

Bir Soğutma Setinin Performansının Görselleştirilmesi

İlyas SEVİNÇ^{1*}, Arif Emre ÖZGÜR²

^{1*} Isparta Uygulamalı Bilimler Üniversitesi, Lisansüstü Eğitim Enstitüsü, Makine Mühendisliği ABD, Isparta, Türkiye, (ORCID: 0009-0007-0442-7000),
hnbiltek@gmail.com

^{2*} Isparta Uygulamalı Bilimler Üniversitesi, Teknoloji Fakültesi, Makine Mühendisliği ABD Bölümü, Isparta, Türkiye, (ORCID: 0000-0001-6382-5462),
emreozgur@isparta.edu.tr

(İlk Geliş Tarihi 21.02.2025 ve Kabul Tarihi 24.12.2025)

(DOI: 10.35354/tbed.1644443)

ATIF/REFERENCE: Sevinç, İ. & Özgür, A.E. (2026). Bir Soğutma Setinin Performansının Görselleştirilmesi. *Teknik Bilimler Dergisi*, 16(1), 21-30.

Öz

Bu çalışmada, R134a soğutkanı için doyma basıncı, doyma sıcaklığı, doymuş sıvı ve buhar entalpileri ile kızgın buhar faz durumu için sıcaklık, basınç ve entalpi değerleri Engineering Equation Solver (EES) programından yararlanılarak elde edilmiş, istatistiksel analiz yöntemleri kullanılarak R134a soğutkanına dair bir veri seti oluşturulmuştur. Bu veri setiyle P-h diyagramı çizdirilerek görselleştirilmiştir. Python programı kullanılarak R134a soğutkanı için makine öğrenmesi yöntemiyle bir eğitim modeli oluşturulmuştur. DS18B20 sıcaklık sensörleri Arduino Uno'ya bağlanmış ve programı yazılmıştır. Python ile Arduino arasında seri port üzerinden iletişim oluşturularak ve veriler alınmıştır. Deney seti üzerindeki basınç değerleri, manuel olarak, Tkinter arayüzüne girilmiştir. Makine öğrenmesi ile elde edilen eğitim modeline sıcaklık ve basınç değerleri yerleştirilerek entalpi değerleri hesaplanmış ve P-h diyagramı üzerine işaretlenerek birleştirilmiştir. Tkinter ara yüzünde, verileri sadece bir kez çalıştırabilme, on saniyede bir güncelleyebilme, güncellenen verileri durdurabilme, ekran görüntüsü alabilme, verileri Excel'e aktarabilme gibi fonksiyonlar da oluşturulmuştur.

Anahtar Kelimeler: R134a, P-h Diyagramı, Deney Seti, Python Makine Öğrenmesi, Arduino, Veri Görselleştirilmesi

Visualization of the Performance of a Cooling Set

Abstract

In this study, the saturation pressure, saturation temperature, saturated liquid and vapor enthalpies for the R134a refrigerant and the temperature, pressure and enthalpy values for the superheated vapor phase state were obtained using the Engineering Equation Solver (EES) program, and a data set for the R134a refrigerant was created using statistical analysis methods. The P-h diagram was drawn and visualized with this data set. Regression studies were performed for the R134a refrigerant using the Python program and a training model was created using the machine learning method. DS18B20 temperature sensors were connected to the Arduino Uno and the program was written. Communication was established between Python and Arduino via the serial port and data was received. The pressure values on the experimental set were read and entered into the Tkinter interface and included in the writing. The temperature and pressure values were placed in the training model obtained with machine learning, the enthalpy values were calculated and combined by marking on the P-h diagram. Functions such as being able to run data only once, updating it every ten seconds, stopping the updated data, taking screenshots, and transferring data to Excel have also been created in the Tkinter interface.

Keywords: R134a, P-h Diagram, Experiment Set, Python Machine Learning, Arduino, Data Visualization

* Sorumlu Yazar: hnbiltek@gmail.com

1. Giriş

Veri görselleştirme, ham verilerin çeşitli yazılımlar aracılığıyla analiz ederek tablo, diyagram, grafik, harita veya animasyonlar olarak sunulmasıdır. Buradaki amaç, öğrenmenin daha iyi ve kalıcı olabilmesi için ham verinin karşılığı olan görselleri kullanmaktır. İnsan beş duyu aracılığıyla her an bir şeyler öğrenme eğilimindedir. Araştırmalar, insanların ham verilerden oluşan veri setlerindeki ifadelerle kıyasla bu veri setinden oluşturulmuş görsellere bakarak 60.000 kat daha hızlı tepki verdiklerini ortaya koyarak veri görselleştirmenin önemini vurguluyor. [1]

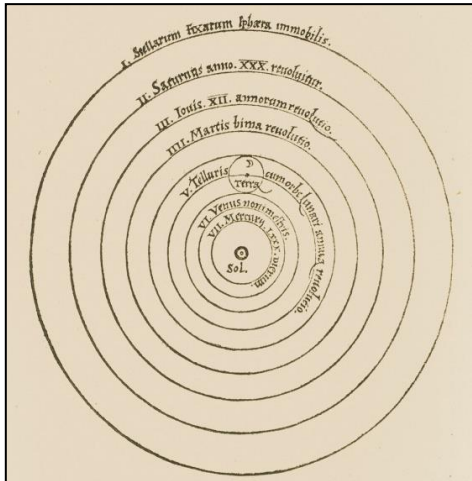
Tarihi Milattan önceki yıllara dayanan veri görselleştirme çalışmaları, ilk zamanlar taşlara veya mağara duvarlarına çizilen figürlerle başlamıştır. İnsanlar yaşadıkları dönemlerin özelliklerini sonraki nesillere bu şekilde aktarmayı başarabilmiştir. Günümüzde dahi bu figürlere bakarak, figürlerin çizildiği dönemle ilgili fikir sahibi olabilmekteyiz.

