**Rasperry Pi Kullanarak Uygun Hava Şartlarını Yakalama**

**Giriş**

Kapalı alanlarda gerçekleştirilen toplu organizasyonlarda CO2 ve sıcaklığın artması sonucunda insanların dikkat dağınıklığı ve uyuklama sorunu yaşamaması için ortamdaki CO2 yoğunluğu arttığı zaman sensörlerle algılanıp raspberry pi sayesinde havalandırma sistemini çalıştırarak CO2 miktarını en uygun seviyeye indirmek. Aynı zamanda sıcaklığın ısı sensörleri sayesinde algılanarak ortamın sıcaklığını optimum düzeyde tutulmasıdır.. Raspberry pi 3, Arduino, MQ-6 gaz sensörü, DHT11 sıcaklık ve nem sensörü kullanılarak bu sistem tasarlanmıştır.

Kodlarımız github’a yüklenmiştir.

<https://github.com/tunahan12>

**Gerekli Donanım Bileşenleri**

**1.** 1 adet Rasperry pi

**2.** 1 adet Arduino

**3.** 1 adet MQ-6 gaz sensörü

**4.** 1 adet DHT11 sıcaklık ve nem sensörü

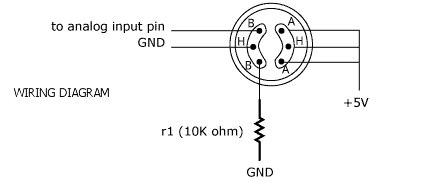
**NOT:** Çıkışlar havalandırma ve soğutma sistemini temsilen 2 ayrı motor olarak konulduğu için donanım bileşenlerine eklenmemiştir.

**Gerekli Yazılım Bileşenleri**

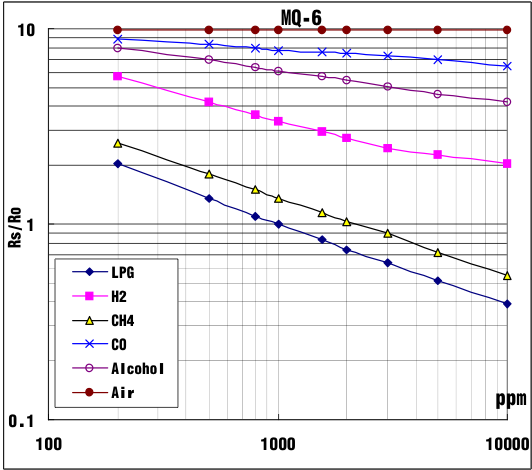
1. Python
2. C Dili

**Kullanılan Bileşenlerin Özellikleri**

1. Rasperry pi 3;Raspberry Pi’nin 2018 Pi Günü (14 Mart) çıkmış olan en güncel sürümüdür. Fiziksel olarak kartın boyutu ve yerleşimi B+/2/3 sırasını takip edecek şekilde değişmeden aynı formunu korumuş, buna karşılık işlemcisi ve kablolu/kablosuz ağ bağlantı hızı güncellenmiştir. Pi 3B+ üzerinde yer alan gigabit Ethernet kontrolcü sayesinde eskisine göre 3 kat daha hızlı ve PoE (Power-over-Ethernet) destekli kablolu; çift band 802.11ac kablosuz bağlantı sayesinde ise kablosuz bağlantıda 2 kat hızlı bağlantı desteği sunulmaktadır. Ayrıca Bluetooth desteği de Pi 3’te sağlanan Bluetooth BLE 4.1’den BLE 4.2’ye terfi etmiştir.
2. **Arduino Uno;**ATmega328 mikrodenetleyici içeren bir Arduino kartıdır. Arduino 'nun en yaygın kullanılan kartı olduğu söylenebilir. Arduino Uno 'nun ilk modelinden sonra Arduino Uno R2, Arduino Uno SMD ve son olarak Arduino Uno R3 çıkmıştır. Arduino 'nun kardeş markası olan **Genuino** markasını taşıyan **Genuino Uno** kartı ile tamamen aynı özelliklere sahiptir. Arduino Uno 'nun 14 tane dijital giriş / çıkış pini vardır. Bunlardan 6 tanesi PWM çıkışı olarak kullanılabilir. Ayrıca 6 adet analog girişi, bir adet 16 MHz kristal osilatörü, USB bağlantısı, power jakı (2.1mm), ICSP başlığı ve reset butonu bulunmaktadır. Arduino Uno bir mikrodenetleyiciyi desteklemek için gerekli bileşenlerin hepsini içerir. Arduino Uno 'yu bir bilgisayara bağlayarak, bir adaptör ile ya da pil ile çalıştırabilirsiniz.
3. MQ-6; LPG/İzobütan/Propan Gaz Sensör; MQ-6 LPG sensörü izobütan ve 300ppm ve 10.000ppm konsanstrasyonlarda propan algılar. 10.000ppm ve 300ppm aralığında algılama yapabilmek gaz kaçağı için uygundur.  H2, LPG, CH4, CO, Alkol, Duman veya Propan tespiti için uygundur .

