

**LAPORAN SIMULASI
MICROCONTROLLER DAN MICROPROCESSOR
PADA SISTEM ALARM DETEKSI KEBAKARAN BERBASIS IOT DAN ARDUINO
DENGAN APLIKASI BLYNK**



**Dosen Pengampu :
Ir. Husnibes Muchtar, MT**

**Disusun oleh :
Barokah Asmarahman Takarob
2018420035**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
UNIVERSITAS MUHAMMADIYAH JAKARTA
2021**

1. PENDAHULUAN

1.1 Latar Belakang

Kebakaran menjadi salah satu musibah yang banyak ditakuti dan menimbulkan trauma bagi penduduk dunia khususnya di Indonesia. Beberapa pemicu timbulnya kebakaran bisa disebabkan oleh karena faktor eksternal maupun kelalaian manusia, diantara yang sering kita dengar adalah kebakaran karena arus pendek listrik, ledakan tabung gas, membuang puntung rokok sembarangan, suhu yang ekstrim dan lain-lain. Kebakaran menimbulkan dampak kerugian yang sangat besar seperti hilangnya tempat tinggal, harta benda, dan aset-aset berharga, serta bahkan korban jiwa. Musibah dalam kasus kebakaran tersebut sebenarnya dapat dicegah, banyak cara yang dapat digunakan untuk mencegahnya. Salah satu cara untuk mencegahnya adalah dengan menggunakan suatu alat deteksi kebakaran.

Berdasarkan latar belakang kasus diatas, maka Saya membuat simulasi pada tugas Mata Kuliah Mikroprosesor dan Mikrokontroler dengan tema "Sistem Alarm Deteksi Kebakaran Berbasis IoT dan Arduino Dengan Aplikasi Blynk". Perancangan sistem alarm kebakaran ini merupakan suatu rancangan sistem yang terhubung melalui jaringan internet dan dapat memberikan informasi berupa notifikasi keadaan atau situasi sebuah ruangan kepada pihak terkait melalui aplikasi Blynk yang diharapkan dapat digunakan sebagai langkah awal untuk mencegah terjadinya musibah kebakaran dalam skala besar.

1.2 Rumusan Masalah

1. Desain, komponen, dan program, serta aplikasi yang digunakan.
2. Uji coba simulasi sistem alarm kebakaran.

1.3 Batasan Masalah

Simulasi ini hanya bisa mendeteksi api dan asap kemudian memberikan notifikasi pada smartphone namun tidak bisa menentukan sebuah titik api serta memadamkan api apabila terjadi kebakaran.

1.4 Tujuan

Simulasi ini bertujuan untuk memberikan edukasi sekaligus inspirasi guna membangun sebuah perangkat berbasis IoT (Internet Of Things) yang diaplikasikan untuk penanggulangan bencana kebakaran.

2. METODOLOGI SIMULASI

Metodologi pada simulasi ini dilaksanakan dengan menerapkan :

1. Pendalaman materi
2. Perancangan sistem
3. Uji coba rangkaian

3. TINJAUAN PUSTAKA

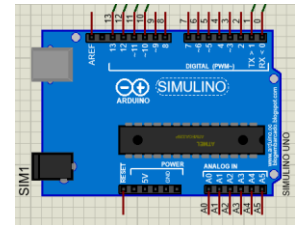
3.1 IoT (*Internet Of Things*)

IoT merupakan segala sesuatu tentang objek yang terkoneksi dan terintegrasi dengan jaringan internet dan menjadi infrastruktur public yang digunakan untuk menyediakan interkoneksi baik fisik maupun virtual. Singkatnya, IoT terdiri dari beberapa bagian diantaranya adalah sensor, jaringan, koneksi internet, dan aplikasi. Beberapa teknologi komunikasi dapat digunakan untuk koneksi perangkat ke IoT.



3.2 Arduino Uno

Arduino uno merupakan mikrokontroler yang dikontrol secara penuh oleh mikroprosesor ATmega328. Mikroprosesor yang digunakan ini sudah dilengkapi dengan converter sinyal analog ke digital (ADC) sehingga tidak diperlukan penambahan ADC eksternal.



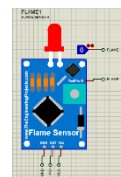
3.3 Blynk

Blynk merupakan platform aplikasi pada smartphone yang menyediakan layanan untuk pengendalian modul mikrokontroler/mikroprosesor melalui koneksi internet.



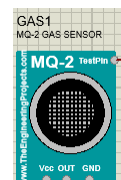
3.4 Flame Sensor

Flame sensor merupakan salah satu alat pendeteksi kebakaran melalui adanya nyala api yang bekerja dengan cara mendeteksi panjang gelombang kurang lebih antara 760 nm sampai dengan 1.100 nm, dan transducer yang digunakan dalam mendeteksi nyala api tersebut adalah infrared.



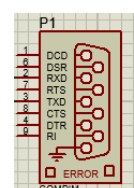
3.5 Gas Sensor MQ-2

Gas sensor MQ-2 adalah sensor yang digunakan untuk mendeteksi konsentrasi gas yang mudah terbakar di udara serta asap, kemudian menghasilkan output berupa tegangan analog.



3.6 Compim

Compim disini berfungsi untuk menghubungkan perangkat simulasi yaitu Proteus dan Arduino dengan platform aplikasi Blynk secara serial.

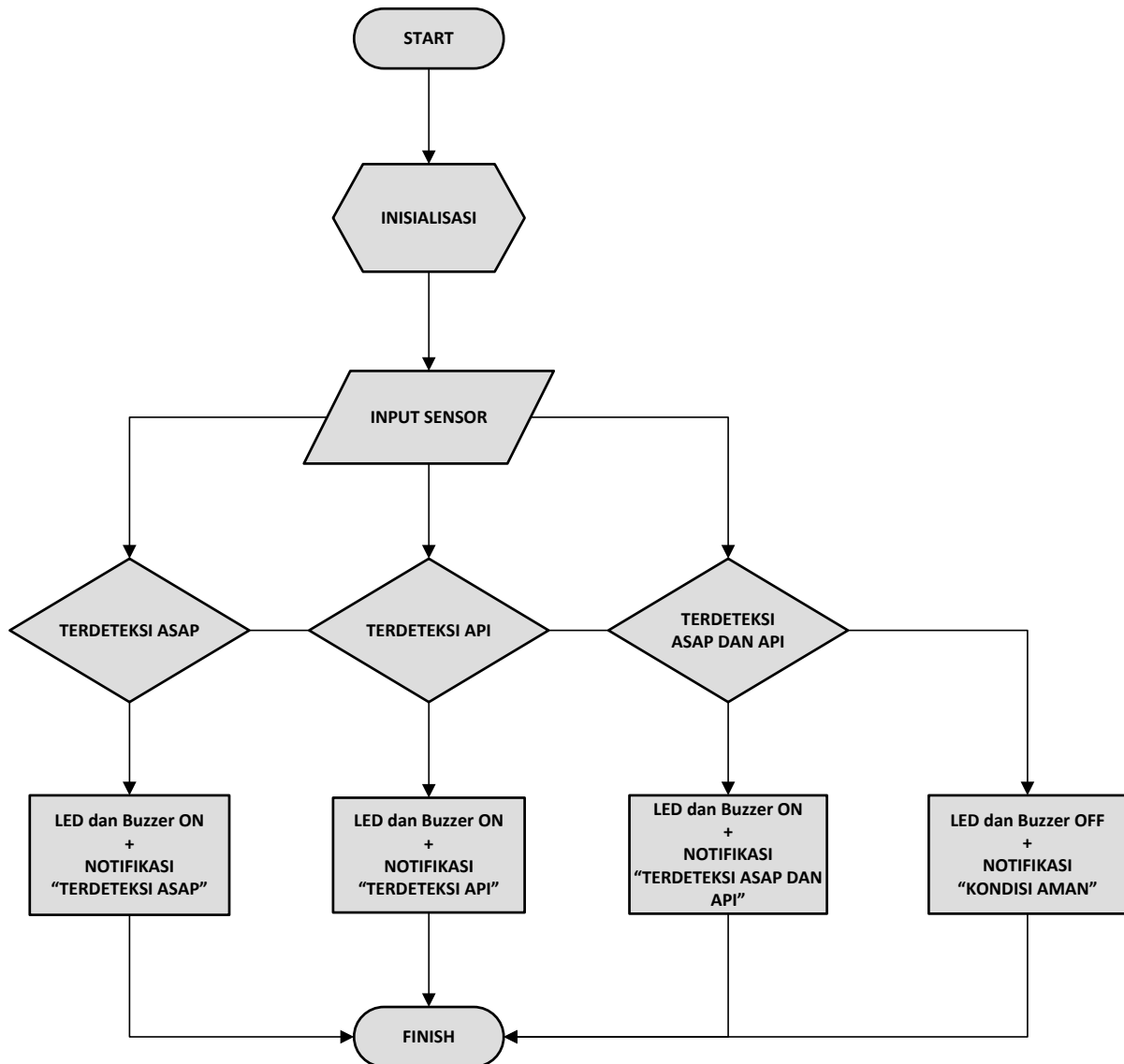


4. METODE SIMULASI

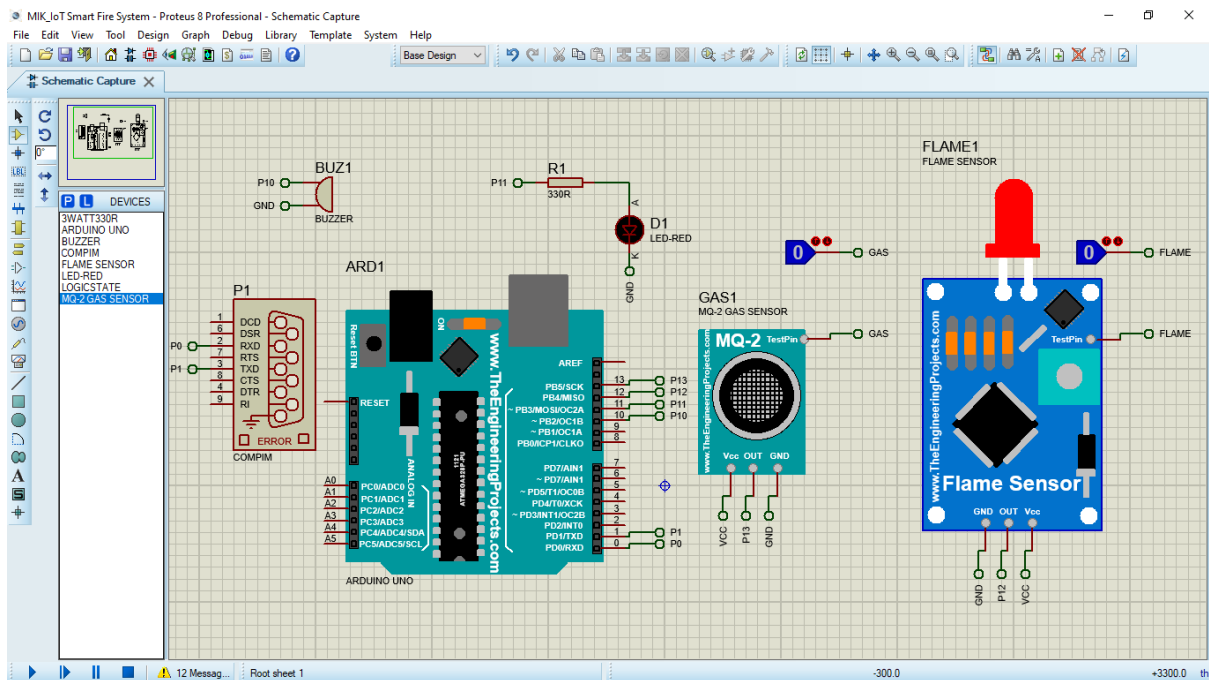
4.1 Blok Diagram



4.2 Flowchart



4.3 Rangkaian Simulasi



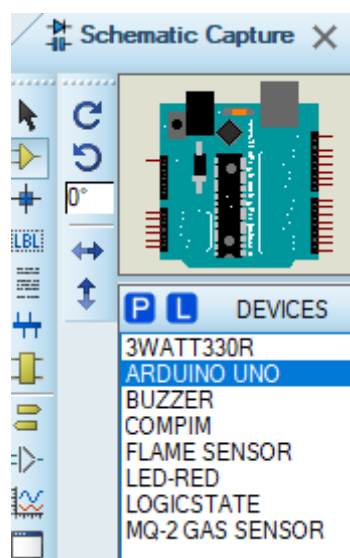
4.3.1 Perangkat dan komponen yang dibutuhkan

