

Uygulama Adı:	GSM (SIM900) Modülüne Sahip IoT cihazı ile ThingSpeak Kullanımı	No:	
---------------	---	-----	--

### Uygulamanın Tanıtımı:

SIM 900 GSM modülüne sahip GGS01-GSM/GPS Arduino uyumlu GSM-GPS Shield (<http://www.gndteknik.com/gndkits/arduino/gsm-gps-shield>) ile Arduino MEGA IoT cihazı ile DHT11 sensöründen algılanan ortam sıcaklık ve nem bilgisini [www.thingspeak.com](http://www.thingspeak.com) IoT bulut platformuna aktarılması.



Şekil 1. Sistem mimarisi

### Ekipman Listesi ve Kullanılan Teknolojiler:

- GGS01-GSM/GPS Arduino uyumlu GSM-GPS Shield
- Arduino MEGA, DHT22, BreadBoard
- Arduino IDE Program Geliştirme Ortamı
- Thingspeak

### Kullanılan Teknolojilere Yönelik Teknik Bilgiler:

**GGS01-GSM/GPS Arduino uyumlu GSM-GPS Shield** (<http://www.gndteknik.com/gndkits/arduino/gsm-gps-shield>)

GGS01-GSM/GPS Modülü, Arduino UNO ve MEGA ile uyumlu olup SIM900 ya da SIM 800 GSM ve SIM28 GPS entegrelerine sahiptir. Ayrıca, Seri Arayüz Desteği (UART, SPI, I2C) ile Sayısal G/Ç (GPIO) pinlerine sahiptir.

Arama, SMS atma gibi GSM işlemlerini yerine getirebildiği gibi GPRS teknolojisi ile internet bağlantısı sunar. Böylelikle M2M, SCADA, Telemetry, uzaktan cihaz takibi ve kontrolü gibi uygulamalarda kullanılmaktadır.

SIM900 GSM Modülün sunduğu özellikler ise, dört farklı band desteği ( 850 / 900 / 1800 / 1900 MHz), GPRS çoklu-slot sınıfı 10/8, GPRS mobil istasyon sınıf B, AT komutları ile kontrol, kısa mesaj servisi (SMS), gömülü TCP/UDP yığını (web sunucuya veri yükleme), gerçek zamanlı saat (RTC), seçilebilir seri port, hoparlör ve kulaklık çıkışıdır. Ayrıca düşük güç tüketimine (uyku 1.5 mA) sahiptir. Çalışma sıcaklığı ise - 40 C ile + 85 C'dir.

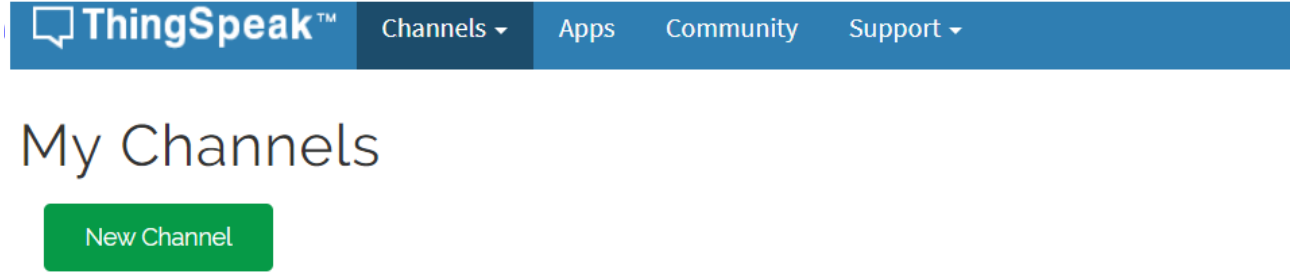
ESP8266 WiFi modülüne sahip Wemos D1, NodeMCU gibi kartları Arduino IDE'nizde tanımlı kartlar arasına ekleyebilmek için Şekil 2'de görüldüğü gibi **Dosya > Tercihler** sekmesindeki ekranda **"Ek Devre Kartları**

## ThingSpeak

Mathworks altyapısına sahip ThingSpeak IoT bulut platformları içerisinde grafiksel sunum özellikleri ile öne çıkmaktadır.

