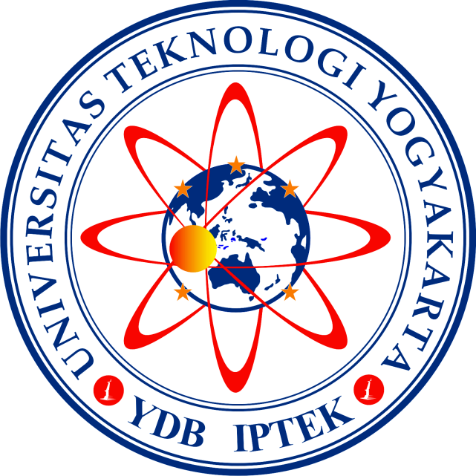
**INTERNET OF THINGS (IOT) PROJECT**

**UJIAN AKHIR SEMESTER**

**MATA KULIAH PEMROGRAMAN MOBILE**



Oleh :

**Nama : Fransiskus Wahyu Sulistyawan**

**NIM : 515-0711-039**

**Kelas : A**

**PROGRAM STUDI TEKNIK ELEKTRO**

**FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO**

**UNIVERSITAS TEKNOLOGI YOGYAKARTA**

**2018**

1. **Landasan Teori**
2. Sistem dengan konsep IOT(Internet of Things)

Istilah Internet of Things (IoT) memang baru di perkenalkan tahun 1999 oleh Kevin Ashton yang merupakan co-founder dan executive director dari Auto-ID Center di MIT, dan juga merupakan seorang inovator dan ahli analisis konsumen. Namun sebenarnya IoT telah dikembangkan sejak awal 1980-an dengan Alat Internet pertama, yaitu Coke Machine di Carnegie Melon University. Para programer dapat terhubung ke mesin melalui Internet, memeriksa status mesin dan menentukan apakah ada atau tidak minuman dingin yang menunggu mereka, tanpa harus pergi ke mesin tersebut.

**Apa itu Internet of Things?** Internet of Things (IoT) adalah teknologi untuk mengendalikan/mengatur/mentransfer data dari benda-benda fisik melalui jaringan internet. IoT telah berkembang dari teknologi nirkabel, Micro Electro Mechanical Systems (MEMS), dan Internet. “A Things” pada Internet of Things dapat didefinisikan dengan banyak subjek, misalkan orang dengan monitor implant jantung, hewan peternakan dengan transponder biochip, sebuah mobil yang telah dilengkapi built-in sensor untuk memperingatkan pengemudi ketika tekanan ban rendah. IoT erat hubungannya dengan istilah machine-to-machine (M2M) di bidang manufaktur dan listrik, perminyakan, dan gas.

Semua Tools dibangun dengan kemampuan komunikasi M2M yang sering disebut dengan sistem cerdas atau “Smart”, (contoh: Smart Label, Smart Meter, Smart Grid Sensor). Samsung dan LG menjadi salah satu contoh perusahaan yang menampilkan teknologi IoT dalam produk rumah tangga seperti mesin cuci, kulkas, microwave, AC dan TV yang dapat dikontrol dengan smartphone atau smartwatch, dan beberapa perusahaan teknologi besar lainnya sudah mulai mengaplikasikan IoT kedalam Smart solution lainnya seperti : Smart Home, Smart Factory, Smart Stadium serta Smart City.

1. Arduino

Arduino adalah pengendali mikro single-board yang bersifat open-source, diturunkan dari Wiring platform, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. Hardwarenya memiliki prosesor Atmel AVR dan softwarenya memiliki bahasa pemrograman sendiri.

Arduino juga merupakan platform hardware terbuka yang ditujukan kepada siapa saja yang ingin membuat purwarupa peralatan elektronik interaktif berdasarkan hardware dan software yang fleksibel dan mudah digunakan. Mikrokontroler diprogram menggunakan bahasa pemrograman arduino yang memiliki kemiripan syntax dengan bahasa pemrograman C. Karena sifatnya yang terbuka maka siapa saja dapat mengunduh skema hardware arduino dan membangunnya.

Arduino menggunakan keluarga mikrokontroler ATMega yang dirilis oleh Atmel sebagai basis, namun ada individu/perusahaan yang membuat clone arduino dengan menggunakan mikrokontroler lain dan tetap kompatibel dengan arduino pada level hardware. Untuk fleksibilitas, program dimasukkan melalui bootloader meskipun ada opsi untuk mem-bypass bootloader dan menggunakan downloader untuk memprogram mikrokontroler secara langsung melalui port ISP.

