2
1.1.4 - Su Haznesi	2
1.1.5 - Jumper ve Kablolar	3
1.2 - Projenin İşleyişi	3
1.3 - Sistem Modelleme	3
1.3.1 - Yazılım	3
1.3.2 - Donanım	3
1.4 - Benzetim Sonuçları	4
1.5 - Görseller	5

EK-1 (Kod Çıktısı)

EK-2 (DS18B20 Datasheet)

AKIŞ ŞEMASI



1.0 Projenin Tanımı

Cattle kontrol projesi ile haznedeki suyun sıcaklığının ölçülmesi ve belirlenen sıcaklıkta bir süre tutulması sağlanmıştır.

1.1. Proje Elemanları

1.1.0. Arduino UNO:

ATmega328 işlemci kullanan Arduino çeşididir. 14 dijital giriş/çıkış pini bulunur, bunlardan 6'sı PWM çıkışı olarak kullanılabilir. 6 analog giriş pinine sahiptir. 16 MHz kristal osilatörü, USB bağlantısı, 2.1mm güç girişi, ICSP başlığı ve reset butonu vardır. Mikroişlemciyi destekleyecek herşeye sahiptir. Çalıştırmak için DC 7~12V güç kaynağına bağlamak yeterlidir. 1kb EEPROM hafızası vardır.

1.1.1. Rezistans:

Rezistans, elektrik enerjisini ısı enerjisine dönüştüren direnç tellerine verilen genel isimdir. Rezistans oldukça yüksek ısılara karşı direnç gösteren bir üründür. Rezistansların içerisinde nikel, demir, krom ve alüminyum alaşımları bulunmaktadır.

1.1.2. Sıcaklık Sensörü (DS18B20 Dijital Sıcaklık Sensörü):

Su geçirmez dijital bir sensördür. Çalışma voltajı 3.0V- 5.0V aralığındadır. 12 bitlik okuma gerçekleştirilebilir 0.5°C hassasiyetle -10°C ile 125°C arasında ölçüm yapabilir. Tepki süresi 750ms'den daha düşüktür. Sensörün datasheeti ekte belirtilmiştir.(EK-2)

1.1.3. Röle:

Röle, başka bir elektrik devresinin açılıp kapanmasını sağlayan bir elektriksel anahtardır. Üzerinde bulunan elektromanyetik bobine rölenin türüne uygun olarak bir gerilim uygulandığında bobin mıknatıs özelliği kazanır ve karşısında duran metal bir paleti kendine doğru çekerek bir veya daha fazla kontağı birbirine irtibatlayarak bir anahtar görevi yapar.

1.1.4. Su haznesi:

Hazne olarak plastik bir kap kullanılmıştır.

1.1.5. Jumper ve kablolar:

Sensör ve rölenin Arduiona'ya bağlanması için erkek-dişi jumperlar kullanılmıştır.

1.2. Projenin İşleyişi

Hazneye konan suyun sıcaklığı sıcaklık sensörü ile ölçülecek ve ölçülen sıcaklık 0°C üstüdeyse Arduino'dan gelen sinyal ile röle tetiklenecek ve 220V şehir şebekesi rezistansı aktif hale getirerek suyun sıcaklığını arttırmaya başlayacaktır. Sıcaklık sensörü ile suyun sıcaklığı ölçülmeye devam edilecek ve su 90°C' ye ulaştığında Arduino röleyi kesecek ve rezistans kapanacaktır ve seri port ekranında "kaynıyor" yazısı görülecektir. Daha sonra sistem, belirlenen süre kadar kendisini beklemeye alacaktır. Bu süre bitiminde haznedeki suyun sıcaklığı sensör yardımıyla tekrar ölçülecek ve sıcaklık 90°C altındaysa Arduino röleyi aktifleştirerek rezistansın suyu kaynatmasını sağlayacaktır. Bu şekilde haznedeki su sürekli olarak ısıtılacaktır.

1.3. Sistem Modelleme

1.3.1. Yazılım

Sistem yazılımı Arduino programı ile yazılmıştır. Sıcaklık sensörü için 1-Wire ara yüzü kullanılmıştır. Belirlenen bekleme süreleri "delay" komutlarıyla gerçekleştirilmiştir. Kod çıktısı ektedir.(EK-1)

1.3.2. Donanım

Dijital sıcaklık sensörü , röle ve rezistansın bağlantıları gerçekleştirilmiştir. Dijital sensör için Arduino 1-Wire ara yüzü sayesinde haberleşme için bir data pini, 5V besleme ve Ground pinleri kullanılmıştır. Röle için bir dijital çıkış pini ve Ground pini kullanılmıştır. Rezistansın kablo bağlantıları yapılarak 220V ile çalışacak hale getirilmiştir. Arduino'dan gelen sinyaller ile rezistansın aktif hale gelebilmesi için şehir şebekesinden gelen fazlardan birine röle bağlanmıştır. Röle Arduino'dan gelen sinyal ile aktifleşerek 220 voltun rezistansa ulaşarak suyu ısıtmasını sağlamaktadır. Gerçekleştirilen işlemler Arduino programı içerisinde seri port ekranından görülebilmektedir.