Laptop :

- Software Proteus, Arduino IDE, VSPE (*Virtual Serial Port Emulator*)
- Aplikasi Blynk pada Smartphone

Proteus :

- Arduino Uno
- Flame Sensor
- Gas Sensor MQ-2
- Compim
- LED
- Buzzer



4.3.2 Coding pada Arduino IDE

```
//Tugas MIK_IoT Smart Fire System_Barokah Asmarahman Takarob_2018420035

#include <BlynkSimpleStream.h> 1

int asap = 13, api = 12, LED = 11, buzzer = 10; 2

char auth[] = "HdnOmrK9skIiJox7slttS8XjgTnzfbNk"; //AUTH TOKEN PADA APLIKASI BLYNK 3

WidgetLCD lcd (V0); //UNTUK DISPLAY PADA BLYNK 4

void setup()
{
  pinMode (api, INPUT);
  pinMode (asap, INPUT); 5

  pinMode (LED, OUTPUT);
  pinMode (buzzer, OUTPUT);

  Serial.begin(9600); //KOMUNIKASI BLYNK SECARA SERIAL
  Blynk.begin(auth, Serial); 6
}

void loop()
{
  lcd.clear();
  int flame = digitalRead(api);
  int smoke = digitalRead(asap); 7

  if(flame == HIGH && smoke == HIGH) //Flame Sensor LOGIC 1 dan Gas Sensor LOGIC 1 8
  {
    lcd.clear();
    lcd.print(0, 0, "TERDETEKSI API & ASAP !!!");
    digitalWrite(LED, HIGH);
    digitalWrite(buzzer, HIGH);
    Blynk.notify("BAHAYA, ADA API & ASAP DI RUMAH !!!");
    delay(100);
  }
  else if(flame == LOW && smoke == HIGH) //Flame Sensor LOGIC 0 dan Gas Sensor LOGIC 1 10
  {
    lcd.clear();
    lcd.print(0, 0, "TERDETEKSI ASAP !!!");
    digitalWrite(LED, HIGH);
    digitalWrite(buzzer, HIGH);
    Blynk.notify("BAHAYA, ADA ASAP DI RUMAH !!!");
    delay(100);
  }
  else if(flame == HIGH && smoke == LOW) //Flame Sensor LOGIC 1 dan Gas Sensor LOGIC 0 12
  {
    lcd.clear();
    lcd.print(0, 0, "TERDETEKSI API !!!");
    digitalWrite(LED, HIGH);
    digitalWrite(buzzer, HIGH);
    Blynk.notify("BAHAYA, ADA API DI RUMAH !!!");
    delay(100);
  }
}
```

```

else //Flame Sensor dan Gas Sensor LOGIC 0
{
  lcd.clear();
  lcd.print(0, 0, "KONDISI AMAN");
  digitalWrite(LED, LOW);
  digitalWrite(buzzer, LOW);
  delay(200);
}
Blynk.run(); //MENGAKTIFKAN FUNGSI BLYNK
}

```

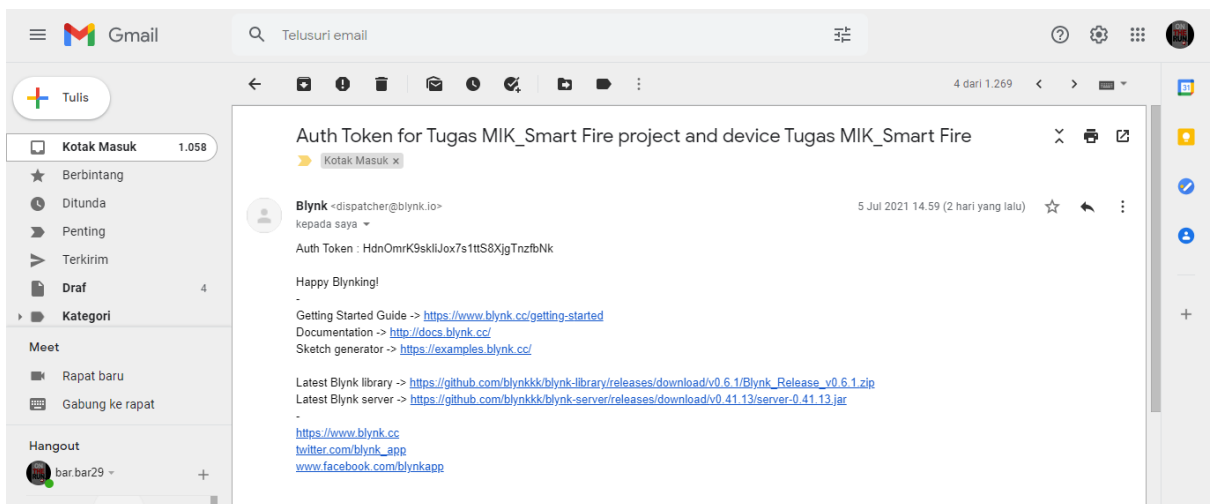
14

15

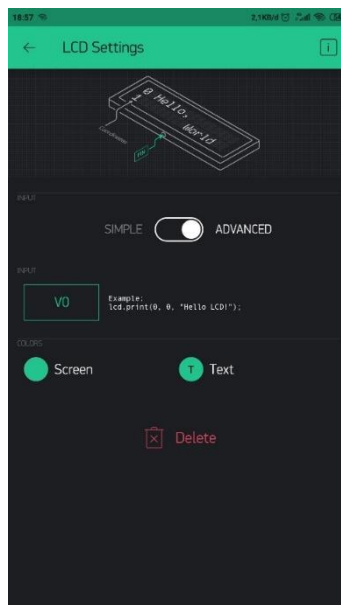
16

Penjelasan Coding Pada Arduino IDE :

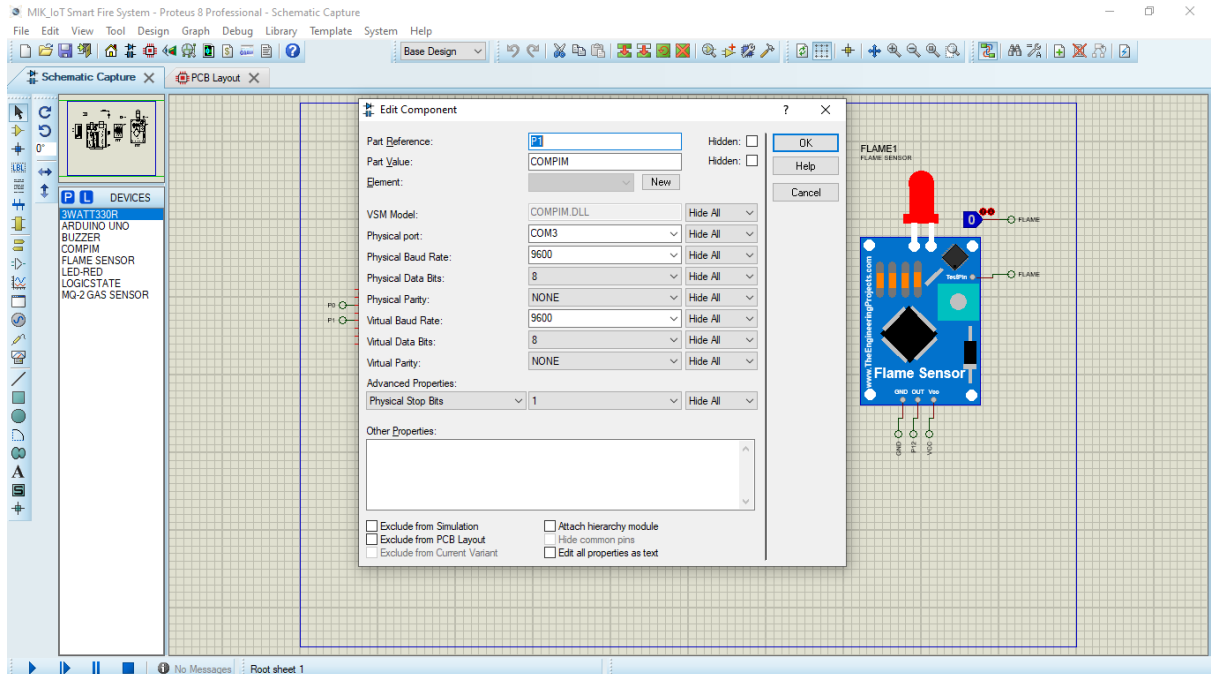
1. *BlynkSimpleStream* merupakan library yang digunakan.
2. Variabel dan koneksi pada digital pin Arduino Uno, yakni (asap) pada pin 13, (api) pada pin 12, (LED) pada pin 11, dan (buzzer) pada pin 10.
3. *Auth Token* tersebut diperoleh dari Blynk yang dikirim via *e-Mail* sebagai identitas.



4. *Virtual Port* yang ditentukan sesuai dengan menu *LCD Setting* pada aplikasi Blynk di Smartphone yakni **V0**.



- Flame Sensor (api) dan Gas Sensor MQ-2 (asap) sebagai INPUT, sedangkan LED dan Buzzer sebagai OUTPUT.
- Blynk dan Proteus terhubung secara serial melalui *Auth Token* dan *Compim* dengan nilai *Virtual Baud Rate* sebesar 9600.