1. Proteus

Proteus adalah sebuah software yang digunakan untuk mendesain PCB yang juga dilengkapi dengan simulasi PSpice pada level skematik sebelum rangkaian skematik di-upgrade ke PCB untuk memastikan PCB dapat berfungsi dengan semestinya. Proteus mengkombinasikan program ISIS untuk membuat skematik desain rangkaian dengan program ARES untuk membuat layout PCB dari skematik yang dibuat. ISIS disini bukanlah ISIS yang merupakan kumpulan teroris namun digunakan sebagai program untuk perancangan dan pendidikan, sedangkan ARES atau disebut juga Advanced Routing and Editing Software digunakan untuk membuat modul layout PCB.

Proteus sangat berguna untuk desain rangkaian mikrokontroller. Proteus juga berguna untuk belajar elektronika seperti dasar-dasar elektronika sampai pada aplikasi mikrokontroller. Software ini menyediakan banyak contoh aplikasi desain sehingga pengguna bisa belajar dari contoh-contoh yang sudah ada.

Fitur-fitur dari Proteus adalah sebagai berikut :

* + Memiliki kemampuan untuk mensimulasikan hasil rancangan baik digital maupun analog maupun gabungan keduanya.
  + Mendukung instrumen-instrumen virtual seperti voltmeter, ammeter, osciloscope, logic analyser, dan masih banyak lagi.
  + Memiliki model-model peripheral yang interactive seperti LED, tampilan LCD, RS232, dan berbagai jenis library lainnya.
  + Memiliki kemampuan menampilkan berbagi jenis analisis secara grafis seperti transient, frekuensi, noise, distorsi, AC dan DC, dan masih banyak lagi.
  + Mendukung simulasi berbagai jenis microcontroller.
  + Mendukung berbagai jenis komponen-komponen analog.
  + Mendukung open architecture sehingga pengguna dapat memasukkan program seperti C++/ Arduino untuk keperluan simulasi.
  + Mendukung pembuatan PCB yang di-update secara langsung dari program ISIS ke program pembuat PCB-ARES.

1. ENC28J60

ENC28J60 Ethernet Module adalah sebuah Modul Stand-Alone Ethernet Controller berbasis Microchip ENC28J60 yang berfungsi sebagai antarmuka dari serial SPI ke Ethernet. Semua kebutuhan Network Protocol akan ditangani oleh modul ini. Modul ini dapat digunakan langsung oleh semua mikrokontroler yang memiliki interface SPI dengan kecepatan transfer data hingga 20MHz. Selain itu modul ini juga sudah memiliki konektor RJ-45, sehingga memudahkan anda untuk dapat menggunakan modul ini.

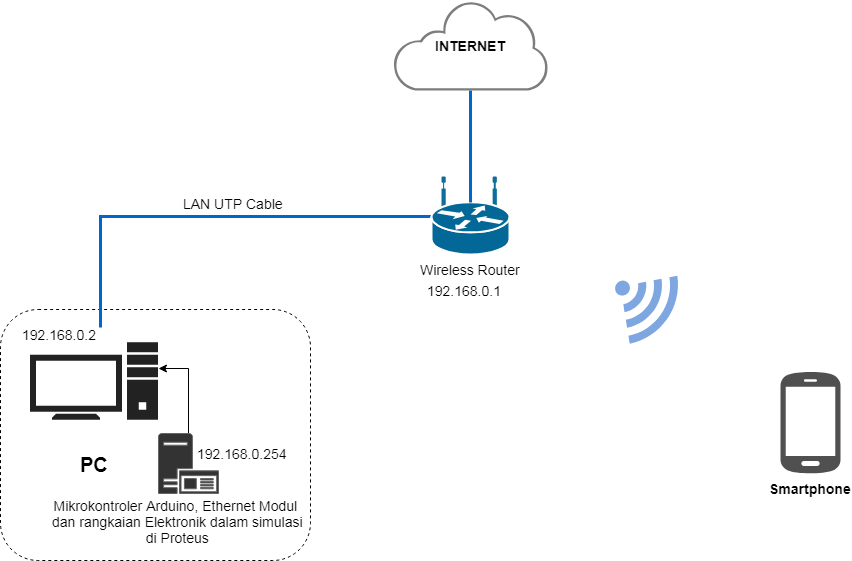
Berikut adalah beberapa fitur dari modul ini :

* Berbasis IC ENC28J60, dengan package SOP28
* Penggunaan Interface SPI
* Menggunakan header 5×2 standar, sehingga memudahkan pemasangan dengan MCU
* Built-in isolation transformer RJ45 connectors HR911102A
* Power LED
* Supply: +3.3 V
* Menggunakan xtal 25Mhz