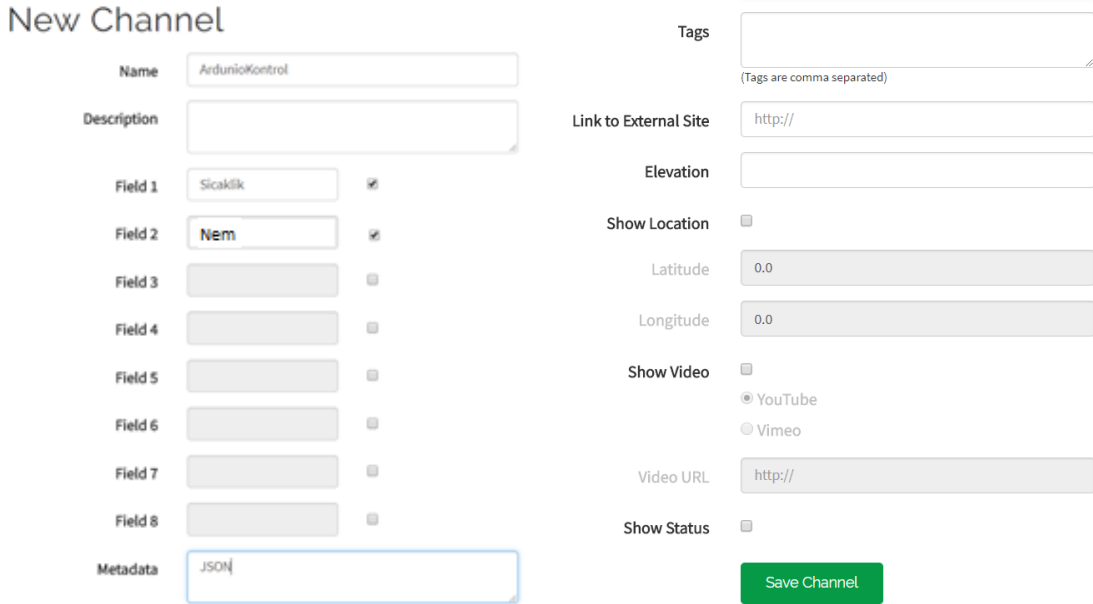
ThingSpeak'te veri iletişimi (gönderme ve alma işlemleri) kanallar aracılığıyla gerçekleştirilir. Public ve Private seçenekleri ile kanallar üzerinden iletişim gizli/güvenli ya da herkese açık yapılabilir.

ThingSpeak'e giriş yapıldığında Şekil 2'de görülen arayüzden New Channel seçeneği ile IoT cihazınız ile haberleşilecek kanal işlemleri gerçekleştirilir.