Milattan önceki döneme bakıldığında, UNESCO tarafından 2003 yılında Dünya Mirası olarak ilan edilen Bhimbetka Kaya Barınakları'nda; avcı toplayıcıların yaşamlarından sahneler mağara duvarlarına resmedilmiştir. Henüz pek çok hayvanın evcilleştirilmediği, insanların göçebe olarak yaşadığı ve medeniyetlerin henüz oluşmadığı Paleolitik Çağ'a ait görsel Şekil 1'de gösterilmiştir. [2]



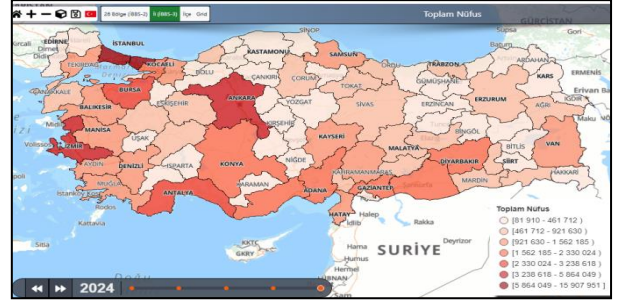
Şekil 1. Bhimbetka Kaya Barınakları

Milattan sonraki döneme bakıldığında, Güneş merkezli gök sisteminin kurucusu olan Nicolas Copernicus, 16. yüzyılda hazırlamış olduğu diyagramda, sırasıyla Merkür, Venüs, Dünya, Mars, Jüpiter ve Satürn gezegenlerinin güneş etrafında sabit hızda döndüğünü göstermiştir. [3]



Şekil 2. Copernicus'un Gezegenler Diyagramı

Günümüze gelindiğinde ise, bilim alanındaki gelişmeler endüstri devrimlerini gerçekleştirmiş, bilgisayarların ve internetin keşfiyle verileri toplama, analiz etme ve sonuçları tablo, grafik, diyagram, harita veya animasyon gibi veri görselleştirme araçları ile gösterme oldukça kolaylaşmıştır. Türkiye İstatistik Kurumu tarafından yapılan bir çalışmada, ülkemizdeki nüfusun illerimize göre dağılımını gösteren, aynı zamanda illerin nüfus yoğunluğunu farklı renklerde sembolize eden harita Şekil 3'de verilmiştir. [4]



Şekil 3. Türkiye Nüfusunun İllere Göre Dağılımı

Dedeoğlu [5], yapmış olduğu çalışmada bir resmin, birçok kelimenin yerini alabileceğinden bahsetmiştir.

Gürler [6], yapmış olduğu çalışmada eğitimde karmaşık ders konularının öğrencilerin zihninde tam olarak canlandırılmaması ve kavramların anlaşılmasından, ezber yönteminden ziyade görselliği yüksek ders materyallerinin kullanımının gerekliliğinden bahsetmiştir.

Bora [7], yapmış olduğu çalışmada verileri anlaşılabilir ve okunaklı hale getirerek, aktarılmak istenenlerin daha kolay yayılabilmesi için, insanların dikkat öncesi algı yeteneklerinden faydalanarak, bilgiyi kısa sürede özümseyebilmelerinden ve veri görselleştirmenin öneminden bahsetmiştir.

Günümüzde kullanılan deney setlerinde, soğutma teknolojilerinin temellerini gösterebilmek için basınç-sıcaklık ilişkisi, soğutma periyodu, ısı değiştiricinin fonksiyonu, basınç anahtarlarının ayarlanması, termostatik genleşme vanasının kullanılması gibi deneyler yapılabilmektedir. Değişik genleşme seçenekleriyle saydam evaporatörde soğutucu akışkanın buharlaşmasını gösterebilen S-801 Temel Soğutma Eğitim Seti Şekil 4'de gösterilmiştir. [8]



Şekil 4. S-801 Temel Soğutma Eğitim Seti

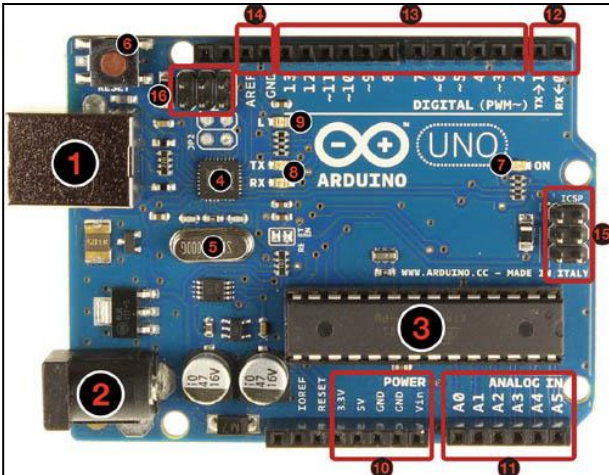
Üzerinde Solenoid valfler, basınç anahtarı ve göstergeleri, sıcaklık ölçüm noktaları, kompresör, multimetre, çeşitli fan ve butonların bulunduğu bir başka eğitim seti olan COK-0801S Temel Soğutma Eğitim Seti Şekil 5’de gösterilmiştir. [9]



Şekil 5. COK-0801S Temel Soğutma Eğitim Seti

Yurtiçi ve yurtdışı faaliyet gösteren firmaların deney setleri incelendiğinde temel soğutma çevrimi deneylerinin yapılabildiği fakat verilerin ham verilerden ibaret kaldığı, bilgisayar ya da tablet gibi bir ekranında veriler arasındaki ilişkinin gösterilmediği görülmektedir.

Arduino Uno Geliştirme Kartı, 6 tanesi PWM çıkışı olarak kullanılabilen 14 dijital giriş/çıkış pini, 6 analog giriş, bir adet 16 MHz kristal osilatörü, USB bağlantısı, power jakı, ICSP başlığı ve reset butonu gibi daha birçok özelliği bulunan, DC/AC bir adaptörle veya pille çalışabilen, ATmega328’e dayalı bir mikrodenetleyici karttır. Kütüphane yelpazesi oldukça geniş olan Arduino’nun tasarım ve yazılımları açık kaynaklı olduğundan birçok şirket çeşitli Arduino versiyonları üretebilmektedir. Arduino Uno Geliştirme Kartı’nın görseli Şekil 6’da verilmiştir. [10]

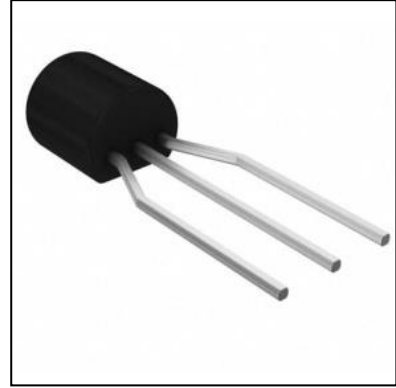


Şekil 6. Arduino Uno Geliştirme Kartı

Uz [11], yapmış olduğu çalışmada Arduino IDE uygulamasının, C ve C++ dilleri ile Arduino’ların programlanabildiğinden, açık kaynak kodlamayla sahip oldukça kullanışlı bir geliştirme ortamı olduğundan, metin tabanlı kod yazımı için uygun olduğundan ve farklı uygulamalar üzerinden blok tabanlı Arduino uygulamaları yapılabileceğinden bahsetmiştir.