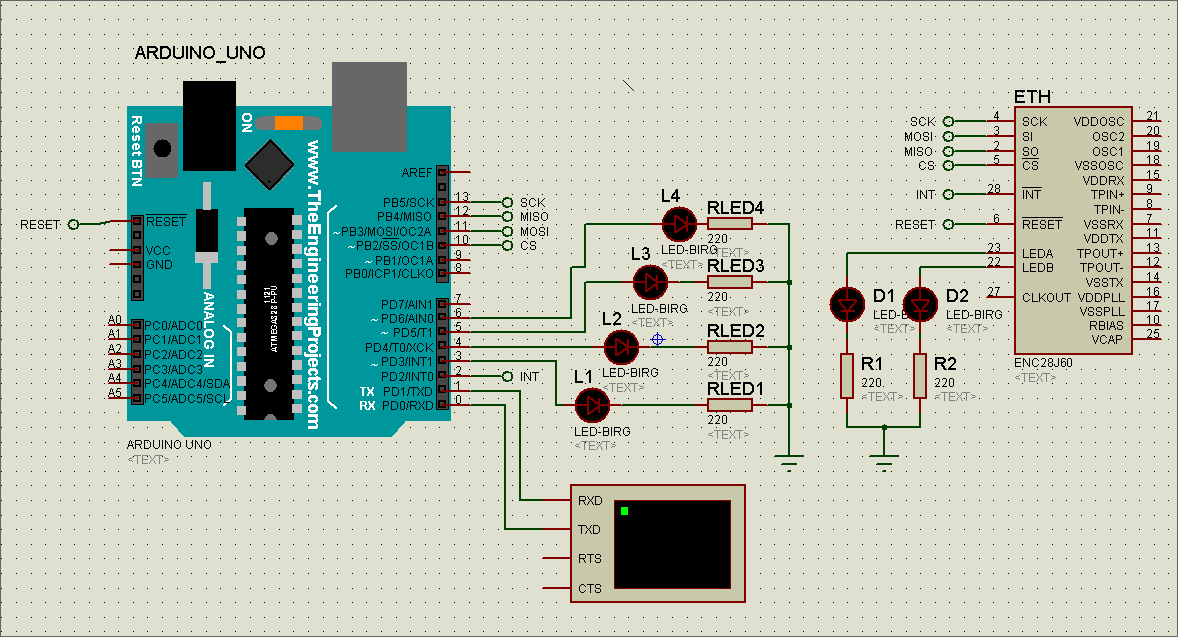
1. **Alat dan Bahan**
2. Adapun software yang diperlukan antara lain :
   1. ISIS Proteus Versi 7
   2. Arduino IDE Versi 1.8.5
   3. WinPcap Driver Software ini dapat diperoleh di <https://www.winpcap.org/install/>
3. Adapun komponen yang diperlukan dalam simulasi di Proteus antara lain :
   1. Library Arduino UNO yang telah ditambahkan ke Proteus, library ini dapat diperoleh di theengineeringprojects.com (<https://www.theengineeringprojects.com/2015/12/arduino-uno-library-proteus.html>)
   2. Ethernet Modul ENC28J60, modul ini akan mensimulasikan Ethernet Card fisik dari komputer kita ke Proteus, sehingga modul ini ter-bridge ke LAN Card komputer.
   3. 6 Buah active LED
   4. 6 Buah Resistor 220 Ohm
   5. Virtual Terminal, untuk menampilkan output parameter melalui interface serial Arduino.
4. Adapun library yang diperlukan dalam Arduino IDE
   1. Library EtherCard.h, dapat diperoleh di GitHub (<https://github.com/jcw/ethercard>)
5. **Langkah Kerja**
6. Proyeksi Hasil

Membuat sistem yang mengimplementasikan konsep IOT untuk mengendalikan 4 buah lampu melalui aplikasi mobile.