Yazılım ve sistem elemanlarının bağlantıları yapıldıktan sonra , sistemi gerçekleştirmek adına hazne delinerek rezistans ve sensör hazne içerisine yerleştirilmiştir. Yalıtım işleri ve 220V kablo çekimi gerçekleştirilerek sistem çalışmaya hazır hale getirilmiştir.

1.4 Benzetim Sonuçları (Seri Port Ekran Çıktısı)

```
Sicaklik: 81.44
Sicaklik: 82.44
Sicaklik: 82.12
Sicaklik: 81.12
Sicaklik: 79.75
Sicaklik: 78.12
Sicaklik: 76.44
Sicaklik: 74.75
Sicaklik: 73.12
Sicaklik: 72.00
Sicaklik: 71.62
Sicaklik: 70.81
Sicaklik: 70.56
Sicaklik: 71.06
Sicaklik: 70.69
Sicaklik: 70.31
Sicaklik: 70.44
Sicaklik: 70.87
Sicaklik: 70.87
Sicaklik: 70.62
Sicaklik: 70.50
Sicaklik: 71.06
Sicaklik: 71.19
Sicaklik: 70.87
Sicaklik: 70.94
Sicaklik: 70.69
Sicaklik: 70.06
Sicaklik: 69.56
Sicaklik: 69.56
Sicaklik: 69.87
Sicaklik: 70.81
Sicaklik: 71.19
Sicaklik: 70.94
Sicaklik: 70.44
Sicaklik: 69.87
Sicaklik: 69.25
Sicaklik: 69.81
Sicaklik: 71.94
Sicaklik: 74.75
Sicaklik: 79.12
Sicaklik: 85.25
Sicaklik: 91.25
Sicaklik: 91.25
Kaynama
Beklemede
```