Kaya [12], yapmış olduğu çalışmada Arduinoların, uygulamalarda kullanım kolaylığından, farklı sistemler tarafından kodlanmasının deseklendiğinden, çeşitli sensörlerle birlikte kullanılarak farklı projeler üretilenbileceğinden, çevreyle etkileşim kurabileceğinden ve elektronik mikroişlemci yapısına sahip olduğu için bilgisayar gibi kullanılabileceğinden bahsetmiştir.

DS18B20, 3 ile 5.5 arasındaki voltaj gerilimde çalışabilen, bünyesinde bulunan sıcaklık sensör devresi ile bulunduğu çevrenin sıcaklığını ölçebilen bir sensördür. Sıcaklık sensörünün bir görseline Şekil 7’de yer verilmiştir. [13]



Şekil 7. DS18B20 Sıcaklık Sensörü

Guido Van Rossum tarafından 1980’lerin sonunda bir hobi projesi olarak geliştirilen Python, öğrenmesi ve kullanımı kolay, etkileşimli, nesne yönelimli ve yüksek seviyeli, hata ayıklaması ve sözdizimi oldukça kolay açık kaynaklı bir programlama dilidir. Python, üzerinde barındırdığı kütüphane ve modüllerle veri madenciliği, veri analizi, veri görselleştirmesi gibi uygulamalarıyla mühendislik, istatistik, mobil uygulamalar, oyun geliştirme, etkileşimli web uygulamaları ve yapay zeka gibi alanlarda kullanım kolaylığı sağlamaktadır. Açık kaynaklı kodlamaya sahip olmasından dolayı tüm dünyadaki Python kullanıcıları tarafından hazırlanan dökümanlara ulaşmak oldukça kolaydır. Bu durum aynı zamanda problem çözümlerini de oldukça kolaylaştırmaktadır.

Sarı [14], yapmış olduğu çalışmada Python programının birçok yerde yoğun olarak kullanıldığından ve özellikle büyük veri kümelerinin hızlı bir şekilde analiz edilebilmesinde kolaylık sağladığından bahsetmiştir.

Bakdemir [15], yapmış olduğu çalışmada makine öğrenmesinin, bilgisayarların veri analitiği yaparak öğrenme yeteneği kazanmasını sağlayan yapay zekanın alt alanı olduğundan, kullanılan algoritmaların bir problemin çözümü için model oluşturarak ve verileri analiz ederek gelecek olan veriler için tahmin etme ya da karar verme yeteneğinden bahsetmiştir.

Bu çalışmada DS18B20 sıcaklık sensörleri ile elde edilen gerçek zamanlı ölçüm verileri Arduino Uno geliştirme kartıyla okunarak, seri port haberleşmesi yöntemiyle de veri görselleştirme yazılımına aktararak entalpi değerleri hesaplanıp grafik çizdirilebilecek ve oluşturulan görsel bir tablet ya da bilgisayar eşliğinde deney setlerine üzerine monte edilerek kullanılabilecektir. Oluşturulan veri görselleştirme yazılımı sayesinde, temel soğutma deney setlerini kullanan kişilerin, görselleştirilmiş

deney sonuçları ile ham verilerin arasındaki ilişkiyi daha hızlı anlayarak yorumlamaları hedeflenmiştir. Kullanılan DS18B20 sensörlerinin üretici verilerine göre ölçüm hassasiyeti ± 0.5 °C olup, Arduino okuma çözünürlüğü ve seri haberleşme süreci ölçüm belirsizliğine katkı sağlamaktadır.

2. Materyal ve Metot

Arduino Uno Geliştirme Kartı, bu programda OneWire.h ve DallasTemperature.h kütüphaneleri kullanılmaktadır. DS18B20 sıcaklık ölçme sensörü 2 nolu pine bağlanmıştır. OneWire ve DallasTemperature nesnelerini oluşturulmuştur. Seri port saniyede maksimum 9600 bit veri aktarabilecek şekilde “9600 baud” hızında ayarlanarak seri port ekranına DS18B20 sıcaklık sensöründen saniye de bir kez veri gelecek şekilde programlanmış ve Şekil 8’de gösterilmiştir.

```
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 2
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
void setup() {
  Serial.begin(9600);
  sensors.begin();
}
void loop() {
  sensors.requestTemperatures();
  for (int i = 0; i < sensors.getDeviceCount(); i++) {
    Serial.println(sensors.getTempCByIndex(i));
  }
  delay(1000);
}
```

Şekil 8. Arduino Uno’da Yazılan Kodlar

Jupyter Notebook, açık kaynak kodlu bir programdır. Çeşitli programlama dilleri için etkileşimli bir ortam sağlamasından dolayı günümüzde sıklıkla tercih edilmektedir. Bilgisayara Python kurulumu tamamlandıktan sonra Windows işletim sisteminde komut istemi CMD (Command Prompt) çalıştırılarak Şekil 9’daki kod satırı yazılmış ve Jupyter Notebook kurulumu yapılmıştır.

```
pip install jupyter
```

Şekil 9. Jupyter Notebook Kurulumu

Jupyter Notebook projesinde kullanılmak üzere Şekil 10’daki kütüphaneler projeye yüklenmiştir.

```
import glob
from pylab import *
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
%matplotlib notebook
```

Şekil 10. Jupyter Notebook Kütüphanelerin Yüklenmesi

Python ile verileri analiz etmek için Şekil 11’de görüldüğü gibi Pandas Kütüphanesi altındaki DataFrame nesnesi kullanılmış ve Excel verileri de diziler yardımıyla projeye yüklenmiştir.

```
NewData = pd.DataFrame()
for f in glob.glob(r"C:\Users\asus\Desktop\jupyter\PH.xlsx*"):
    df = pd.read_excel(f, decimal=",")
    NewData = pd.concat([NewData, df], ignore_index = True, axis = 0)
```