- Pembacaan secara digital (*dengan Logicstate*) melalui Flame Sensor (pin 12) untuk mengidentifikasi adanya api, dan Gas Sensor (pin 13) untuk mengidentifikasi adanya asap.
- Kondisi dimana disimulasikan Flame Sensor **HIGH** (Logicstate 1) dan Gas Sensor **HIGH** (Logicstate 1), artinya kedua sensor tersebut mendeteksi adanya api serta asap.
- Pada Smartphone, LCD Blynk akan menampilkan **"TERDETEKSI API & ASAP !!!"** kemudian LED dan Buzzer akan ON, dilanjutkan dengan notifikasi pop-up **"BAHAYA, ADA API & ASAP DI RUMAH !!!"**.
- Kondisi dimana disimulasikan Flame Sensor **LOW** (Logicstate 0) dan Gas Sensor **HIGH** (Logicstate 1), artinya Flame Sensor tidak mendeteksi adanya api namun Gas Sensor mendeteksi adanya asap.
- Pada Smartphone, LCD Blynk akan menampilkan **"TERDETEKSI ASAP !!!"** kemudian LED dan Buzzer akan ON, dilanjutkan dengan notifikasi pop-up **"BAHAYA, ADA ASAP DI RUMAH !!!"**.
- Kondisi dimana disimulasikan Flame Sensor **HIGH** (Logicstate 1) dan Gas Sensor **LOW** (Logicstate 0), artinya Flame Sensor mendeteksi adanya api namun Gas Sensor tidak mendeteksi adanya asap.

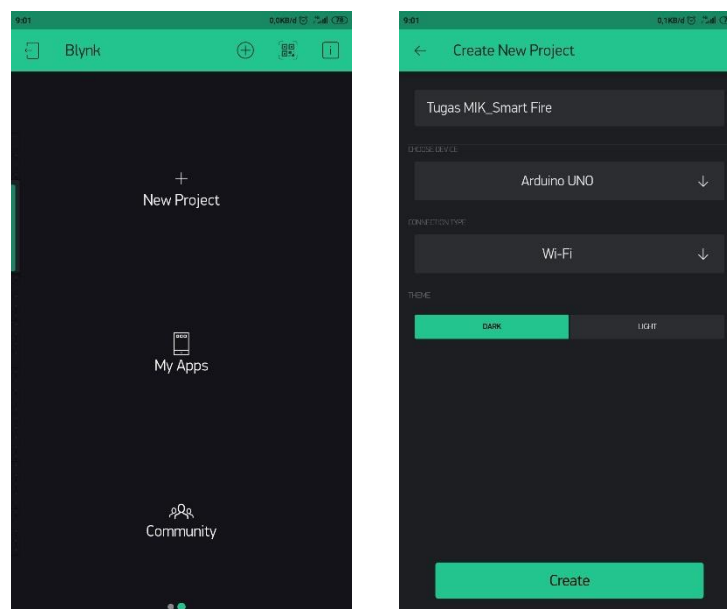
13. Pada Smartphone, LCD Blynk akan menampilkan "**TERDETEKSI API !!!**" kemudian LED dan Buzzer akan ON, dilanjutkan dengan notifikasi pop-up "**BAHAYA, ADA API DI RUMAH !!!**".
14. Merupakan kondisi normal, yaitu Flame Sensor maupun Gas Sensor dalam keadaan Logicstate 0 artinya kedua sensor tersebut tidak mendeteksi adanya api serta asap.
15. Pada Smartphone, LCD Blynk akan menampilkan "**KONDISI AMAN**" kemudian LED dan Buzzer akan OFF.
16. Perintah untuk mengaktifkan fungsi Blynk.

5. PENGUJIAN DAN PROSES SIMULASI

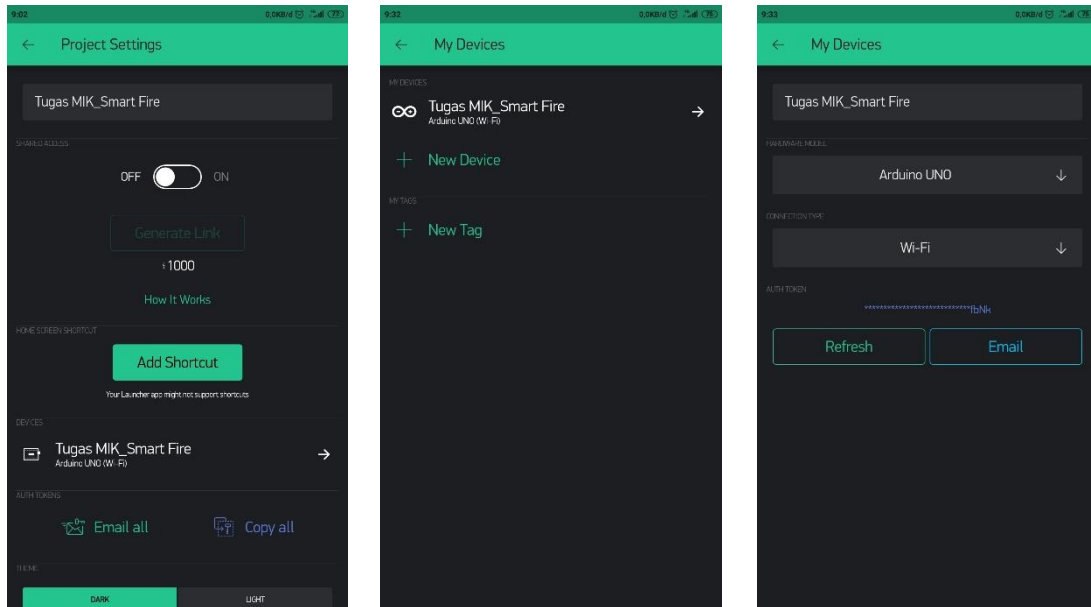
5.1 LANGKAH PENGUJIAN SIMULASI

5.1.1 BLYNK

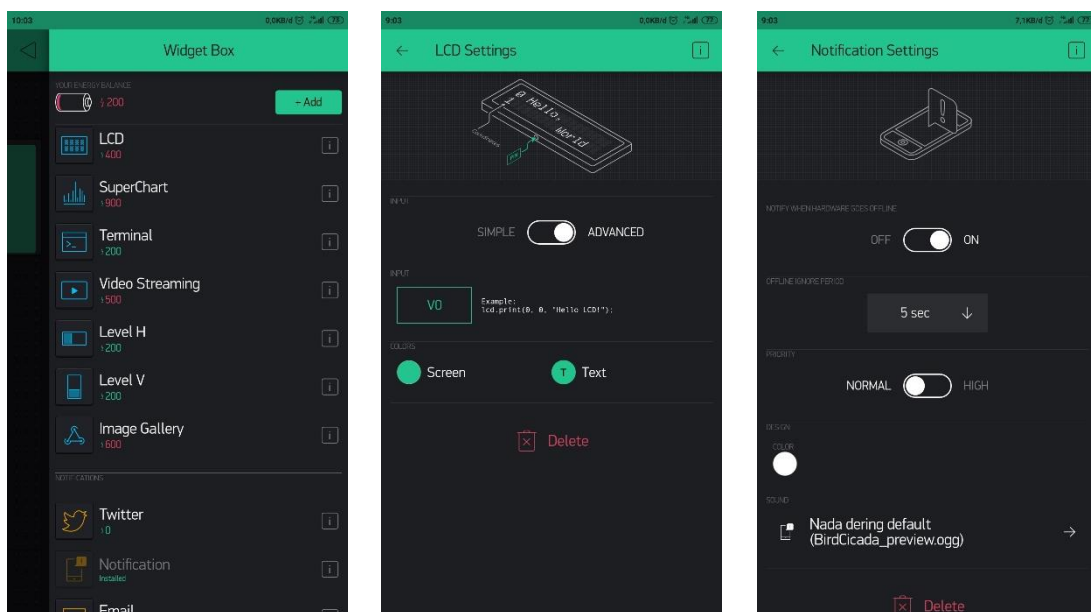
1. Buka aplikasi BLYNK pada Smartphone, lalu buat project baru dengan klik **New Project**. Kemudian ketik nama judul project dan pilih perangkat yang digunakan yaitu **Arduino UNO** beserta koneksinya yaitu **Wi-Fi**, kemudian klik **Create**.



2. Buka pengaturan pada Project Setting (kiri), dan klik pada sub menu **Devices** yang tertera nama judul project. Kemudian pada menu My Device (tengah), klik sub menu **MY DEVICES**. Selanjutnya, pada My Devices (kanan), klik **Email** untuk mengirimkan *Auth Token* yang akan digunakan pada coding program Arduino IDE.

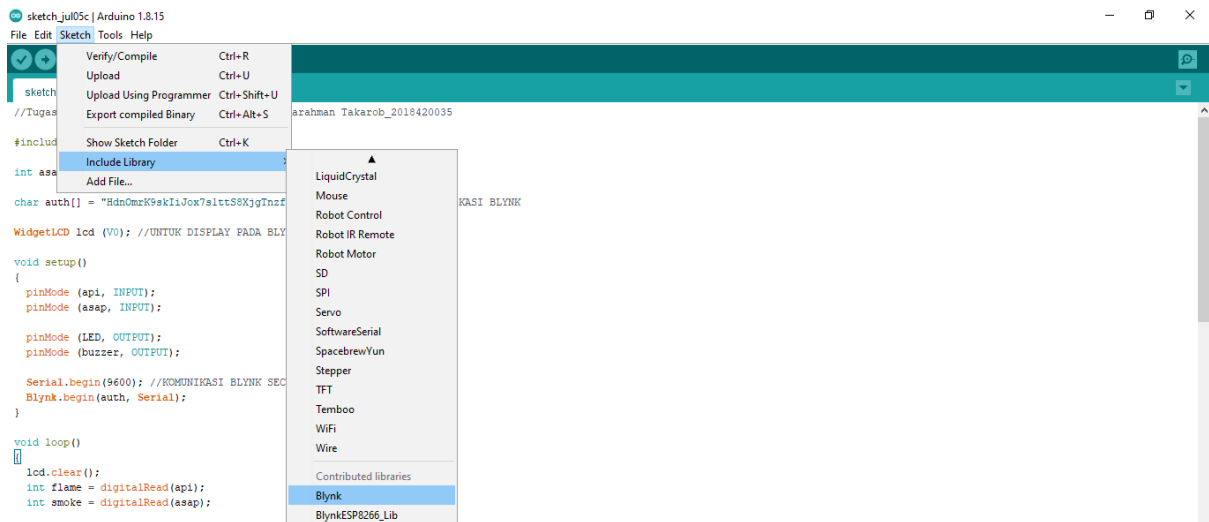


3. Pada menu Widget Box (kiri), tambahkan **LCD** dan **Notification**. Pada LCD Settings (tengah), tentukan INPUT secara **ADVANCED** dan **V0** sebagai virtual port yang akan digunakan dalam coding program Arduino IDE. Pada Notification Settings (kanan), tentukan pengaturan sesuai dengan kebutuhan.