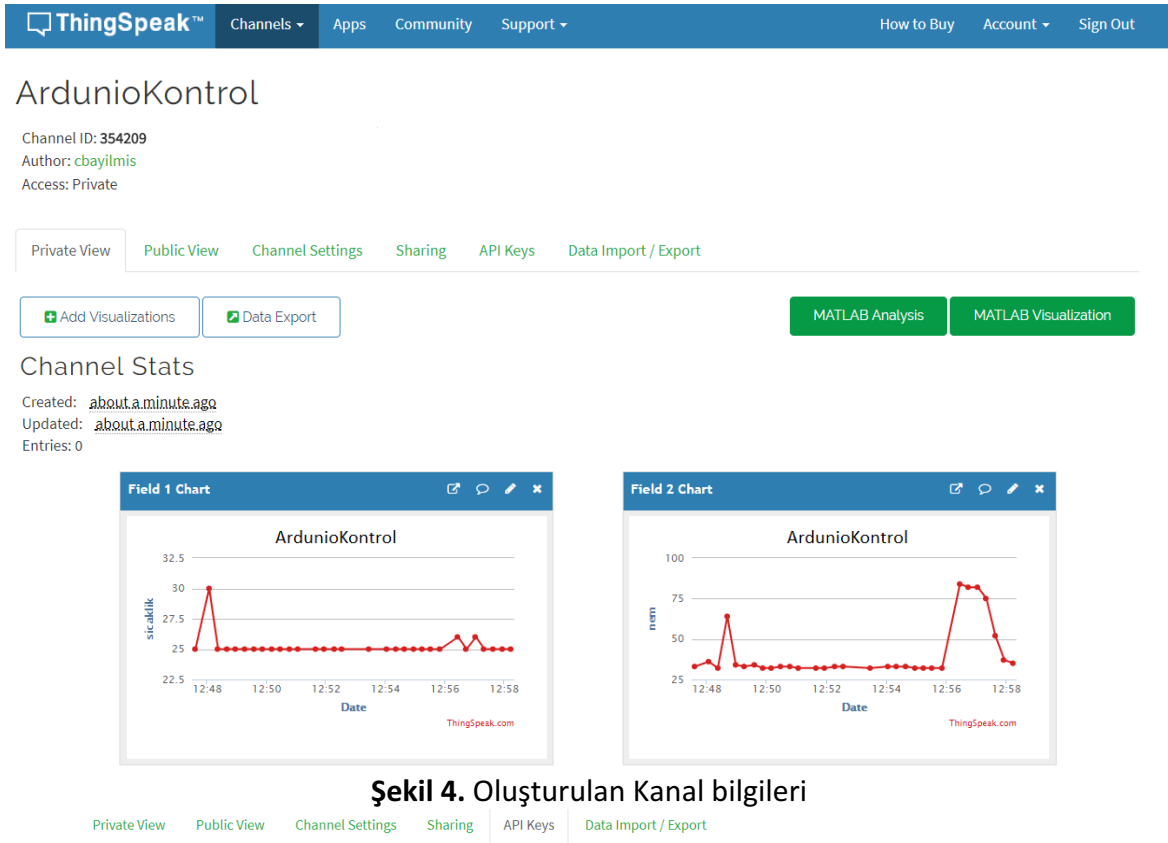


Şekil 2. ThingSpeak yeni kanal oluşturma

Şekil 3'den görüldüğü üzere 1 kanal içerisinde 8 alan (Field) yani 8 farklı veri bağlantısı tanımlanabilir. Kanalımızın Public veya Private olmasının da bu sayfa üzerinden belirliyoruz. İlgili alanlar doldurulduktan sonra ekranın altındaki "Save Channel" butonuna tıklanır. Kanal ile ilgili alanlar üzerinde istenirse sonradan da düzenleme yapılabilir.

The image shows the 'New Channel' form in ThingSpeak. It includes fields for Name (set to 'ArduinoKontrol'), Description, and Metadata (set to 'JSON'). There are eight 'Field' input boxes, with 'Field 2' set to 'Nem'. On the right, there are options for Tags, Link to External Site, Elevation, Show Location (with Latitude and Longitude set to 0.0), Show Video (with YouTube selected), Video URL, and Show Status. A green 'Save Channel' button is at the bottom right.

Şekil 3. ThingSpeak kanal bilgileri girme



#### Write API Key

Key OJ6LQV8H1UDXKW5N

Generate New Write API Key

#### Read API Keys

Key KWLNR7RC37LM5GXV

Note

Save Note

Delete API Key

Generate New Read API Key

#### Help

API keys enable you to write data to a channel or read data from a private channel. API keys are auto-generated when you create a new channel.

#### API Keys Settings

- Write API Key:** Use this key to write data to a channel. If you feel your key has been compromised, click **Generate New Write API Key**.
- Read API Keys:** Use this key to allow other people to view your private channel feeds and charts. Click **Generate New Read API Key** to generate an additional read key for the channel.
- Note:** Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.

#### API Requests

##### Update a Channel Feed

GET https://api.thingspeak.com/update?api\_key=OJ6LQV8H1UDXKW5N&field=

##### Get a Channel Feed

GET https://api.thingspeak.com/channels/354209/feeds.json?api\_key=KWLNR7RC37LM5GXV

##### Get a Channel Field

GET https://api.thingspeak.com/channels/354209/fields/1.json?api\_key=KWLNR7RC37LM5GXV

##### Get Channel Status Updates

GET https://api.thingspeak.com/channels/354209/status.json?api\_key=KWLNR7RC37LM5GXV

Şekil 5. Kanal yazma ve okuma api anahtarları

Kanal oluşturma işlemi tamamlandıktan sonra artık veriyi gönderip, grafiksel olarak izleme aşamasına geldik. Veriyi gönderirken get metodunu kullanacağız. “GET /update?key=Write Api Key&Field Name=” buradaki Write Api Key her kanal için özgün bir şekilde üretilir. Veriyi hangi alanda göstereceğimizi de metodun sonuna ekliyoruz. (Kanalı oluştururken verdiğimiz field adı). Write Api Key Şekil 5’de görülmektedir.

Örnek Kullanım: “GET /update?key= OJ6LQV8H1UDXKW5N&Sicaklik=”

Get metodunu kodumuza ekledikten sonra artık verimiz internet ortamına aktarılmaktadır.