Şekil 11. Jupyter Notebook’a Excel Verilerinin Yüklenmesi

Şekil 12’de grafiği çizdirmek ve P-h diyagramının Basınç (P) kısmını logaritmik yapabilmek için gereken komutlar gösterilmiştir.

```
plt.figure(figsize=(8,6))
plt.plot(dx1,dy1,'b-')
plt.plot(dx3,dy3,'b-')
plt.plot(dx2,dy2,'r-')
plt.yscale("log")
plt.show()
```

Şekil 12. Jupyter Notebook’ta Grafik Çizilmesi

Projede kullanılan R134a akışkan gazına ait veriler T (Sıcaklık/°C) , P (Basınç/kPa) ve h (entalpi/(kJ/kg)), EES programı kullanılarak Excel formatında kayıt edilmiş, katsayıları bulunması gereken değişkenler ve çıkması gereken gerçek değere ait veriler Şekil 13’deki komutlarla yüklenmiştir.

```
df = pd.read_excel('R134a.xlsx')
X = df[['T (girdi)', 'P (girdi)']]
y = df['h (çıktı)']
```

Şekil 13. R134a Akışkan Gazına Ait Verilerin Yüklenmesi

Makine öğrenmesinde en çok kullanılan Python paketlerinden Scikit-Learn paketi, Linear Regression özelliği ile orijinal veri kümemizi modellememize yardımcı olmuştur. Test bölmemizi gerçekleştirebilmek ve eğitim verileri oluşturmak için Şekil 14’deki kodlar yazılmıştır.

```
X_train,X_test, y_train, y_test = train_test_split(X, y,
    test_size = 0.3, random_state = 5)
y_train.shape
y_test.shape
reg_model = linear_model.LinearRegression()
reg_model = LinearRegression().fit(X_train, y_train)
print('Intercept: ', reg_model.intercept_)
list(zip(X, reg_model.coef_))
```

Şekil 14. R134a Akışkan Gazına Ait Verilerin Modellenmesi

Şekil 15’deki komut yazılarak gerçek ve tahmini değerler arasındaki ilişki gözlenmiştir.

```
reg_model_diff = pd.DataFrame({'Gerçek Değer':
    y_test, 'Tahmini Değer': y_pred})
reg_model_diff
```

Şekil 15. Gerçek ve Tahmini Değerler İçin Yazılan Komut

Python da grafik kullanıcı arayüzü (GUI) olarak Tkinter, PyQt, PyGI ve Kivy gibi modüller bulunmaktadır. Tkinter, Python kurulumu ile birlikte gelen, pencere ve menü gibi çeşitli nesneleri tasarlamaya olanak sağlayan bir modül olmasından dolayı tercih edilmiştir. Tkinter Windows, MacOS ve Linux gibi işletim sistemlerinde çalışabilmektedir. Programın arayüzü tasarımı için Şekil 16’daki kodlar yazılmıştır.

```

master=Tk()
canvas=Canvas(master, height=800, width=1000)
canvas.pack()
f_ust1=Frame(master, bg='#add8e6')
f_ust1.place(relx=0.05, rely=0.05, relwidth=0.10, relheight=0.132)
f_ust=Frame(master, bg='#add8e6')
f_ust.place(relx=0.15, rely=0.05, relwidth=0.8, relheight=0.132)
f_alt_sol1=Frame(master, bg='#add8e6')
f_alt_sol1.place(relx=0.05, rely=0.21, relwidth=0.23, relheight=0.11)
f_alt_sol2=Frame(master, bg='#add8e6')
f_alt_sol2.place(relx=0.05, rely=0.33, relwidth=0.23, relheight=0.43)
f_alt_sol3=Frame(master, bg='#add8e6')
f_alt_sol3.place(relx=0.05, rely=0.77, relwidth=0.23, relheight=0.12)
f_alt_sag=Frame(master, bg='#add8e6')
f_alt_sag.place(relx=0.29, rely=0.21, relwidth=0.66, relheight=0.68)

```

Şekil 16. Tkinter Arayüz Tasarımı

Tkinter arayüzünde, Değerleri Gönder butonuna tıklandığında, girilen basınç değerlerinin yazılıma gönderilmesi, yapılan hesaplarla entalpi değerlerinin belirlenmesi, bu değerlere göre de P-h diyagramının çizdirilmesi, Güncelle butonuna tıklandığında, bu işlemlerin on saniye de bir olacak şekilde devam etmesi, Durdur butonuna tıklandığında, güncellenen grafiklerin durdurularak sabit olarak ekranda son grafiğin kalması, Ekran Görüntüsü Al butonuna tıklandığında son grafik ekrandayken görünümün ekran görüntüsü alınarak programın çalıştığı klasör içerisine kaydedilmesi, Temizle butonuna tıklandığında ara yüzdeki grafiği yok ederek programın başa dönmesi, Excel'e Aktar butonuna tıklandığında, tüm verilerin Excel çalışma sayfasına aktarılarak kaydedilmesi ve son olarak Çıkış Yap butonuna tıklandığında programdan çıkış yapılması şeklinde çalışan yazılım oluşturulmuştur.

Python programlama dilinin Arduino Uno ile haberleşebilmesi için Şekil 17'deki kod bloğu yazılmıştır.

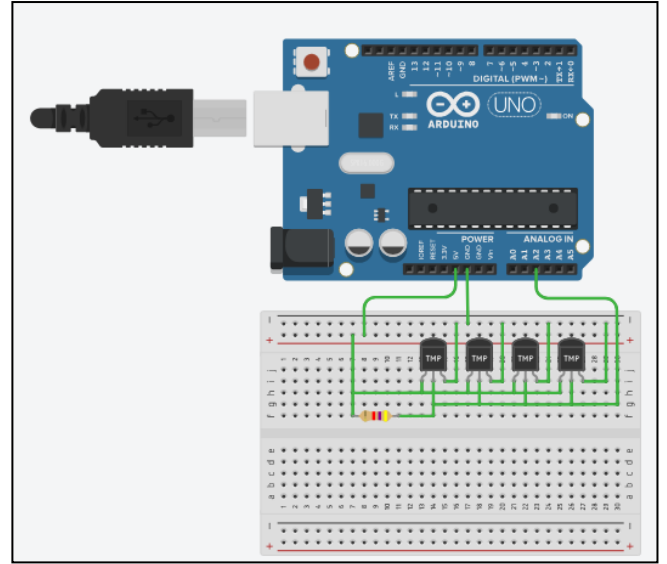
```

def t_oku():
    ports = list(serial.tools.list_ports.comports())
    for p in ports:
        pass
    ser = serial.Serial(port=p[0], baudrate=9600, timeout=2)

```

Şekil 17. Seri Port İletişiminin Kurulması

Arduino Uno üzerine dört adet DS18B20 sıcaklık sensörü ile bir adet 4.7 k Ω direnç Tincarcad programı üzerinde bağlanarak Şekil 18' de gösterilmiştir. Kullanılan DS18B20 sensörlerinin üretici verilerine göre ölçüm hassasiyeti ± 0.5 °C olup, Arduino okuma çözünürlüğü ve seri haberleşme süreci ölçüm belirsizliğine katkı sağlamaktadır.