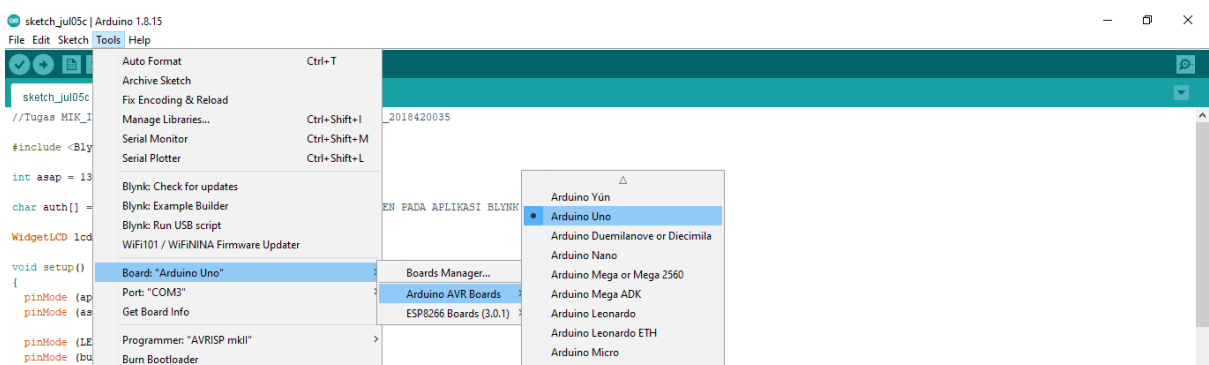


5.1.2 Arduino IDE

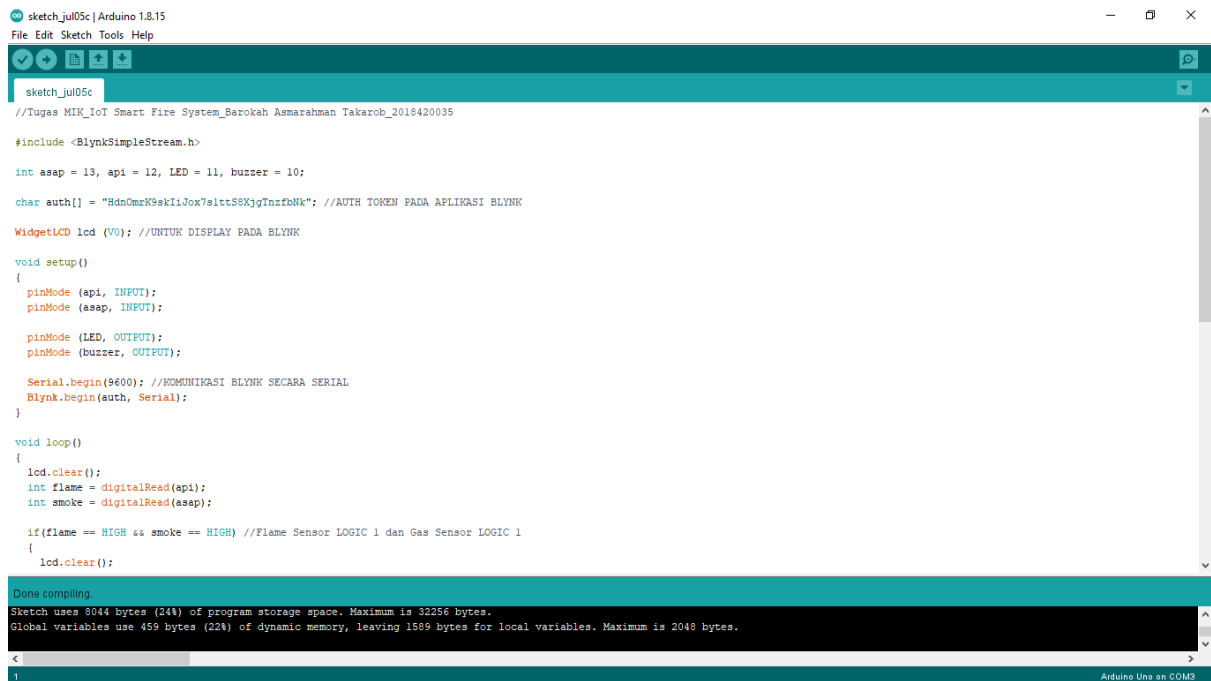
1. Pastikan pada Arduino IDE sudah terdapat **library** dan **tools** Blynk.



2. Pastikan Board Arduino pada program adalah **Arduino Uno**.



3. Kemudian setelah coding program pada Arduino IDE selesai disusun, maka selanjutnya adalah proses **Verify** untuk mendeteksi apabila ada variable atau parameter yang belum sesuai dengan program/library/tools yang tersedia.



```
sketch_jul05c | Arduino 1.8.15
File Edit Sketch Tools Help

sketch_jul05c
//Tugas MIK IoT Smart Fire System_Barakah Asmarahman Takarob_2018420035

#include <BlynkSimpleStream.h>

int asap = 13, api = 12, LED = 11, buzzer = 10;

char auth[] = "HdnOmK9skIiJox7sitt58XjgTnsfbNk"; //AUTH TOKEN PADA APLIKASI BLYNK

WidgetLCD lcd (V0); //UNTUK DISPLAY PADA BLYNK

void setup()
{
  pinMode (api, INPUT);
  pinMode (asap, INPUT);

  pinMode (LED, OUTPUT);
  pinMode (buzzer, OUTPUT);

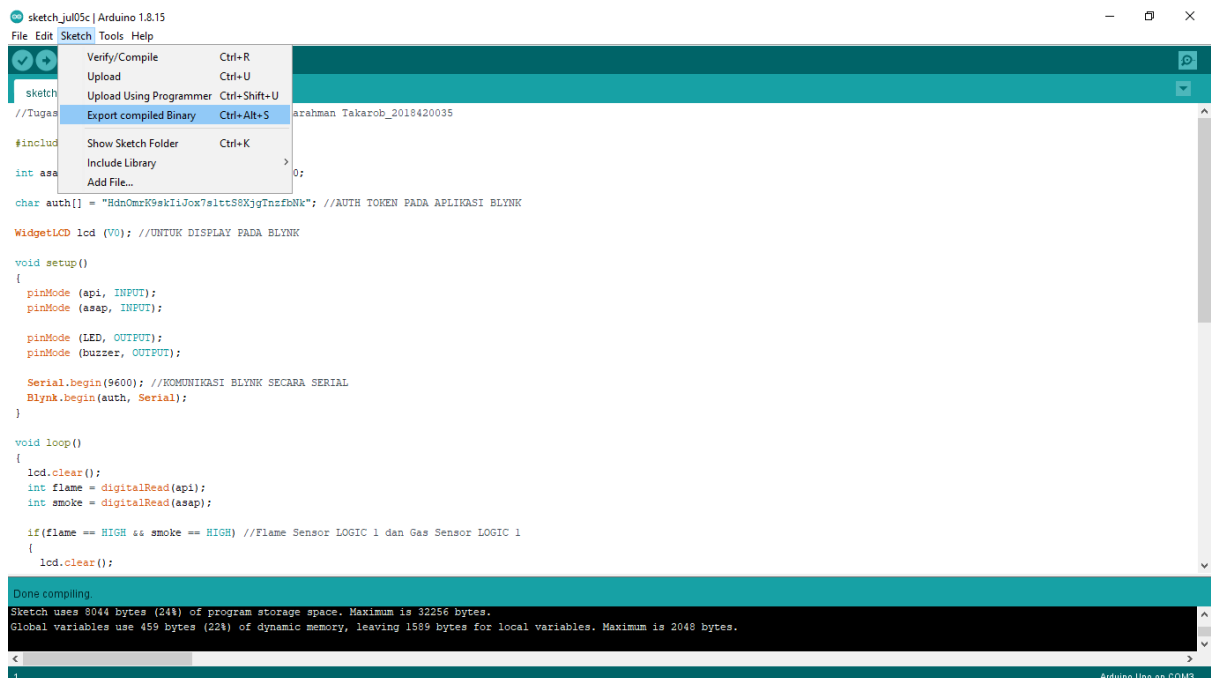
  Serial.begin(9600); //KOMUNIKASI BLYNK SECARA SERIAL
  Blynk.begin(auth, Serial);
}

void loop()
{
  lcd.clear();
  int flame = digitalRead(api);
  int smoke = digitalRead(asap);

  if(flame == HIGH && smoke == HIGH) //Flame Sensor LOGIC 1 dan Gas Sensor LOGIC 1
  {
    lcd.clear();
  }
}

Done compiling.
Sketch uses 8044 bytes (24%) of program storage space. Maximum is 32256 bytes.
Global variables use 459 bytes (22%) of dynamic memory, leaving 1589 bytes for local variables. Maximum is 2048 bytes.
```