## SIM900 AT Komutları ile ThingSpeak Erişimi

```

/*****
  BSM 451 Nesnelerin İnterneti ve Uygulaması Dersi
  Doç. Dr. Cüneyt BAYILMIŞ
  GSM Modülü ile ThingSpeak Uygulaması
  *****/

#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>

#define DHTPIN          22          // DHT Sensörün bağlı olduğu pin

#define DHTTYPE          DHT22      // DHT 22 (AM2302)
|
// https://learn.adafruit.com/dht/overview

DHT_Unified dht(DHTPIN, DHTTYPE);

uint32_t delayMS;
int sıcaklik;
int nem;

void setup() {
  Serial.begin(9600);
  Serial1.begin(19200);///GSM
  // Initialize device.
  dht.begin();
  Serial.println("DHT22 Unified Sensor Example");
  // Print temperature sensor details.
  sensor_t sensor;
  dht.temperature().getSensor(&sensor);
  Serial.println("-----");
  Serial.println("Temperature");
  Serial.print  ("Sensor:      "); Serial.println(sensor.name);
  Serial.print  ("Driver Ver:   "); Serial.println(sensor.version);
  Serial.print  ("Unique ID:    "); Serial.println(sensor.sensor_id);
  Serial.print  ("Max Value:    "); Serial.print(sensor.max_value); Serial.println(" *C");
  Serial.print  ("Min Value:    "); Serial.print(sensor.min_value); Serial.println(" *C");
  Serial.print  ("Resolution:   "); Serial.print(sensor.resolution); Serial.println(" *C");
  Serial.println("-----");
  // Print humidity sensor details.
  dht.humidity().getSensor(&sensor);
  Serial.println("-----");
  Serial.println("Humidity");
  Serial.print  ("Sensor:      "); Serial.println(sensor.name);
  Serial.print  ("Driver Ver:   "); Serial.println(sensor.version);
  Serial.print  ("Unique ID:    "); Serial.println(sensor.sensor_id);
  Serial.print  ("Max Value:    "); Serial.print(sensor.max_value); Serial.println("%");
  Serial.print  ("Min Value:    "); Serial.print(sensor.min_value); Serial.println("%");
  Serial.print  ("Resolution:   "); Serial.print(sensor.resolution); Serial.println("%");
  Serial.println("-----");

  delayMS = sensor.min_delay / 1000;
}

```

```
void loop() {

    delay(delayMS); // Ölçümler arası gecikme

    sensors_event_t event;

    // SICAKLIK
    dht.temperature().getEvent(&event);
    if (isnan(event.temperature)) {
        Serial.println("Error reading temperature!");
    }
    else {
        Serial.print("Temperature: ");
        Serial.print(event.temperature);
        Serial.println(" *C");
        sicaklik = event.temperature;
    }
    //NEM.
    dht.humidity().getEvent(&event);
    if (isnan(event.relative_humidity)) {
        Serial.println("Error reading humidity!");
    }
    else {
        Serial.print("Humidity: ");
        Serial.print(event.relative_humidity);
        Serial.println("%");
        nem = event.relative_humidity;
    }
    SendSQL();
    delay(500);
}

void SendSQL()
{
    Serial1.println("AT+SAPBR=3,1,\"CONTTYPE\",\"GPRS\"");
    delay(500);
    Serial1.println("AT+SAPBR=3,1,\"APN\",\"CMNET\""); // Turkcell için CMNET yerine internet yazın
    delay(500);
    Serial1.println("AT+SAPBR=1,1");
    delay(500);
    Serial1.println("AT+SAPBR=2,1");
    delay(500);
    Serial1.println("AT+HTTINIT");
    delay(500);
    Serial1.println("AT+HTTTPARA=\"CID\",1");
    delay(500);
    Serial1.print("AT+HTTTPARA=\"URL\", \"http://api.thingspeak.com/update?api_key=OJ6LQV8H1UDXKW5N&\"");
    Serial1.print("field1=");
    Serial1.print(sicaklik);
    Serial1.print("&field2=");
    Serial1.print(nem);
    Serial1.println("");
    delay(3000);
    Serial1.println("AT+HTTTPACTION=0");
    delay(8000);
    Serial1.println("AT+HTTPTERM");
    delay(500);
    Serial1.println("AT+SAPBR=0,1");
    delay(500);
    Serial.println("Veri Gonderildi");
}
```

## KAYNAK

Doç. Dr. Cüneyt BAYILMIŞ ve Doç. Dr. Kerem KÜÇÜK, “Nesnelerin İnternet’i: Teori ve Uygulamaları”, Papatya Yayınevi, 2019.