Şekil 18. Arduino Uno Ve Sensörlerin Bağlantı Şekli

3. Araştırma Sonuçları ve Tartışma

Yapılan bu çalışmada deney seti üzerinde farklı noktalara sıcaklık sensörleri bağlanmış, Arduino Uno'ya bu sıcaklık sensörlerinden gelen veriler gerçek zamanlı olarak aktarılmıştır. Arduino Uno ile bilgisayar arasında bağlantı yapılarak, sensör verileri seri port üzerinden Python programına aktarılmıştır.

EES (Engineering Equation Solver) programından elde edilen veriler ile R134a soğutucu akışkanı (soğutkan) için doyma basıncı, doyma sıcaklığı, doymuş sıvı ve buhar entalpileri ile kızgın buhar faz durumu için sıcaklık, basınç ve entalpi değerleri elde edilerek Excel ortamında kaydedilmiştir. Bu veri seti Python programına aktarılmış, P-h diyagramı çizdirilmiş, regresyon çalışmaları yapılmış ve makine öğrenmesi yöntemiyle bir eğitim modeli oluşturulmuştur.

Eğitim modeli için hazırlanan gerçek ve tahmini değerler arasındaki ilişki Şekil 19'da gösterilmiştir.

	Gerçek Değer	Tahmini Değer
1377	376.8	368.693208
184	281.5	276.332444
680	338.1	339.083154
723	320.7	321.946117
973	317.8	319.677475
...	...	...
672	295.2	296.744803
1558	364.1	355.209900
1631	332.8	330.260257
1200	369.8	365.039478
687	375.7	376.129211

Şekil 19. Gerçek Ve Tahmini Değerler Arasındaki İlişki

Uygulanan Linear regression sonucunda modelimiz için oluşturulan katsayılar Şekil 20'deki gibi olmuştur. Bu doğrusal regresyon modeli, yalnızca EES verilerinin kapsadığı sınırlı sıcaklık ve

basınç aralığında geçerlidir ve bu aralığın dışında kullanılmamalıdır.

Intercept: 243.0668542070619

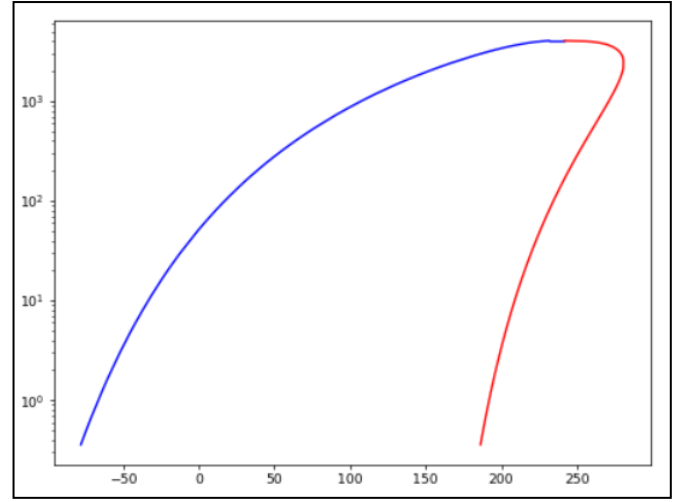
[('T (girdi)', 1.0584587763151974), ('P (girdi)', -0.012601559505754176)]

Şekil 20. Eğitim Modelinin Katsayıları

Makine öğrenmesi yöntemiyle elde edilen katsayılarla birlikte Python programında kullanılan formül kısaca aşağıdaki gibi olmuştur.

$$H=243.0668 + T * 1.0584 + P * (-0.0126)$$

Şekil 21'de program çalıştırıldığında bu verilerden yola çıkılarak doyma eğrileri için çıkan sonuç gösterilmiştir.



Şekil 21. Basınç-Entalpi Grafiği Doyma Eğrileri

Programın arayüzü Şekil 22'de görüldüğü gibi Tkinter'da tasarlanmıştır.



Şekil 22. Tkinter'da Arayüz Tasarımı

h_4 noktası h_3 'e eşit kabul edilmiştir. Bu kabul, genleşme vanasının izoentalpik çalıştığı varsayımına dayanmaktadır ve temel soğutma çevrimi analizlerinde yaygın olarak kullanılan bir yaklaşımdır.

Soğutma etkinlik katsayısı (COP), literatürde kabul edilen termodinamik tanıma uygun olarak, evaporatörde çekilen ısıнын kompresör işine oranı

şeklinde tanımlanmıştır:

e-ISSN: 2146-2119

$$COP = \frac{(h_1 - h_3)}{(h_2 - h_1)}$$

Evaporatördeki soğutma kapasitesi (q_g) hesaplanırken;

$$q_g = (h_1 - h_4)$$

Kompresörün güç girişi (w_k) hesaplanırken;