4. Setelah proses verifikasi selesai, berikutnya adalah memilih sub menu **Export Compiled Binary** supaya program tersebut bisa di inject pada modul Arduino Uno di Proteus.



```
sketch_jul05c | Arduino 1.8.15
File Edit Sketch Tools Help

sketch
  Verify/Compile      Ctrl+R
  Upload             Ctrl+U
  Upload Using Programmer Ctrl+Shift+U
  Export compiled Binary Ctrl+Alt+S
  Show Sketch Folder Ctrl+K
  Include Library
  Add File...

sketch_jul05c
//Tugas MIK IoT Smart Fire System_Barakah Asmarahman Takarob_2018420035

#include <BlynkSimpleStream.h>

int asap = 13, api = 12, LED = 11, buzzer = 10;

char auth[] = "HdnOmK9skIiJox7sitt58XjgTnsfbNk"; //AUTH TOKEN PADA APLIKASI BLYNK

WidgetLCD lcd (V0); //UNTUK DISPLAY PADA BLYNK

void setup()
{
  pinMode (api, INPUT);
  pinMode (asap, INPUT);

  pinMode (LED, OUTPUT);
  pinMode (buzzer, OUTPUT);

  Serial.begin(9600); //KOMUNIKASI BLYNK SECARA SERIAL
  Blynk.begin(auth, Serial);
}

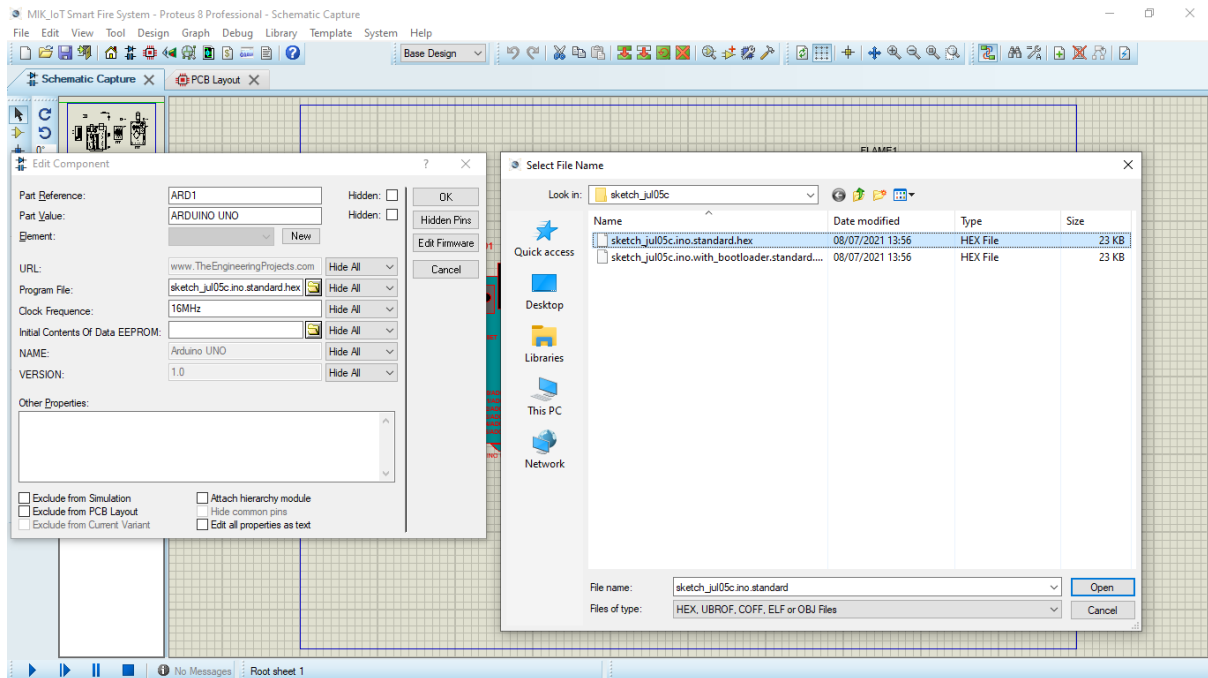
void loop()
{
  lcd.clear();
  int flame = digitalRead(api);
  int smoke = digitalRead(asap);

  if(flame == HIGH && smoke == HIGH) //Flame Sensor LOGIC 1 dan Gas Sensor LOGIC 1
  {
    lcd.clear();
  }
}

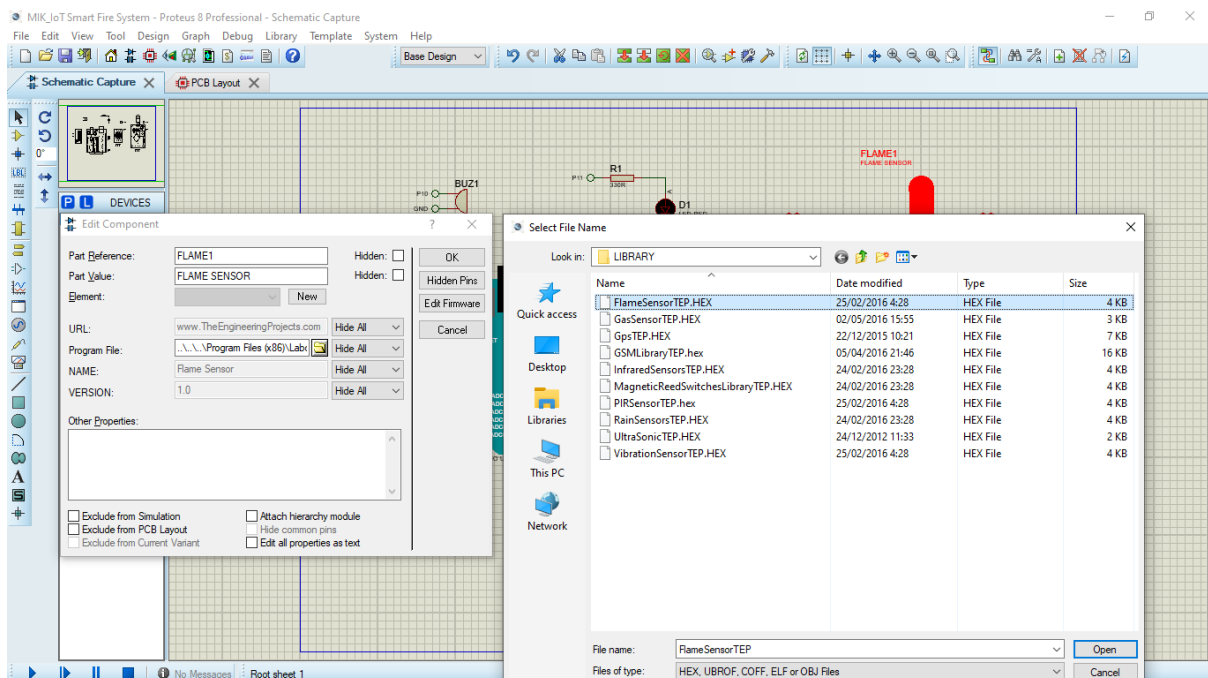
Done compiling.
Sketch uses 8044 bytes (24%) of program storage space. Maximum is 32256 bytes.
Global variables use 459 bytes (22%) of dynamic memory, leaving 1589 bytes for local variables. Maximum is 2048 bytes.
```

5.1.3 Proteus

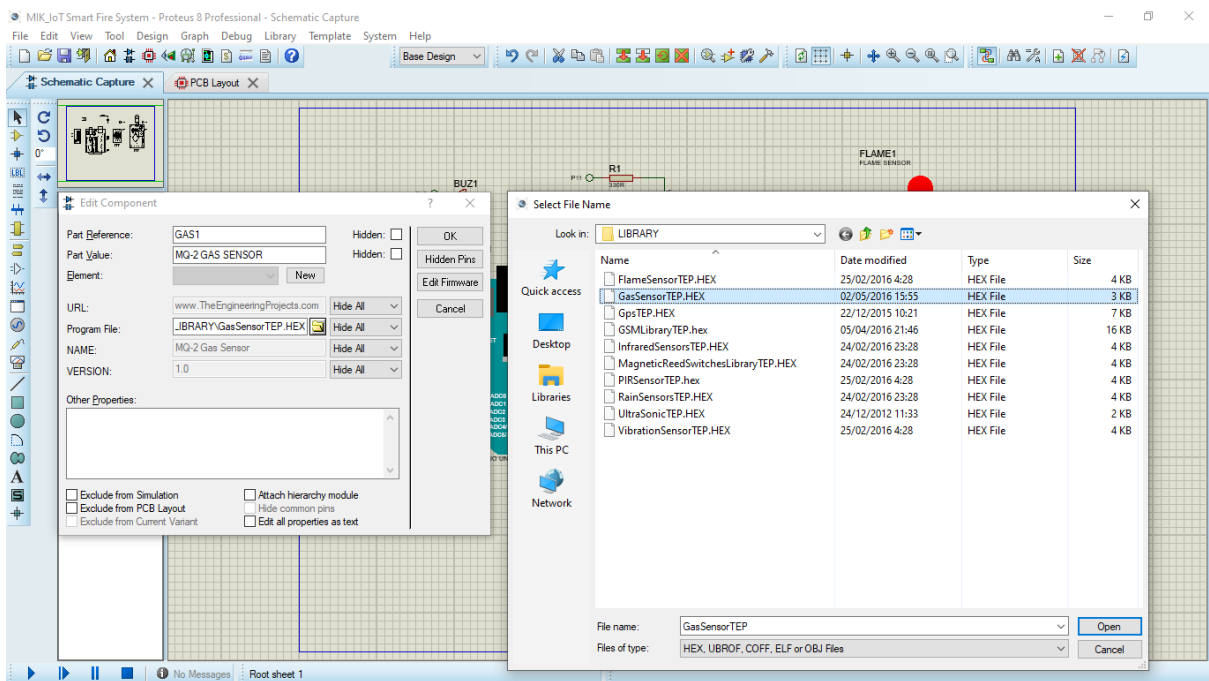
1. Inject program Arduino IDE type **HEX File** dengan format **.hex** ke dalam modul Arduino Uno di Proteus.



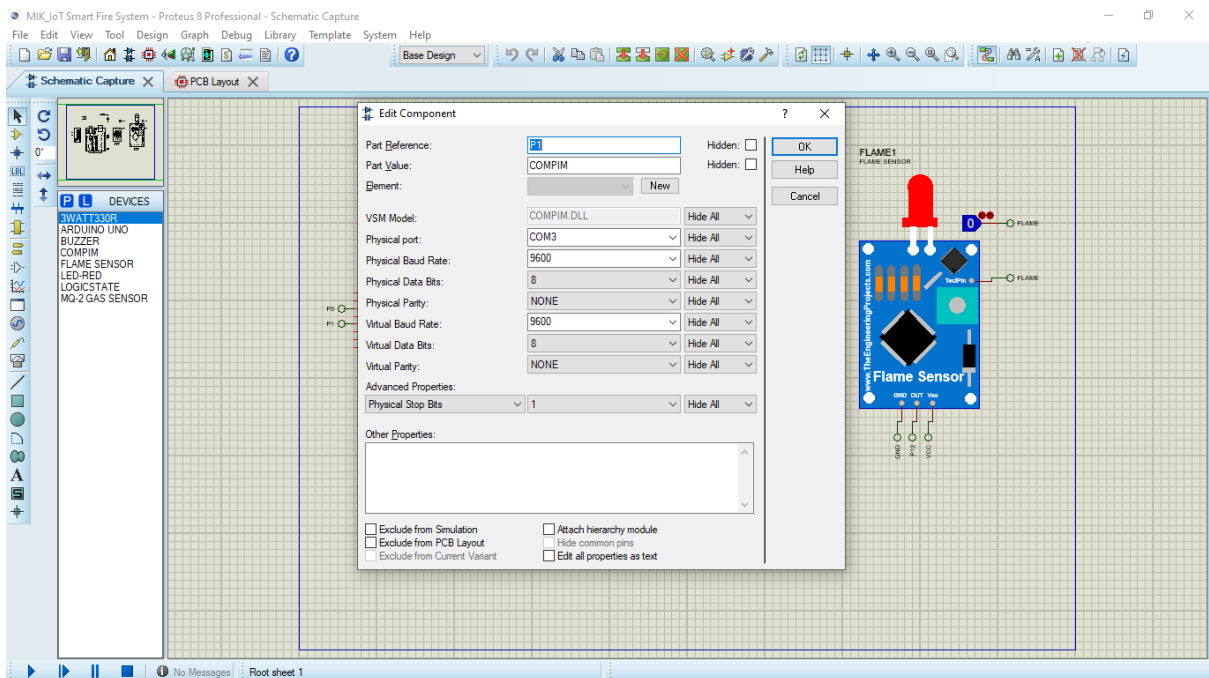
2. Input library **FlameSensorTEP.HEX** ke dalam modul Flame Sensor di Proteus.



3. Input library **GasSensorTEP.HEX** ke dalam modul MQ-2 Gas Sensor di Proteus.

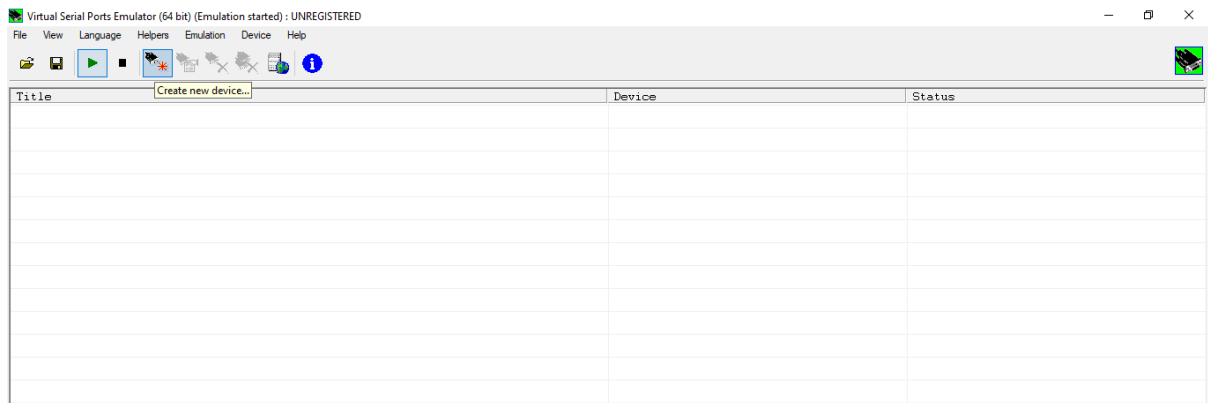


4. Tentukan **Physical Port** pada **COM3**, dan besaran nilai **9600** pada **Physical/Virtual Baud Rate** di Compim untuk koneksi secara serial.