****

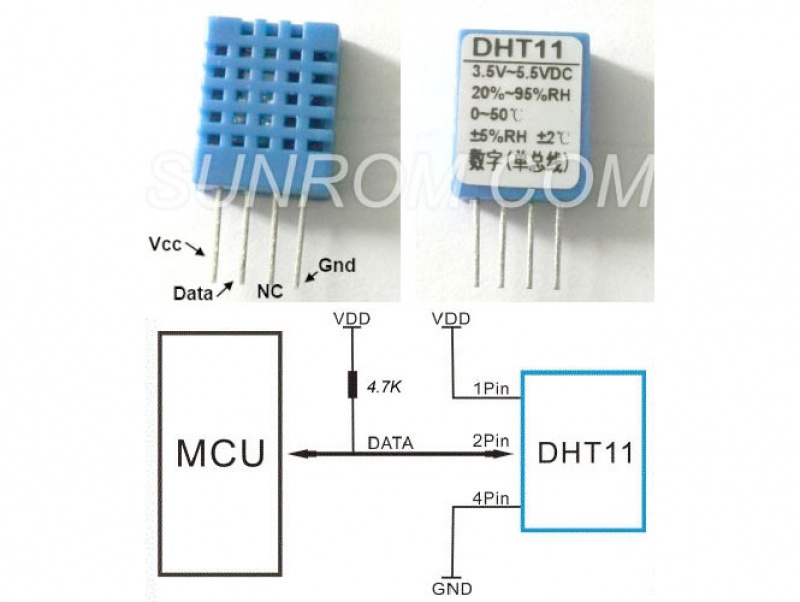
MQ-6 Pin\_Out 1



MQ-6 Okunan\_Gazlar 1

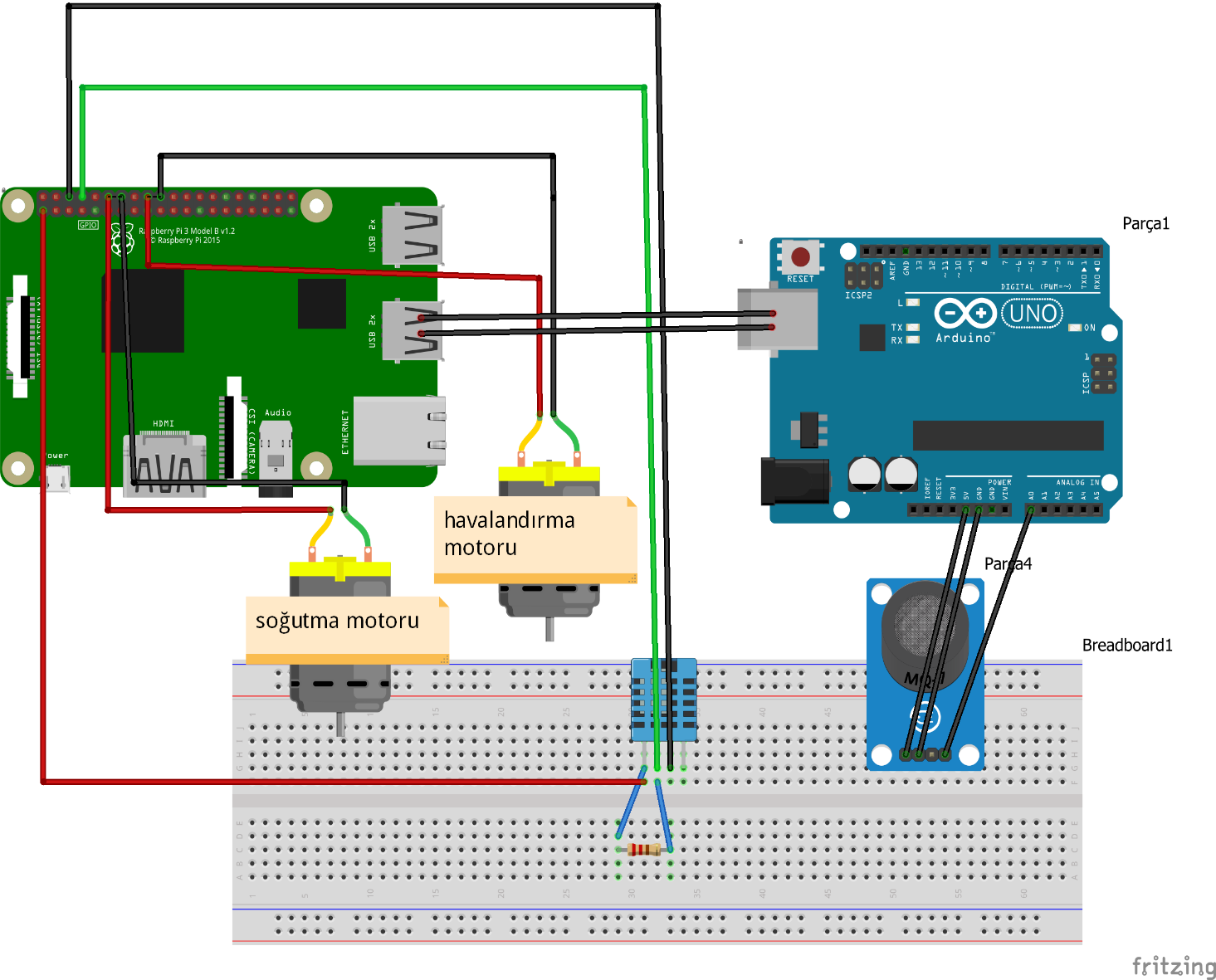
**NOT:**Dış ortamdaki CO2 seviyesi yaklaşık olarak 250 ppm (parts per million)’dir.bu nedenle oda koşulları için tüm kodlarımızda 300 pmm seviyesi gaz sensörü için çıkış sinyali üretmesi için gerekli olan değer olarak seçilmiştir.

1. DHT11 ;Isı ve Nem Sensör Kartı, üzerinde DHT11 sensörü bulunan, bağlantıları çekilip breadboard veya farklı kullanımlar için kolaylaştıtılmış hale sokulmuş modüldür. DHT11 sıcaklık ve nem algılayıcı kalibre edilmiş dijital sinyal çıkışı veren gelişmiş bir algılayıcı birimidir. Yüksek güvenilirliktedir ve uzun dönem çalışmalarda dengelidir. 8 bit mikroişlemci içerir, hızlı ve kaliteli tepki verir. 0 ile 50°C arasında 2°C hata payı ile sıcaklık ölçen birim.

****

DHT-11 Pin\_Out 1

**Şematik Çizimi:**



Şematikçizim.png

.

