

T.C.

Gazi Üniversitesi

Teknik Bilimler Meslek Yüksekokulu

Elektronik Teknolojisi Programı

# SCADA arayüzlü nem ile sıcaklık izleme ve kontrol yazılımı

Gökhan DÖKMETAS 203702001

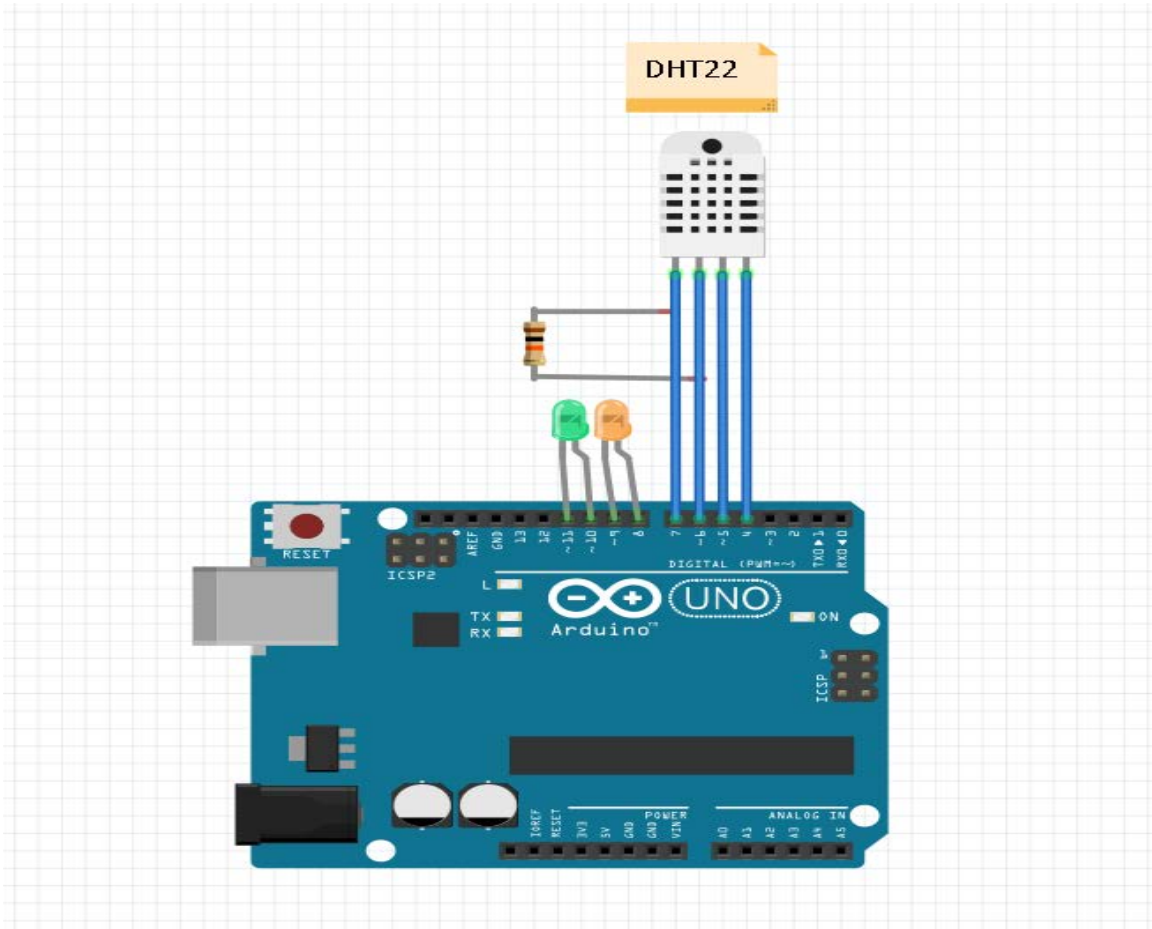
## Projenin amacı ve kısa açıklaması

Sıcaklık ve nem izleme sistemleri endüstrinin pek çok alanında ihtiyaç duyulan, özellikle gıda ve hayvancılıkta belli değer aralıklarını koruma zorunluluğundan dolayı sıkça kullanılan teknolojilerdir. Bu uygulamada özellikle kuluçka makinelerinde kullanılmak üzere odanın/kabinin sıcaklığını ve nemini izleyen ve gerektiğinde operatörün ısıtma veya nemlendirme aletlerinin rölesini açmasına, asgari ve azami değerleri belirterek otomatik kontrol yapmasına imkân veren bir yazılım geliştirilmiştir.