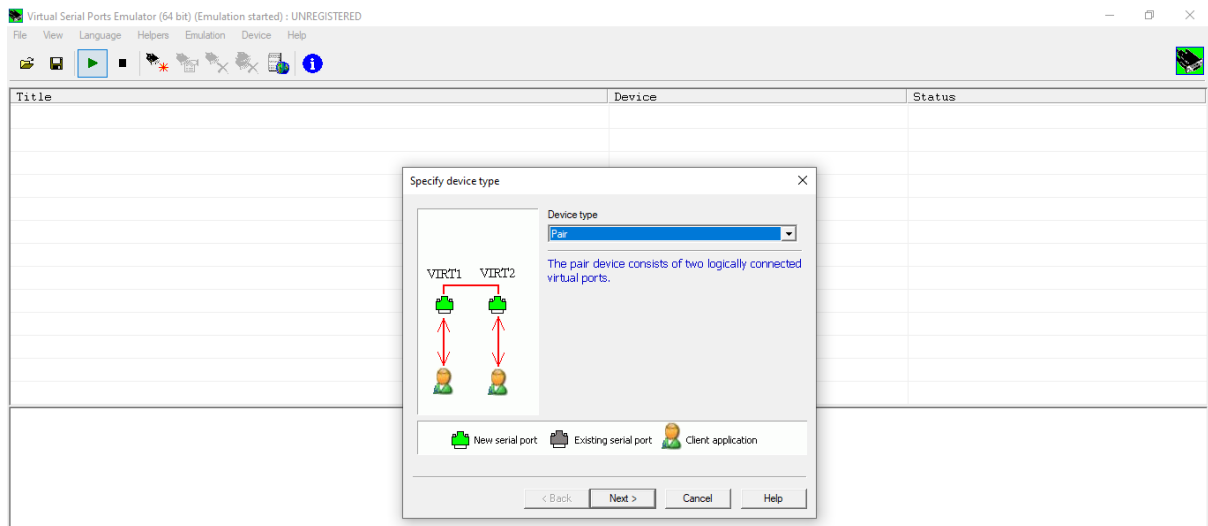


5.1.4 VSPE (Virtual Serial Port Emulator)

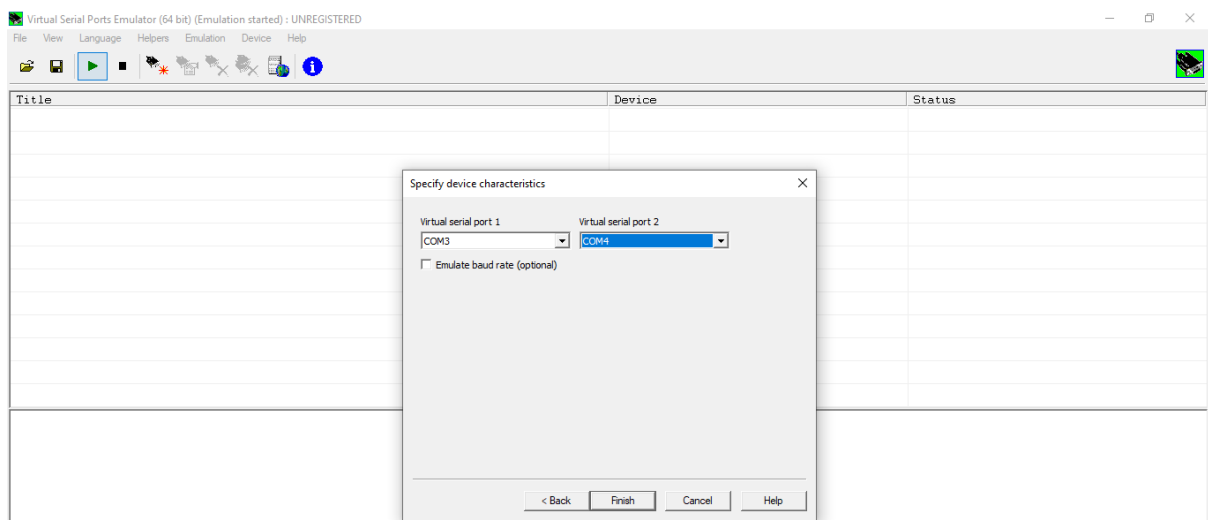
1. Open software VSPE kemudian pilih **Create New Device**.



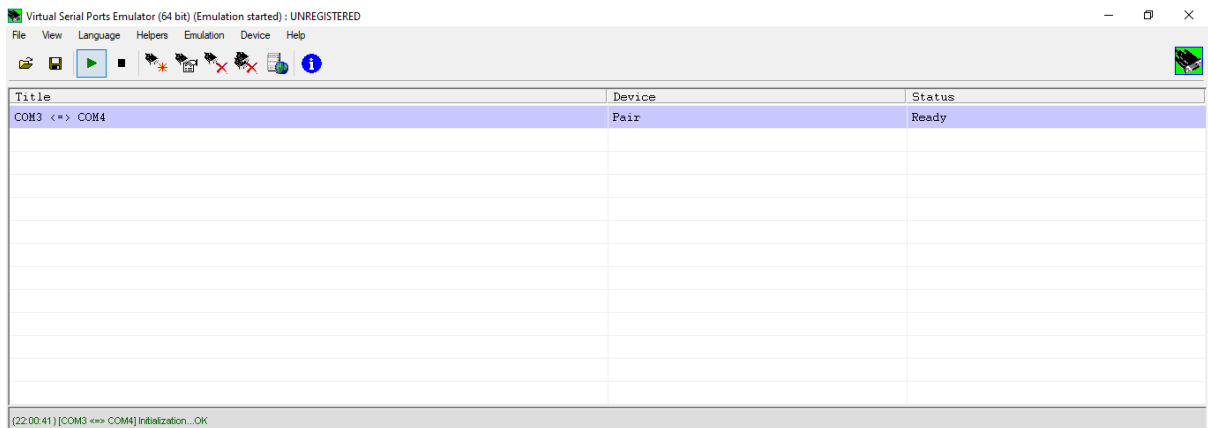
2. Pilih **Device Type** dengan jenis **Pair**, kemudian klik **Next**.



3. Tentukan **Virtual Serial Port**, dalam simulasi ini dipilih **Virtual Serial Port 1** adalah **COM3** dan **Virtual Serial Port 2** adalah **COM4**, kemudian klik **Finish**. Virtual Port COM3 di sini untuk koneksi virtual dengan Arduino IDE, sedangkan Virtual Port COM4 untuk koneksi virtual dengan Blynk.



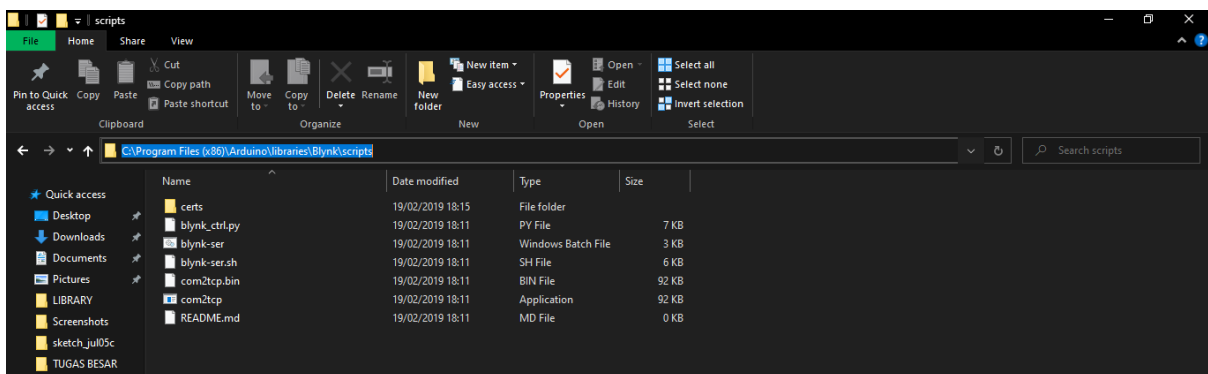
4. Kini Virtual Serial Port COM3 dan COM4 siap digunakan.



5. Open **Command Prompt** untuk menghubungkan koneksi Virtual dengan Blynk.



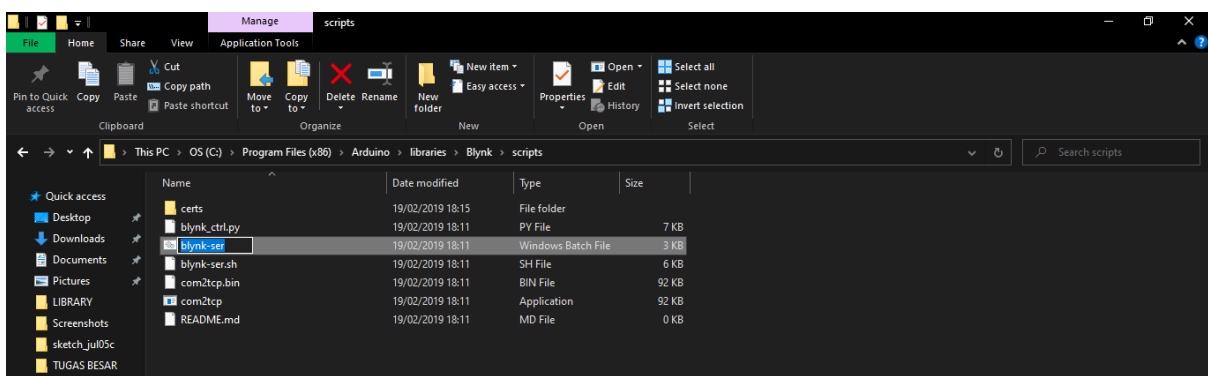
6. Copy directory lokasi penyimpanan file script library Blynk.