**Yapım Aşamaları**

Raspberry Pi vakfının 4.yılında çıkarmış olduğu Raspberry Pi 3 modeli kablosuz ağ (wifi) ve bluetooth özelliği ile son kullanıcıya fazlasıyla hitap etmeyi başardı. Raspberry Pi’yi ilk kez kullanacaklar için hazırladığımız bu yazıda SD Kart’a Raspbian İşletimi Sistemi'nin kurulumu anlatacağız.

1. **Raspbian İşletim Sistemi İndirme;**

**Raspbain Jessie** ve **Raspbian Jessie Lite** olmak üzere iki farklı dosya göreceksiniz. **Lite** olan sürümde grafik ekran arayüzü olmadığı için Raspberry Pi 3’ün HDMI çıkışından görüntü alamıyoruz. Bu sürüm ileri seviye kullanıcıların Raspberry Pi’yi komut satırından (terminalden) kullanmaları için hazırlanmıştır. Siz de ekran görüntüsüne ihtiyaç duymayacak proje hazırlamak isterseniz **Lite**sürümünü kullanabilirsin.

1. **İşletim Sistemini Micro-SD Kart’a Yazdırma**

İndirdiğimiz imaj dosyasını zip içerisinden çıkarıyoruz. Ardından daha önce indirdiğimiz win32diskImager programını açıyoruz. İmaj dosyamızı belirtilen yerden seçiyoruz.