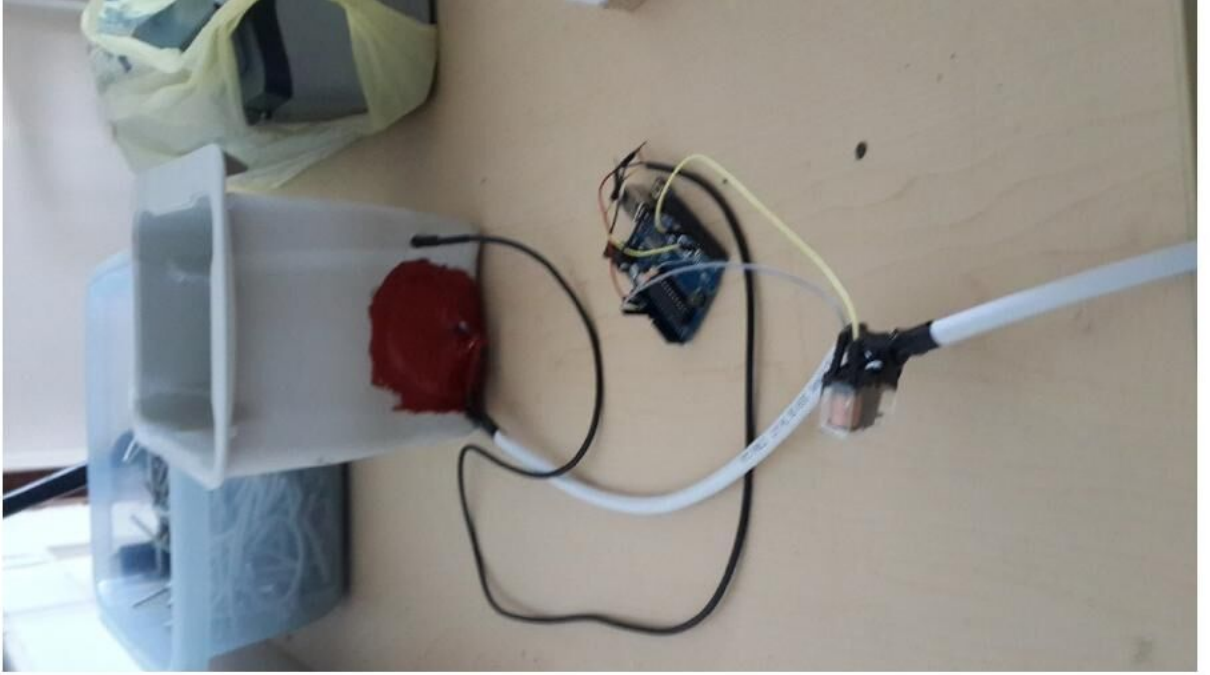
1.5. Görseller

Görsel 1 – Arduino Pin Bağlantıları

Görsel 3 - Rezistans ve Sensörün Hazneye Yerleştirilmesi



Görsel 4 - Yalıtım İşlemi



EK -1 : Kod Çıktısı

```
#include <OneWire.h> // OneWire kütüphanesini ekliyoruz.
```

```
// Sıcaklık sensörünü bağladığımız dijital pini 2 olarak belirliyoruz.
```

```
int DS18S20_Pin = 2;
```

```
// Sıcaklık Sensörü Giriş-Çıkışı
```

```
OneWire ds(DS18S20_Pin); // 2. Dijital pinde.
```

```
// Role için 7 numaralı pin belirlenmiştir.
```

```
int role = 7;
```

```
void setup(void) {
```

```
    Serial.begin(9600); // Seri iletişimi başlatıyoruz.
```

```
    pinMode(role,OUTPUT);
```

```
}
```

```
void loop(void) {
```

```
    // temperature değişkenini sıcaklık değerini alma fonksiyonuna bağlıyoruz.
```

```
    float temperature = getTemp();
```

```
    // Sensörden gelen sıcaklık değerini Serial monitörde yazdırıyoruz.
```

```
    Serial.print("Sicaklik: ");
```

```
    Serial.println(temperature);
```

```
    // 1 saniye bekliyoruz. Monitörde saniyede 1 sıcaklık değeri yazmaya devam edecek.
```

```
    delay(1000);
```

```
    if(temperature > 0 && temperature <= 90) // sensörden ölçülen sıcaklık 0 dereceden büyük 90 dereceden küçükse röleyi aktifleştiriyoruz.
```

```
{
```

```

digitalWrite(role,HIGH); // ilgili pin HIGH yapılarak röle aktifleştirildi.
}

if(temperature >= 91) // suyun sıcaklığı 91 dereceye ulaştığında röleyi 10 saniye daha çalıştırıyoruz.
{
    digitalWrite(role,HIGH);

    Serial.print("Sicaklik: "); // serial monitörde sıcaklık değeri yazdırılıyor
    Serial.println(temperature);
    Serial.print("Kaynama\n"); // serial monitorde "kaynama" yazdırılıyor.
    delay(10000);

    digitalWrite(role,LOW); // belirlenen süre bittiğinde röleyi kapatarak sistemi beklemeye alıyoruz
    Serial.print("Beklemede\n"); // serial monitorde "beklemede" yazdırılıyor.
    delay(60000);
}
}

// **Sensörden gelen verileri Celcius cinsinden sıcaklık değerlerine çeviriyoruz
float getTemp(){
    // suyun sıcaklığını Celcius cinsinden sensörden bu fonksiyonla alıyoruz

    byte data[12];
    byte addr[8];

    if ( !ds.search(addr)) {
        // tek sensör kullanıyoruz bu noktada aramayı resetliyoruz
        ds.reset_search();
        return -1000;
    }
}

```

```
}
```

```
if ( OneWire::crc8( addr, 7) != addr[7]) {
```

```
    Serial.println("CRC is not valid!");
```

```
    return -1000;
```

```
}
```

```
if ( addr[0] != 0x10 && addr[0] != 0x28) {
```

```
    Serial.print("Device is not recognized");
```

```
    return -1000;
```

```
}
```

```
ds.reset();
```

```
ds.select(addr);
```

```
ds.write(0x44,1); //
```

```
byte present = ds.reset();
```

```
ds.select(addr);
```

```
ds.write(0xBE); //
```

```
for (int i = 0; i < 9; i++) { // 9 byte kullanılacak
```

```
    data[i] = ds.read();
```

```
}
```

```
ds.reset_search();
```

```
byte MSB = data[1];
```

```
byte LSB = data[0];
```

```
float tempRead = ((MSB << 8) | LSB); // iki komplementini alıyoruz  
float TemperatureSum = tempRead / 16;  
  
return TemperatureSum;  
  
}
```