7. Input perintah **cd** kemudian paste directory lokasi penyimpanan file script library Blynk tersebut, kemudian tekan ENTER



8. Copy nama file **blynk-ser** dengan type Windows Batch File.



9. Paste nama file **blynk-ser** tersebut pada Command Prompt dan tambahkan perintah **-c COM4** untuk koneksi virtual antara Proteus dengan Blynk, kemudian ENTER.

```
Command Prompt
Microsoft Windows [Version 10.0.19042.1003]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user>cd C:\Program Files (x86)\Arduino\libraries\Blynk\scripts
C:\Program Files (x86)\Arduino\libraries\Blynk\scripts>blynk-ser -c COM4
```

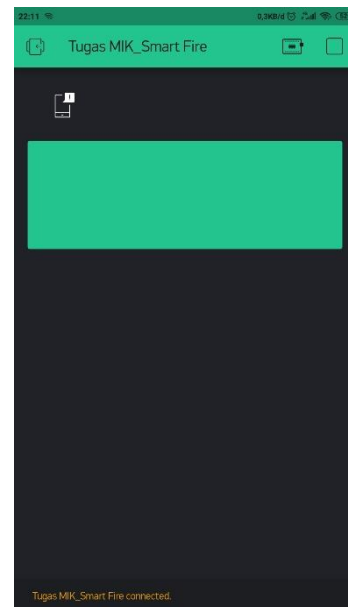
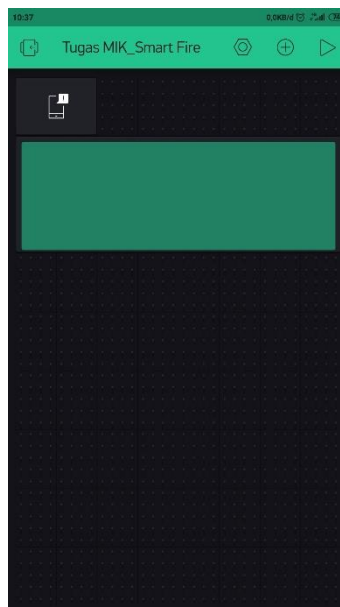
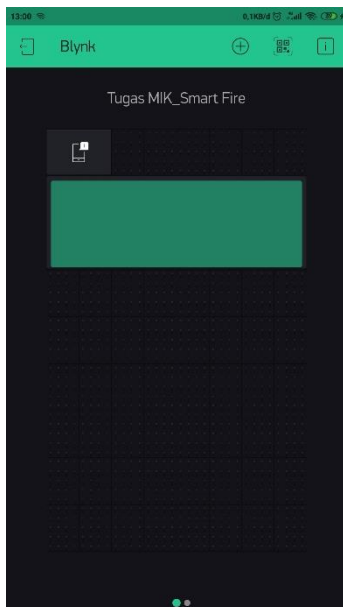
10. Sekarang, simulasi pada Proteus sudah terhubung dengan server Blynk.

```
Command Prompt - blynk-ser -c COM4
Microsoft Windows [Version 10.0.19042.1003]
(c) Microsoft Corporation. All rights reserved.

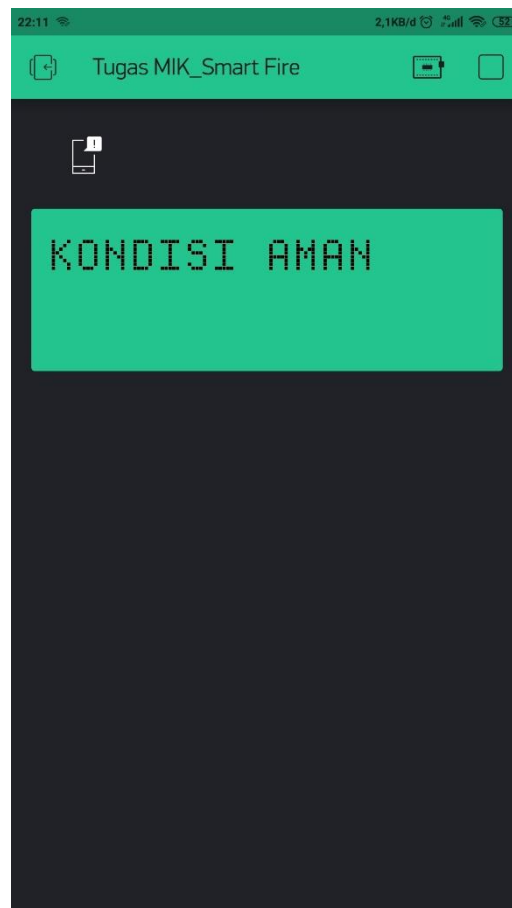
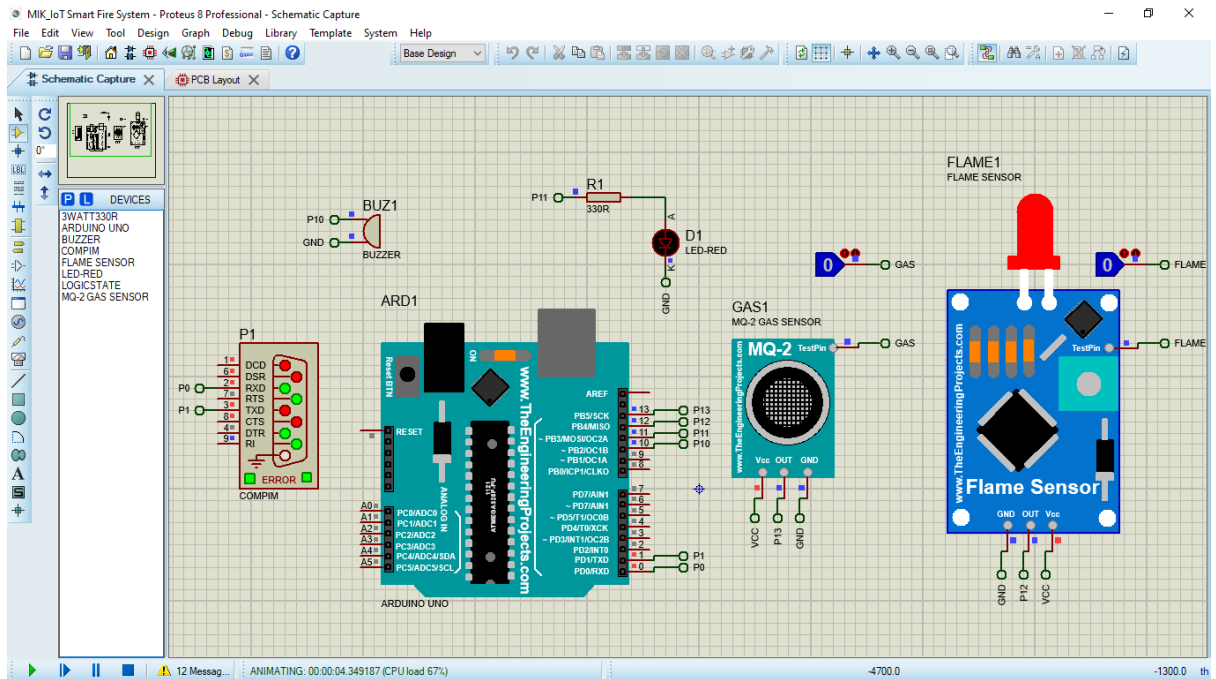
C:\Users\user>cd C:\Program Files (x86)\Arduino\libraries\Blynk\scripts
C:\Program Files (x86)\Arduino\libraries\Blynk\scripts>blynk-ser -c COM4
Connecting device at COM4 to blynk-cloud.com:80...
OpenC0C("\\.\COM4", baud=9600, data=8, parity=no, stop=1) - OK
Connect("blynk-cloud.com", "80") - OK
InOut() START
DSR is OFF
```

5.2 PROSES SIMULASI

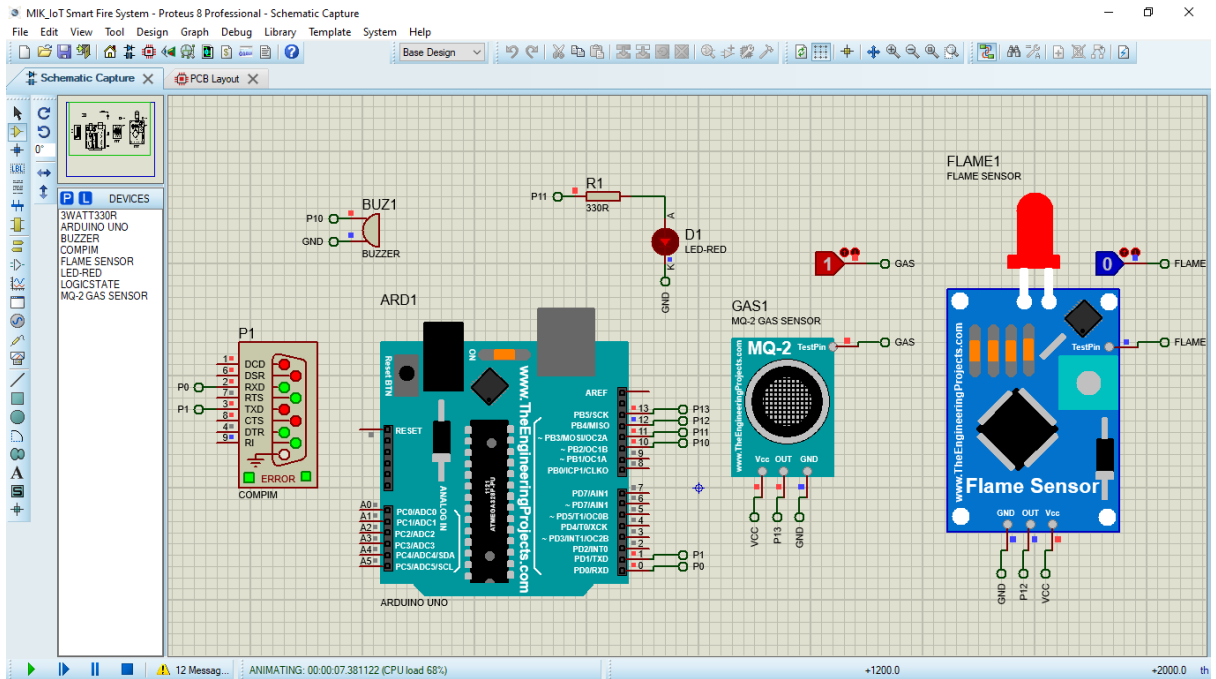
1. Pada aplikasi Blynk, buka project yang sudah dibuat sebelumnya dan akan diuji coba yaitu dengan judul **Tugas MIK_Smart Fire** (kiri). Klik icon **PLAY** ▶ pada sisi pojok kanan atas (tengah), maka project tersebut sekarang sudah terkoneksi dengan Blynk (kanan).