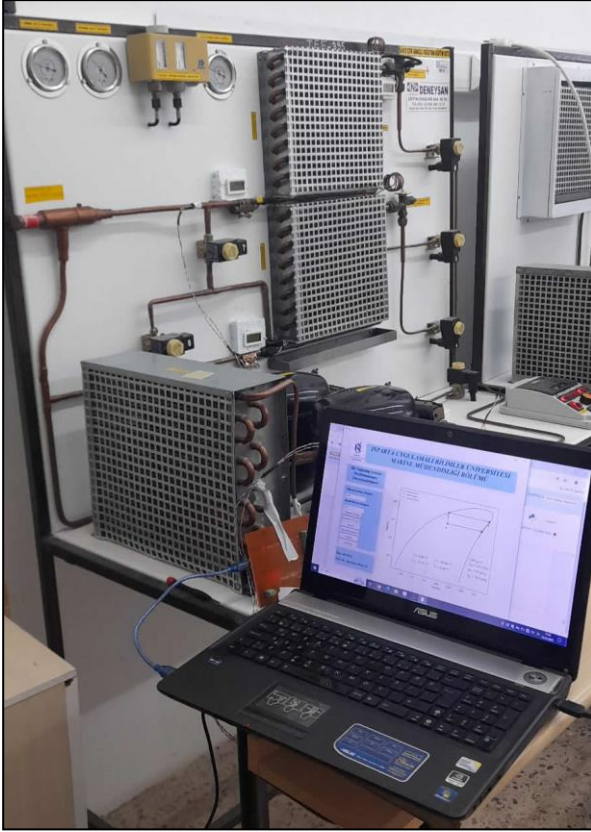
$$w_k = (h_2 - h_1)$$

Kondenser kapasitesi (q_y) hesaplanırken;

$$q_y = (h_2 - h_3)$$

formülleri kullanılmıştır.

Deney setine bilgisayar bağlandıktan sonra, programın çalıştırılması, sensörlerden gelen sıcaklık verilerinin Arduino Uno üzerinden programa aktarılması, deney seti üzerindeki basınç değişimlerinin gözlemlenmesi ve Tkiner ara yüzüne girilmesi, bu veriler ışığında yapılan deneylerin eğitilen model üzerine işlenerek entalpi değerlerinin hesaplanması, entalpi değerlerinin grafik üzerinde işaretlenmesi ve son olarak bu işaretlerin birleştirilmesinden oluşan deneyler yapılmıştır. Yapılan bu deneylerden birinin görüntüsü Şekil 23’de gösterilmiştir.



Şekil 23. Deney Setinin Python Programı İle Kullanımı

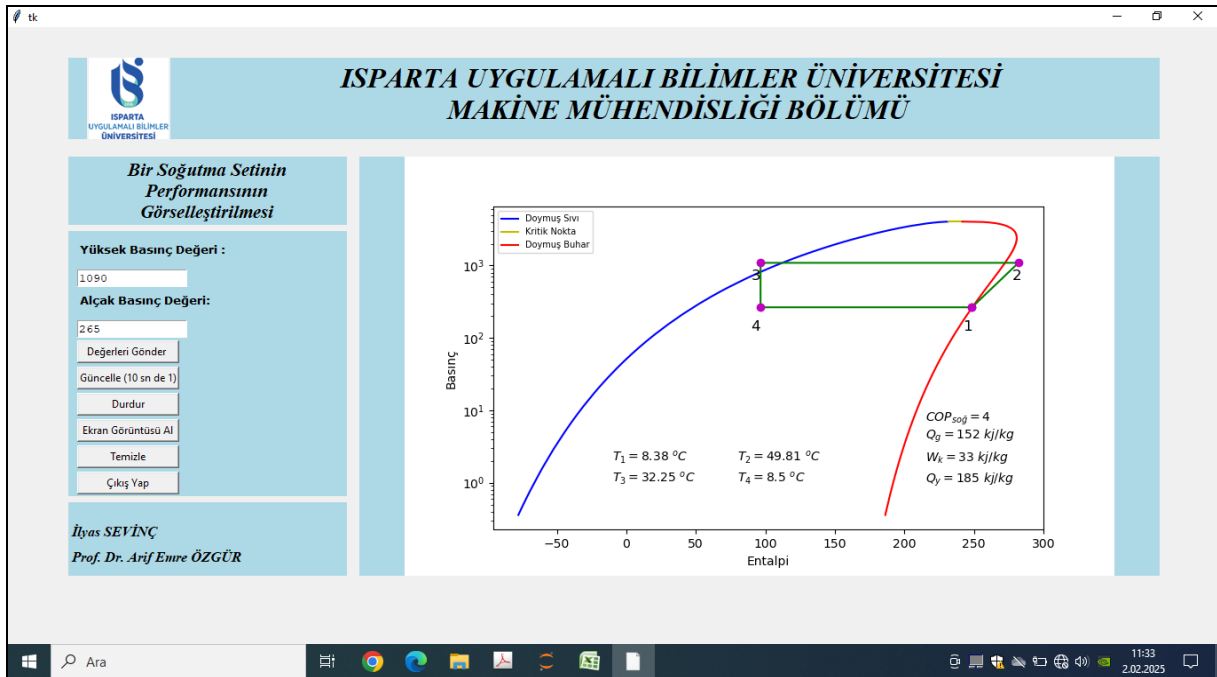
Yapılan deneylerde her on saniye de bir sıcaklık verilerinin yazılıma gelmesi sağlanmış, bu değerlere ve okunan basınç değerlerine göre grafiğin güncellendiği gözlemlenmiştir. Bu deneylerden bazıları aşağıda verilmiştir. DS18B20 sıcaklık sensörlerinden gelen veriler;

$$\begin{aligned} T_1 &= 8.38 \text{ }^{\circ}\text{C} \\ T_2 &= 49.81 \text{ }^{\circ}\text{C} \\ T_3 &= 32.25 \text{ }^{\circ}\text{C} \\ T_4 &= 8.5 \text{ }^{\circ}\text{C} \end{aligned}$$

Deney seti üzerinde okunan basınç değerleri;

Yüksek Basınç Değeri: 1095 kPa
Alçak Basınç Değeri: 268 kPa

şeklinde olduğunda Şekil 24’deki grafik ortaya çıkmıştır.