Sd kartınızın bilgisayara takılı olduğundan emin olduktan sonra **Device** kısmında görebilirsiniz. Ardından Write butonuna tıklayıp yazma işlemini başlatıyoruz. Yazma işlemi yaklaşık 2-3 dk sürmektedir. Yazma işleminin bitmesini yeni açılan pencerede "**Write Succesful.**" yazısını görene kadar bekleyiniz.

1. **Bağlantılar ve Çalıştırma**

Şimdi sıra bir **Raspberry Pi’**ye hayat vermeye geldi. Önce SD Kartı bilgisayardan çıkarıp Pi 3’e takıyoruz.

Raspberry Pi’yi direk olarak HDMI bir ekrana bağlıyoruz. Elimizde HDMI girişi olan bir ekran yoksa VGA girişli bir ekran da kullanabiliriz. Bu durumda **HDMI to VGA** dönüştürücü kullanmamız gerekecek.

Son olarak adaptör ile cihazımıza enerji veriyoruz.

**SENSÖRLERDEN DEĞER OKUMAK İÇİN**

İlk olarak Raspberry Pi'deki Raspbian OS'yi güncellemek için aşağıdaki komutu çalıştırın:

sudo apt-get update

**Sıcaklık ve Nem Sensörü İçin Adımlar:**

**1. Adım:** Aşağıdaki linki kullanarak "**DHT11**" Kitaplığı'nı yükleyin:

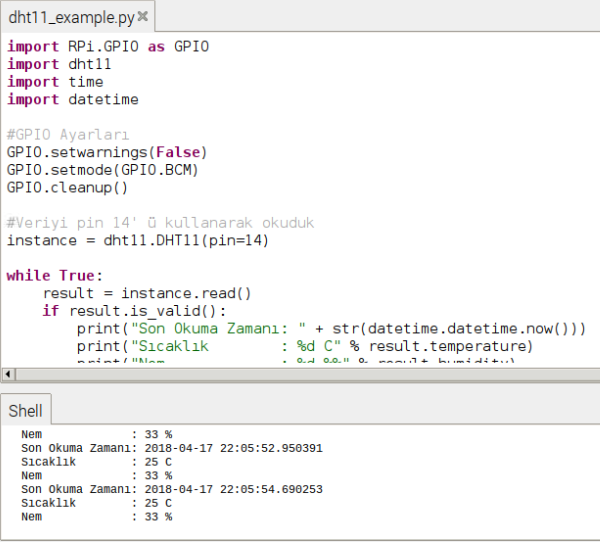
<https://github.com/tunahan12/OTOMATiK-HAVALANDIRMA-VE-ISITMA-SiSTEMi/tree/master/SICAKLIK%20VE%20NEM%20SENS%C3%96R%C3%9C>

**2. Adım:** Bu işlem sonucunda kullanacağımız örnek kod ve kütüphane, **DHT11\_Python**isimli bir klasöre inmiş oluyor. Bu klasördeki “ DHT11\_example.py “dosyasını çalıştırınız.

1. **import** RPi.GPIO as GPIO
2. **import** dht11
3. **import** time
4. **import** datetime
6. #GPIO Ayarları
7. GPIO.setwarnings(False)
8. GPIO.setmode(GPIO.BCM)
9. GPIO.cleanup()
11. #Veriyi pin 14' ü kullanarak okuduk
12. instance = dht11.DHT11(pin=14)
14. **while** True:
15. result = instance.read()
17. **if** result.is\_valid():
18. **print**("Son Okuma Zamanı: " + str(datetime.datetime.now()))
19. **print**("Sıcaklık        : %d C" % result.temperature)
20. **print**("Nem             : %d %%" % result.humidity)
22. time.sleep(0.05)
24. GPIO.setup(18,  GPIO.OUT)
26. **if** result.temperature>=26:
27. GPIO.output(18, GPIO.HIGH)
28. **else**:
29. GPIO.output(18, GPIO.LOW)