## Projenin Donanım Tarafı

Projede kullanılan malzemeler şu şekildedir,

- Arduino UNO R3 (Klon)
- 10k Direnç
- 2 adet LED (Yeşil ve Sarı)
- DHT22 Nem ve Sıcaklık Sensörü



Devre şeması açık kaynak yazılım Fritzing ile çizilmiştir. Devre kurulumunda LED'ler Arduino kartının boş header padlerine bağlanmış olup röle/anahtarlama çıkışını temsil etmektedir. DHT22 sensörünün beslemesi (pin 1) dijital çıkış ayağından verilmiş olup, şase ayağı (pin 4) ise yine dijital çıkışın LOW (sink) yapılmasıyla devreye bağlanmıştır. Arduino'nun dijital çıkışının sağladığı güç DHT22 sensörünü beslemek için yeterlidir. Bunun yeterli olup olmadığı DHT22'nin datasheetinden kontrol edilmiştir.

## 7. Electrical Characteristics:

Item	Condition	Min	Typical	Max	Unit
Power supply	DC	3.3	5	6	V
Current supply	Measuring	1		1.5	mA
	Stand-by	40	Null	50	uA
Collecting period	Second		2		Second

\*Collecting period should be : >2 second.

Kaynak: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>

DHT22 sensörü ölçüm sırasında 1-1.5mA akıma ihtiyaç duyup bekleme modunda ise 40-50uA akım çekmektedir. Bu değerler Arduino'nun dijital çıkışından çekebileceğimiz azami akım miktarının oldukça altındadır. Bu bilgiyi ATmega328P'nin datasheetinden görebiliriz.

Parameters	Min.	Typ.	Max.	Unit
Operating temperature	-55		+125	°C
Storage temperature	-65		+150	°C
Voltage on any pin except RESET with respect to ground	-0.5		$V_{CC} + 0.5$	V
Voltage on RESET with respect to ground	-0.5		+13.0	V
Maximum operating voltage		6.0		V
DC current per I/O pin		40.0		mA
DC current $V_{CC}$ and GND pins		200.0		mA
Injection current at $V_{CC} = 0V$		$\pm 5.0^{(1)}$		mA
Injection current at $V_{CC} = 5V$		$\pm 1.0$		mA

Note: 1. Maximum current per port =  $\pm 30mA$

Kaynak : [https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P\\_Datasheet.pdf](https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf)

## Mikrodenetleyici Yazılımı

Mikrodenetleyici yazılımında Arduino Framework kullanılmış olup DHT22'yi kullanmak için Adafruit firmasının açık kaynak kütüphaneleri kullanılmıştır.

```
#include "DHT.h"
#define DHTPIN 6
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
void setup() {
  Serial.begin(9600);
  pinMode(7, OUTPUT); // DHT Besleme Ayağı
  pinMode(4, OUTPUT); // DHT GND ayağı
  digitalWrite(7, HIGH);
  digitalWrite(4, LOW);
  dht.begin();
  // LED Cikis Ayarlari
  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  digitalWrite(8, LOW);
  digitalWrite(9, LOW); // LED 1 GND
  digitalWrite(10, LOW);
  digitalWrite(11, LOW); // LED 2 GND

  delay(2000);
}
void loop() {
```