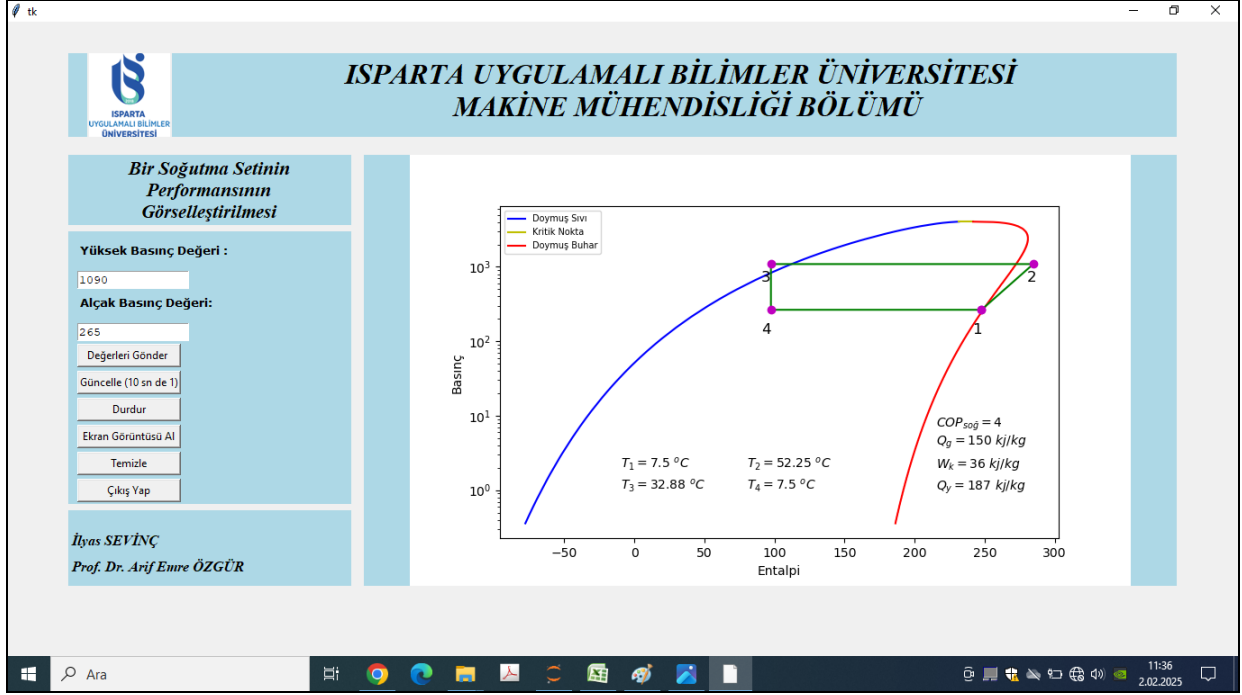
Şekil 24. Veri Görselleştirme Deneyi-1

$T_2=52.25\text{ }^{\circ}\text{C}$
 $T_3=32.88\text{ }^{\circ}\text{C}$
 $T_4=8.0\text{ }^{\circ}\text{C}$

Yüksek Basınç Değeri: 1096 kPa
 Alçak Basınç Değeri: 270 kPa

Deney seti üzerinde okunan basınç değerleri;

şeklinde olduğunda Şekil 25'deki grafik ortaya çıkmıştır.



Şekil 25. Veri Görselleştirme Deneyi-2

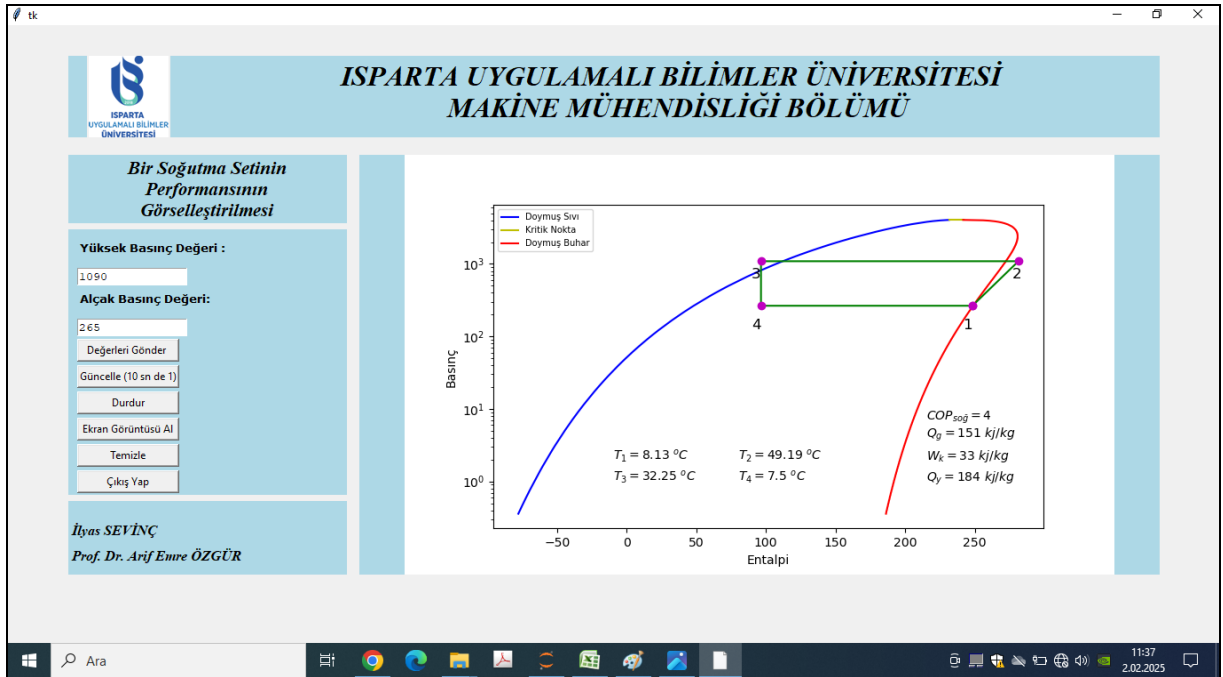
DS18B20 sıcaklık sensörlerinden gelen veriler;

Deney seti üzerinde okunan basınç değerleri;

$T_1=8.13\text{ }^{\circ}\text{C}$
 $T_2=49.19\text{ }^{\circ}\text{C}$
 $T_3=32.25\text{ }^{\circ}\text{C}$
 $T_4=8.13\text{ }^{\circ}\text{C}$

Yüksek Basınç Değeri: 1090 kPa
 Alçak Basınç Değeri: 265 kPa

şeklinde olduğunda Şekil 26'daki grafik ortaya çıkmıştır.