DHT11\_EXAMPLE.PY KODU

**3.Adım:** Ekran çıktımız şöyle olacak:



DHT11 KODU EKRAN ÇIKTISI

**Gaz Sensörü İçin Adımlar :**

**1.Adım:** Aşağıdaki linkleri kullanarak kodları yükleyin.

**2.Adım:** Arduino’nun ADC pinlerini kullanarak gaz sensörünün verdiği analog çıkışlar dijital değerlere çevrilir.

<https://github.com/tunahan12/OTOMATiK-HAVALANDIRMA-VE-ISITMA-SiSTEMi/tree/master/ARDU%C4%B0NO%20UNO>

1. **char** dataString[50] = {0};
2. **int** a =0;
4. **void** setup() {
5. Serial.begin(9600);              //Seri iletişim 9600 bps ile başlatıldı.
7. **void** loop() {
8. a++;                          // a her döngüde bir değer artar
9. sprintf(dataString,"%02X",a); // Bu degeri hexa'ya dönüştür.
10. Serial.println(dataString);   // Data gönderme
11. delay(1000);                  // Döngüyü 1000 ms beklet
12. }

MQ-6 GAZ SENSÖRÜ SERİ HABERLEŞMESİ 1

**3.Adım:** Arduino tarafından gelen dijital değerler raspberry’e USB’den gönderilir ve burada okunur.

<https://github.com/tunahan12/OTOMATiK-HAVALANDIRMA-VE-ISITMA-SiSTEMi/tree/master/SER%C4%B0%20HABERLE%C5%9EME>

1. **import** serial
3. ser = serial.Serial('/dev/ttyACM0',9600)
4. s = [0,1]
5. **while** True:
6. read\_serial=ser.readline()
7. s[0] = str(int (ser.readline(),16))
8. **print** s[0]
9. **print** read\_serial

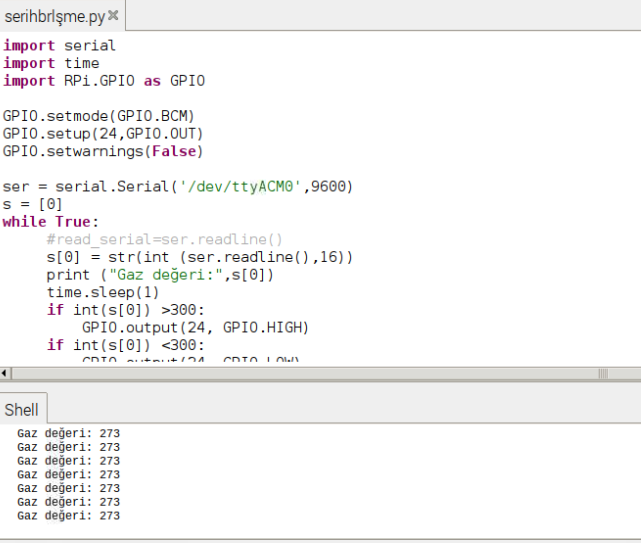
MQ-6 GAZ SENSÖRÜ SERİ HABERLEŞMESİ 2

**4.Adım:** Raspberry ile arduino’yu haberleştirdikten sonra dosya içerisindeki “serihbrlşme.py” dosyasını çalıştırınız.

<https://github.com/tunahan12/OTOMATiK-HAVALANDIRMA-VE-ISITMA-SiSTEMi/tree/master/GAZ%20SENS%C3%96R%C3%9C>

1. **import** serial
2. **import** time
3. **import** RPi.GPIO as GPIO
5. GPIO.setmode(GPIO.BCM)
6. GPIO.setup(24,GPIO.OUT)
7. GPIO.setwarnings(False)
9. ser = serial.Serial('/dev/ttyACM0',9600)
10. s = [0]
11. **while** True:
12. #read\_serial=ser.readline()
13. s[0] = str(int (ser.readline(),16))
14. **print** ("Gaz değeri:",s[0])
15. time.sleep(1)
16. **if** int(s[0]) >300:
17. GPIO.output(24, GPIO.HIGH)
18. **if** int(s[0]) <300:
19. GPIO.output(24, GPIO.LOW)