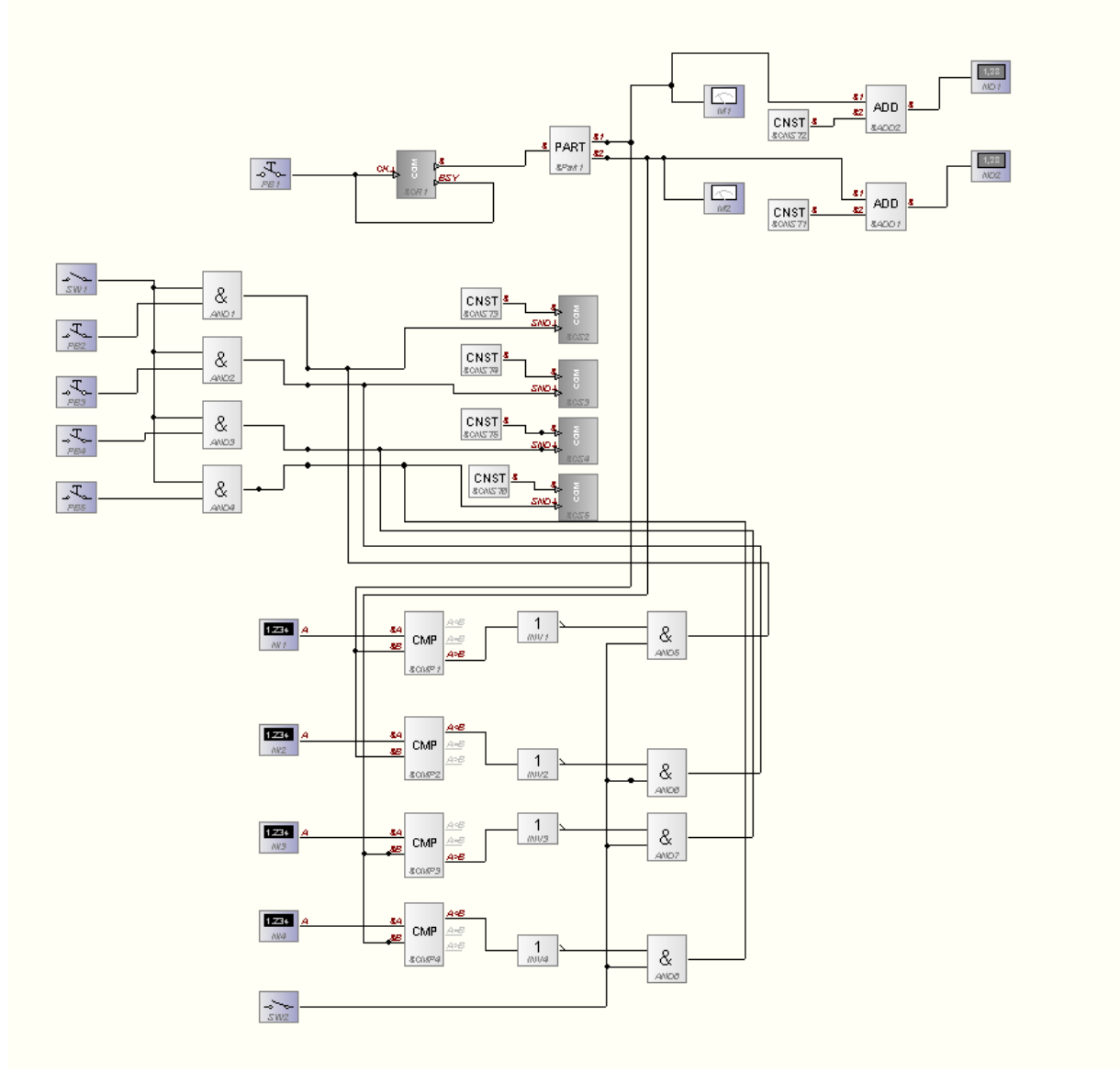
```

float h = dht.readHumidity();
float t = dht.readTemperature();
if (isnan(h) || isnan(t)) {
Serial.println(F("DHT Sensor Okuma Hatası!"));
return;
}
Serial.write(0x0D);
Serial.print(t);
Serial.write(',');
Serial.print(h);

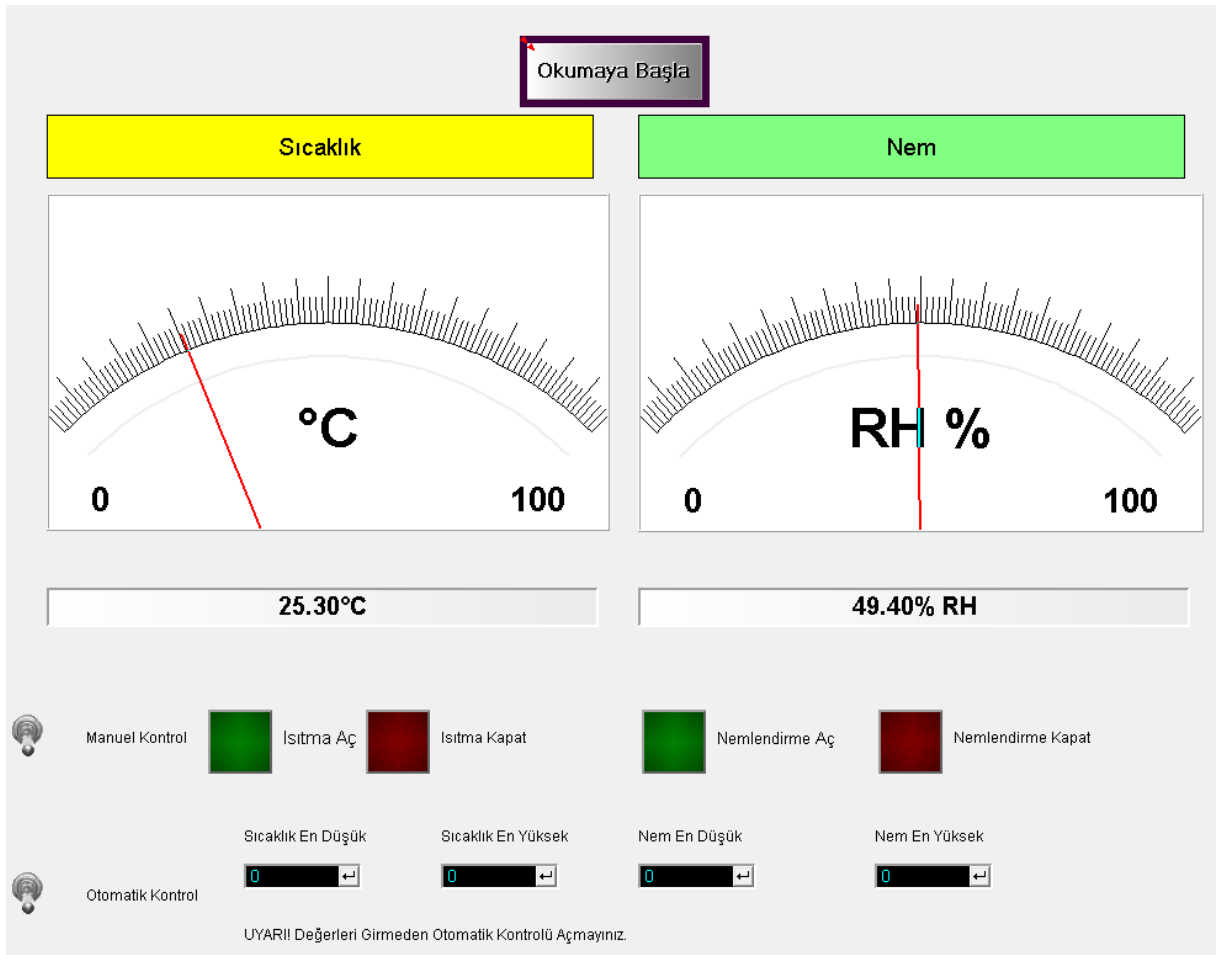
if(Serial.available() > 0)
{
char okuma = Serial.read();
switch(okuma)
{
case '1': // ISITMA AÇ
digitalWrite(8, HIGH);
break;
case '2': // ISITMA KAPAT
digitalWrite(8, LOW);
break;
case '3': // NEMLENDİRME AÇ
digitalWrite(10, HIGH);
break;
case '4': // NEMLENDİRME KAPAT
digitalWrite(10, LOW);
break;
default:
break;
}
}
delay(1000);
}