Şekil 26. Veri Görselleştirme Deneyi-3

Tkinter ara yüzünde Excel'e aktar butonuna tıklandığında ölçülen tüm sıcaklık değerleri (T_1, T_2, T_3, T_4), hesaplanan entalpi değerleri (h_1, h_2, h_3, h_4), soğutma etkinlik katsayısı (COP), e-ISSN: 2146-2119

evaporatördeki soğutma kapasitesi (q_g), kompresörün güç girişi (w_k) ve kondenser kapasitesi (q_y) değerleri Şekil 27'deki Excel dökümanına aktarılmıştır.

	A	B	C	D	E	F	G	H	I	J	K	L
	T1 (°C)	T2 (°C)	T3 (°C)	T4 (°C)	h1	h2	h3	h4	Cop	Qg (kj/kg)	Wg (kj/kg)	Qy (kj/kg)
1	8,38	49,81	32,25	8,5	248,5595	281,99	96,49104	96,49104	4.55	152.07	33.43	185.5
2	8,13	49,19	32,25	8,13	248,3327	281,3967	96,5094	96,5094	4.59	151.82	33.06	184.89
3	8,25	51,33	32,3	8,7	248,4219	283,6114	96,56684	96,56684	4.32	151.86	35.19	187.04
4	8,41	53,1	32,35	8,78	248,5787	285,4723	96,64264	96,64264	4.12	151.94	36.89	188.83

Şekil 27. Deney Sonuçlarının Excel’de Görünümü

4. Sonuç

Bu çalışmada, EES programından elde edilen veriler ile P-h diyagramı çizdilebilmiştir. Makine öğrenmesi yöntemiyle elde edilen eğitim modeli üzerine işlenen basınç ve sıcaklık verileri ile entalpi değerleri hesaplanabilmiş, P-h diyagramında noktasal olarak işaretlenebilmiş ve noktalar birleştirilerek ver görselleştirilmesi deneyleri yapılabilmektedir. Ara yüz programındaki tüm fonksiyonlar kullanılabilmektedir. Güncelle seçeneği ile veriler her on saniye de bir güncellenerek grafik yenilenmiştir. Programa gelen T_1, T_2, T_3, T_4 sıcaklık verileri, soğutma etkinlik katsayısı (COP), evaporatördeki soğutma kapasitesi (q_g), kompresörün güç girişi (w_k) ve kondenser kapasitesi (q_y) değerleri de, veriler her geldiğinde güncellenmektedir. Elde edilen bulgular, geliştirilen sistemin eğitim amaçlı soğutma laboratuvarlarında kullanılabilir olduğunu ve öğrencilerin termodinamik büyüklükler arasındaki ilişkileri daha etkin biçimde kavrayabildiğini göstermektedir.

Bu çalışmada, DS18B20 sıcaklık sensörleri ile elde edilen gerçek zamanlı ölçüm verileri Arduino Uno geliştirme kartıyla okunmuş, seri port haberleşmesi yöntemiyle de veri görselleştirme yazılımına aktarılıp entalpi değerleri hesaplanarak grafik çizdirilebilmiştir. Çalışmada oluşturulan görsel bir tablet ya da bilgisayar eşliğinde deney setlerine üzerine monte edilerek kullanılabilecektir. Kullanılan DS18B20 sensörlerinin üretici verilerine göre ölçüm hassasiyeti ± 0.5 °C olup, Arduino okuma çözünürlüğü ve seri haberleşme süreci ölçüm belirsizliğine katkı sağlamaktadır.

Kaynakça

[1] Veri Görselleştirme Nedir? <https://www.oracle.com/tr/busi>

ness-analytics/what-is-data-visualization/ (Son erişim tarihi: 1 Şubat 2025)

[2] Bhimbetka Kaya Barınakları. <https://www.unescodunya.com/hindistan/bhimbetka-kaya-barinaklari/> (Son erişim tarihi: 1 Şubat 2025)

[3] Copernicus'un Gezegenler Diyagramı. https://farm6.staticflickr.com/5311/7368885570_b434dd261a_o.jpg (Son erişim tarihi: 1 Şubat 2025)

[4] Türkiye Nufusunun İllere Göre Dağılımı. <https://cip.tuik.gov.tr> (Son erişim tarihi: 1 Şubat 2025)

[5] Dedeoğlu Ş., (2015). Bilgilendirme Tasarımında Veri Görselleştirmesinin Yeri Ve Kullanımı. Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi.

[6] Gürler A., (2018). Veri Görselleştirme Ve Bilgilendirme Grafikleri. Yüksek Lisans Tezi, Kahramanmaraş Sütçü İmam Üniversitesi.

[7] Bora E., (2023). Kentsel Ölçekte Veri Görselleştirmenin Önemi Ve Ankara'daki Çevresel Sorunlara Ait Bir Uygulama Projesi. Yüksek Lisans Tezi, Hacettepe Üniversitesi.

[8] S-801 Temel Soğutma Eğitim Seti. <https://www.argemsan.com/egitim/makine-muhendisligi-egitim-setleri/sogutma-sistemleri/s-801-temel-sogutma-egitim-seti> (Son erişim tarihi: 2 Şubat 2025)

[9] COK-0801S Temel Soğutma Eğitim Seti. <https://www.cokesen.com/product/cok-0801s-temel-sogutma-deney-seti/> (Son erişim tarihi: 2 Şubat 2025)

[10] Arduino Uno Özellikleri. https://www.robotiksisitem.com/arduino_uno_ozellikleri.html (Son erişim tarihi: 2 Şubat 2025)

[11] Uz M., (2022). Arduino İle Engelleri Kaldırılım. Yüksek Lisans Tezi, Karamanoğlu Mehmetbey Üniversitesi.

[12] Kaya M., (2022). Fen Bilimleri Eğitiminde Arduino Uygulama Süreçlerine Yönelik Öğretmen Adaylarının

Görüşlerinin İncelenmesi. Yüksek Lisans Tezi, Çukurova Üniversitesi.

- [13] DS18B20 Sıcaklık Sensör Entegresi To-92. <https://www.direnc.net/ds18b20-sicaklik-sensor-entegresi-to-92> (Son erişim tarihi: 2 Şubat 2025)
- [14] Sarı C,. (2023). Python Programlama Dili Kullanılarak Günlük Maksimum Yağış Verilerinin Trend Ve Risk Analizleri. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi.
- [15] Bakdemir M,. (2024). Python İle Makine Öğrenmesi Algoritmalarını Kullanarak Rüzgar Santralinin Elektrik Üretim Tahmini. Yüksek Lisans Tezi, Ordu Üniversitesi.