**5.Adım:** Ekran çıktımız şöyle olacak:



GAZ SENSÖRÜNÜN EKRAN ÇIKTISI

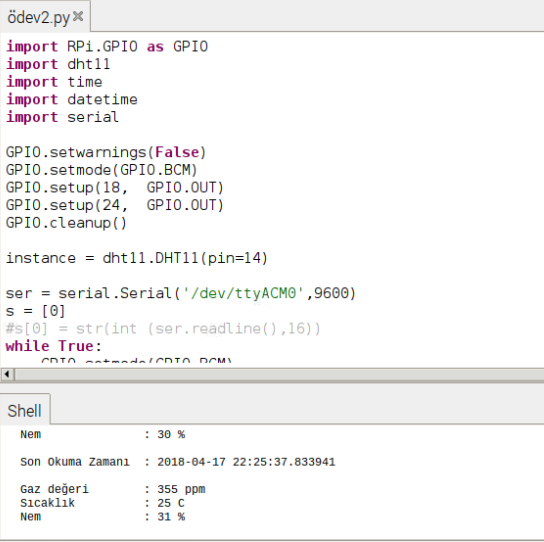
**Sensörlerin Beraberce Okunup İşlendiği Python Kodu:**

<https://github.com/tunahan12/OTOMATiK-HAVALANDIRMA-VE-ISITMA-SiSTEMi/tree/master/ANA%20KOD>

Kodu buradan alabilirsiniz

1. **import** RPi.GPIO as GPIO
2. **import** dht11
3. **import** time
4. **import** datetime
5. **import** serial
7. GPIO.setwarnings(False)
8. GPIO.setmode(GPIO.BCM)
9. GPIO.setup(18,  GPIO.OUT)
10. GPIO.setup(24,  GPIO.OUT)
11. GPIO.cleanup()
13. instance = dht11.DHT11(pin=14)
15. ser = serial.Serial('/dev/ttyACM0',9600)
16. s = [0]
17. #s[0] = str(int (ser.readline(),16))
18. **while** True:
19. GPIO.setmode(GPIO.BCM)
20. GPIO.setup(18,  GPIO.OUT)
21. GPIO.setup(24,  GPIO.OUT)
23. result = instance.read()
24. s[0]=str(int(ser.readline(),16))
25. #time.sleep(1)
27. **if**   int(s[0])>300 **and** result.temperature>26:
29. **if** result.is\_valid():
30. GPIO.output(18, GPIO.HIGH)
31. GPIO.output(24, GPIO.HIGH)
33. #print("\nİlk if")
34. **print**("\nSon Okuma Zamanı  : " + str(datetime.datetime.now()))
35. **print**("\nGaz değeri        :" ,s[0],"ppm")
36. **print**("Sıcaklık          : %d C" % result.temperature)
37. **print**("Nem               : %d %%" % result.humidity)
39. **elif** int(s[0]) >300 **and** result.temperature<26:
41. **if** result.is\_valid():
42. GPIO.output(18, GPIO.LOW)
43. GPIO.output(24, GPIO.HIGH)
45. #print("\nİkinci if")
46. **print**("\nSon Okuma Zamanı  : " + str(datetime.datetime.now()))
47. **print**("\nGaz değeri        :" ,s[0],"ppm")
48. **print**("Sıcaklık          : %d C" % result.temperature)
49. **print**("Nem               : %d %%" % result.humidity)
51. **elif** int(s[0]) <300 **and** result.temperature>26:
53. **if** result.is\_valid():
54. GPIO.output(18, GPIO.HIGH)
55. GPIO.output(24, GPIO.LOW)
57. #print("\nÜçüncü if")
58. **print**("\nSon Okuma Zamanı  : " + str(datetime.datetime.now()))
59. **print**("\nGaz değeri        :" ,s[0],"ppm")
60. **print**("Sıcaklık          : %d C" % result.temperature)
61. **print**("Nem               : %d %%" % result.humidity)
63. **elif** int(s[0]) <300 **and** result.temperature<26:
65. **if** result.is\_valid():
66. GPIO.output(18, GPIO.LOW)
67. GPIO.output(24, GPIO.LOW)
69. #print("\nDördüncü if")
70. **print**("\nSon Okuma Zamanı  : " + str(datetime.datetime.now()))
71. **print**("\nGaz değeri        :" ,s[0],"ppm")
72. **print**("Sıcaklık          : %d C" % result.temperature)
73. **print**("Nem               : %d %%" % result.humidity)
74. **else**:
75. GPIO.cleanup()

Ekran çıktımız şöyle olacak:



PYTHON KODUNUN EKRAN ÇIKTISI

**ÖNERİLER**

Donanımlar uzaktan kontrol edilerek akıllı ev otomasyon sistemlerine dönüştürülebilir.