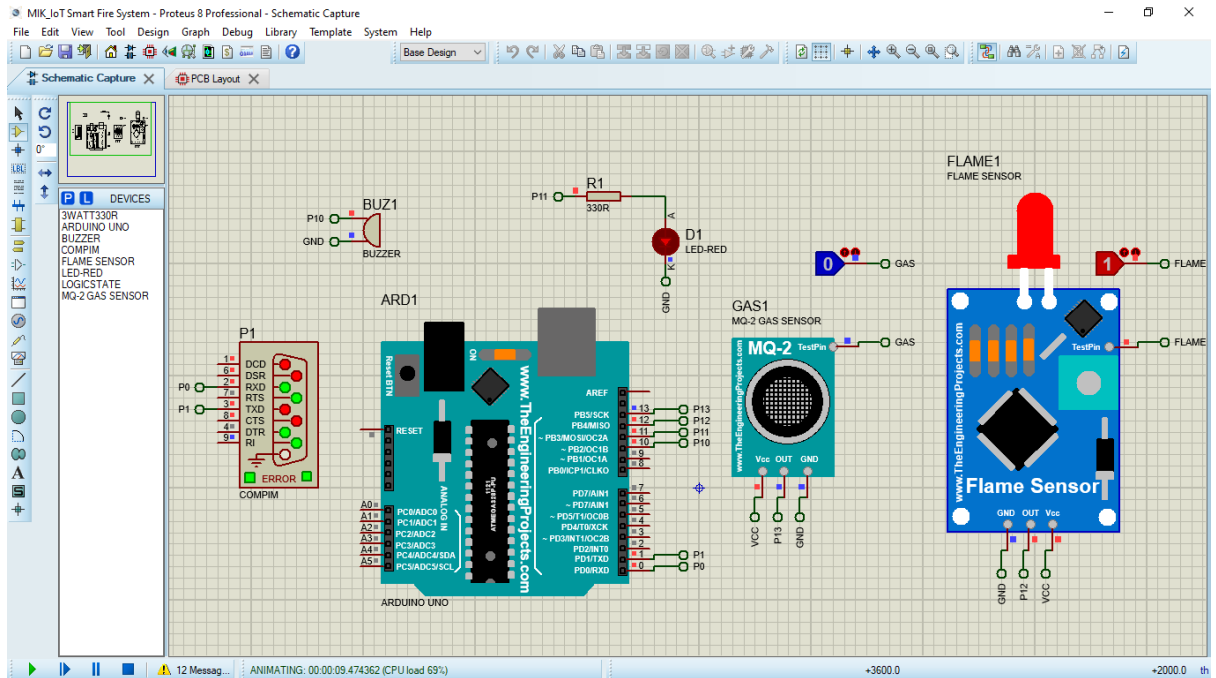
2. Selanjutnya Running rangkaian simulasi yang ada di Proteus, dan hasilnya terlihat pada gambar dibawah ini. Pada kondisi ini, Flame Sensor dan Gas Sensor tersebut diberikan input Logicstate 0 atau dalam keadaan normal yang artinya kedua sensor tersebut disimulasikan belum mendeteksi adanya api ataupun asap, maka LED dan Buzzer tetap OFF dan seketika pada Smartphone akan tampil notifikasi “KONDISI AMAN” melalui LCD Blynk.



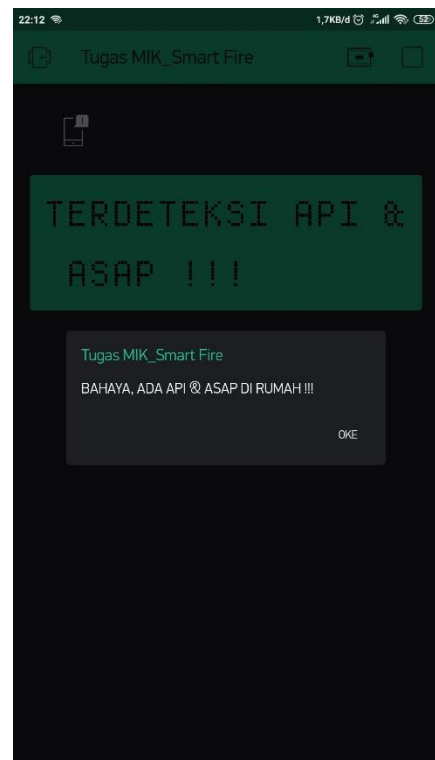
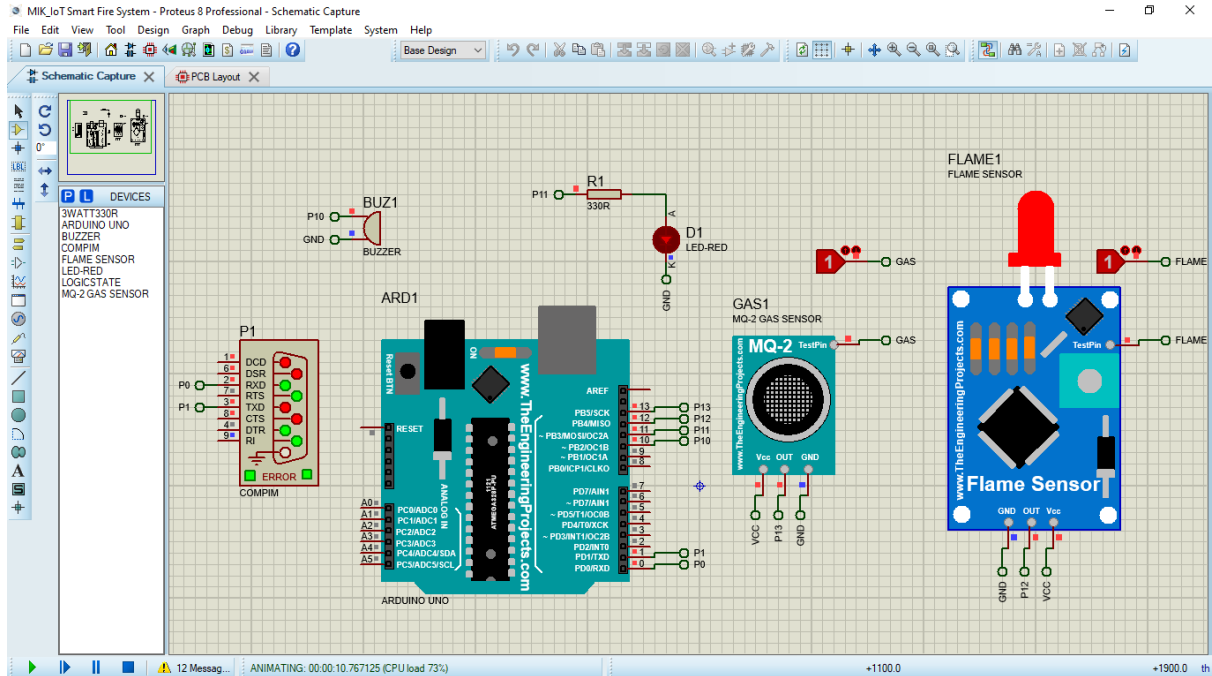
3. Kemudian pada simulasi berikutnya, hanya Gas Sensor yang diberikan input Logicstate 1. Artinya Gas Sensor tersebut disimulasikan mendeteksi adanya asap, maka LED dan Buzzer akan ON, dan seketika pada LCD Blynk di Smartphone akan tampil **“TERDETEKSI ASAP !!!”** dilanjutkan dengan notifikasi po-up **“BAHAYA, ADA ASAP DI RUMAH !!!”**



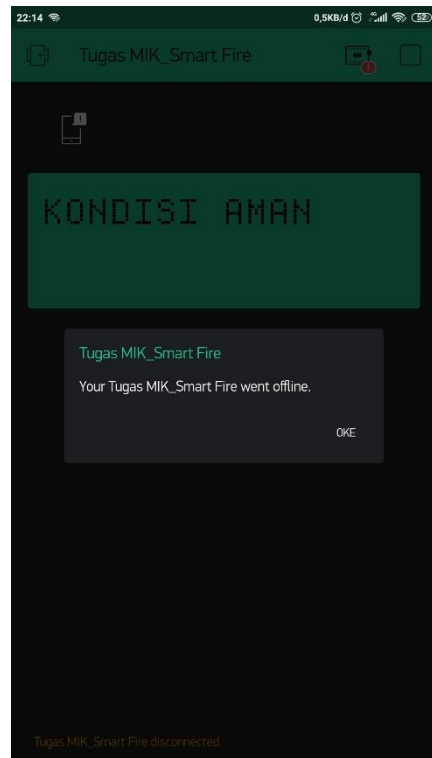
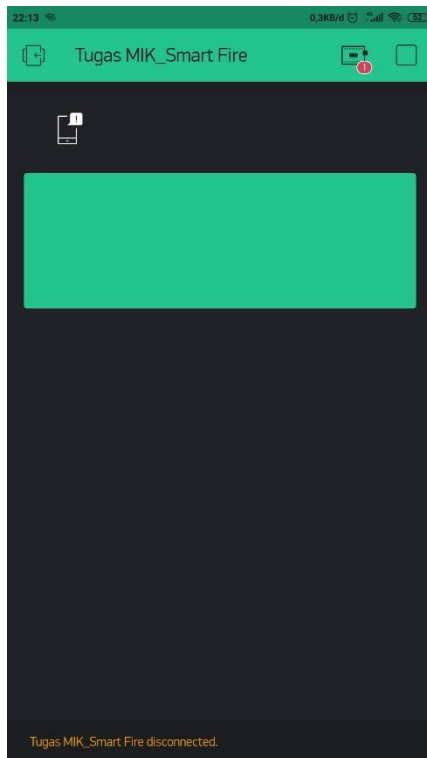
4. Kemudian pada simulasi berikutnya, hanya Flame Sensor yang diinput Logicstate 1. Pada kondisi ini, artinya Flame Sensor tersebut disimulasikan mendeteksi adanya api, maka LED dan Buzzer akan ON, dan seketika pada LCD Blynk di Smartphone akan tampil **“TERDETEKSI API !!!”** beserta notifikasi pop-up **“BAHAYA, ADA API DI RUMAH !!!”**



5. Kemudian pada simulasi berikutnya, masing-masing sensor yaitu Flame Sensor dan Gas Sensor diberikan input Logicstate 1. Artinya Flame Sensor dan Gas Sensor tersebut disimulasikan mendeteksi adanya api serta asap, maka LED dan Buzzer akan ON, dan seketika pada LCD Blynk di Smartphone akan tampil **“TERDETEKSI API DAN ASAP !!!”** dilanjutkan dengan notifikasi po-up **“BAHAYA, ADA API DAN ASAP DI RUMAH !!!”**



6. Apabila Flame Sensor dan Gas Sensor tersebut kembali diberikan Logicstate 0, maka hasilnya akan kembali seperti pada kondisi normal (pada point 2).
7. Hentikan proses running rangkaian simulasi pada Proteus dengan klik Stop. Kemudian LCD Blynk pada Smartphone akan terdapat keterangan bahwa project tersebut *disconnected* dan dilanjutkan dengan notifikasi pop-up.



6. HASIL DAN KESIMPULAN

6.1 HASIL SIMULASI

Berdasarkan seluruh simulasi mulai dari langkah-langkah maupun proses simulasi, maka diperoleh hasil bahwa proses pada rangkaian simulasi bekerja dengan baik dan optimal karena semua output yang dihasilkan sesuai dengan input perintah yang diberikan.

6.2 KESIMPULAN

1. Pengaturan pada menu setting aplikasi Blynk harus ada korelasi dan tidak boleh keliru antara program dengan simulasi.
2. Pada coding Arduino IDE, setiap parameter dan variable yang diinput harus sesuai dengan kebutuhan seperti library dan tools yang tersedia, apabila salah satu library atau tools tidak tersedia maka proses verifikasi program tidak akan valid.
3. Input virtual port yang dibutuhkan juga harus sesuai supaya perangkat dengan aplikasi bisa terkoneksi dengan baik.
4. Simulasi ini perlu dikembangkan dan dimodifikasi menjadi lebih advance supaya dalam penerapannya di lapangan bisa memberikan manfaat yang positif untuk mencegah timbulnya kebakaran

7. DAFTAR PUSTAKA

1. <https://www.youtube.com/watch?v=EvAtsI59Gyc&t=1s>
2. <https://www.youtube.com/watch?v=DprmKRXdkw>
3. https://www.youtube.com/watch?v=Arol_HI-bJo
4. <https://wasiswa.com/cara-menggunakan-virtual-terminal-di-proteus/>
5. <https://catatanonlinemaulana.blogspot.com/2016/08/cara-menggunakan-virtual-serial-port.html>
6. <https://blynk.io/en/getting-started>

TERIMA KASIH