```

Mikrodenetleyici yazılımında algılayıcıdan okunan veri bir çerçeve halinde gönderilmektedir. SCADA yazılımında verinin başını belirlemek için 0x0D karakteri belirlenmiştir. Sonrasında verinin parçalara ayrılması için virgül karakterinin gözetilmesi gerektiği belirlenmiştir. Buna göre 0x0D SICAKLIK VERİSİ , NEM VERİSİ şeklinde bir veri çerçevesi bilgisayara gönderilmektedir. Veri okumada ise ASCII 1, 2, 3 ve 4 karakterleri okunmakta ve buna göre dört ayrı işlem yapılmaktadır.



Scada yazılımı görüntüleme, manuel kontrol ve otomatik kontrol olarak üç kısma ayrılmıştır. Görüntüleme kısmında seri okumayı tetiklemek için bir düğme kullanılmış, her veri geldiğinde güncellemesi için seri okuma elementinin BSY ayağı CK ayağına bağlanmıştır. Seri okumadan alınan string verisi PART elementi ile ayrılmış (gelen sıcaklık ve nem verisi) ve bunlar hem skalada hem de sayısal göstergede gösterilmiştir. Birim ifadeleri için (C, % RH) ADD elementi ile gelen karakter dizisine sabit değerler eklenmiştir. Manuel kontrol için etkinleştirme anahtarı kullanılmalıdır. Bu AND elementi ile diğer düğmelere bağlanmış ve her düğmeye basışta ayrı ayrı karakter değerleri mikrodenetleyici kartına gönderilmiştir. Otomatik kontrolde ise kullanıcıdan dört ayrı sayısal değer alınıp bunlar okuma verisi ile karşılaştırılmıştır. CMP elementi ile bu değerlerin büyüklüğü karşılaştırılıp eğer belli şartları sağlıyorsa belli komutların gönderilmesi sağlanmıştır. Bunun için de yine bir etkinleştirme anahtarı bulunup otomatik kontrolün çalışıp çalışmayacağı yine bir anahtarla belirlenmiştir.



Arayüzde seri haberleşme **COM6** portuna göre ayarlanmıştır. Arduino klon olduğu için CH341 sürücülerini yüklenmeli ve gerektiğinde aygıt yöneticisinden COM değeri değiştirilmelidir. “Okumaya Başla” düğmesine tıklandığında ekranda sıcaklık ve nem değerleri canlı olarak (sensör okuma zamanı ve 1000mS bekleme gecikmesi ile) gözlenmeye başlayacaktır. Manuel Kontrol anahtarı etkinleştirildiği zaman Isıtma Aç, Isıtma Kapat, Nemlendirme Aç, Nemlendirme Kapat düğmeleri çalışır hale gelecektir. Bunu devre dışı bırakmak manuel kontrol anahtarından önce çıkışları kapatmak tavsiye edilir. Otomatik kontrolde ise öncelikle ısıtma çıkışının etkinleştirileceği en düşük sıcaklık değeri, ısıtma çıkışının kapatılacağı en yüksek sıcaklık değeri, nemlendirme çıkışının etkinleştirileceği en düşük nem değeri ve çıkışın kapatılacağı en yüksek nem değeri belirlenip ondan sonra otomatik kontrol anahtarından etkinleştirilir.

Osilasyonu önlemek için en az sıcaklıkta en düşük ve en yüksek arası 2 derece, nemde %10 bir fark belirtilmesi gerekir.