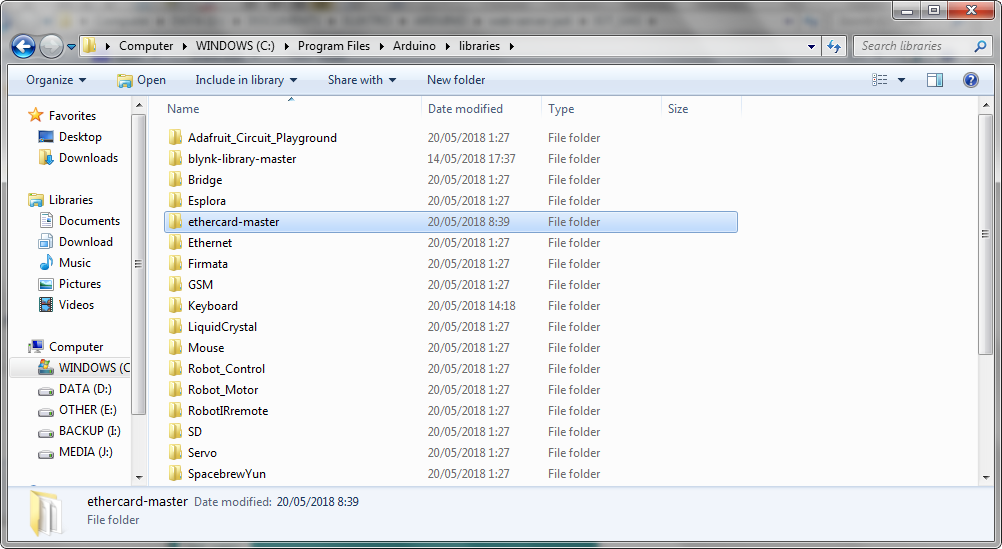
1. Rencana topologi jaringan



1. Wiring diagram di Proteus



1. Melakukan instalasi WinPcap driver, driver ini diperlukan agar Modul Ethernet ENC28J60 dapat berjalan di simulasi. Software ini dapat diperoleh di (<https://www.winpcap.org/install/>)
2. Menambahkan library EtherCard.h ke Arduino IDE. Secara manual : yaitu dengan meng-extract file .ZIP dari library EtherCard.h yang telah kita download kemudian memindahkannya ke folder library Arduino IDE (C:\Program Files\Arduino\libraries)



1. Melakukan pemrograman pada Arduino IDE.

Hal-hal yang perlu diperhatikan dalam langkah ini antara lain yaitu :

1. Memasukkan library EtherCard.h kedalam program (#include <EtherCard.h>)
2. Konfigurasi alamat IP (DHCP/Statis), dalam percobaan ini saya menggunakan IP statis (192.168.0.254) sesuai dengan rencana topologi.
3. Pin input dan output pada program harus sesuai dengan wiring diagram yang telah dibuat di Proteus.

Berikut merupakan coding program IOT yang telah saya buat :

/\*Simple IOT Project

\* Ujian Akhir Semester (UAS)

\* Created by : Fransiskus Wahyu S

\* NIM : 515-0711-039 / TE / UTY

\* Menggunakan library EtherCard.h untuk ethernet modul dengan chip ENC28J60

\* Yogyakarta, Mei 2018

\*/

#include <EtherCard.h>

#define REQUEST\_RATE 5000

#define STATIC 1 //1 untuk aktifkan seting IP statis, jika 0 DHCP client aktif

#if STATIC

// alamat IP (jika diatur statis)

static byte myip**[]** **=** **{** 192**,**168**,**0**,**254 **};**

static byte gwip**[]** **=** **{** 192**,**168**,**0**,**1 **};**

#endif

char**\*** on **=** "ON"**;**

char**\*** off **=** "OFF"**;**

boolean L1Stat**,** L2Stat**,** L3Stat**,** L4Stat**;**

char**\*** L1Label**;** char**\*** L2Label**;** char**\*** L3Label**;** char**\*** L4Label**;**

char**\*** L1linkLabel**;** char**\*** L2linkLabel**;** char**\*** L3linkLabel**;** char**\*** L4linkLabel**;**

int L1**=**3**,**L2**=**4**,**L3**=**5**,**L4**=**6**;** //mendeklarasikan pin Lampu 1 hingga 4

//alamat fisik ethernetcard (MAC Address)

static byte mac**[]** **=** **{** 0x74**,**0x69**,**0x69**,**0x2D**,**0x30**,**0x31 **};**

byte Ethernet**::**buffer**[**800**];** // tcp/ip send and receive buffer

void setup**(){**

Serial**.**begin**(**9600**);**

Serial**.**println**(**"\n[HelloWorld!]"**);**

setIP**();** //memanggil fungsi setIP

int a**;**

**for** **(**a**=**3**;**a**<=**6**;**a**++){**

pinMode **(**a**,**OUTPUT**);**

digitalWrite**(**a**,** LOW**);**

**}**

//menentukan kondisi awal

L1Stat = false; L1Label = off; L1linkLabel = on;

L2Stat = false; L2Label = off; L2linkLabel = on;

L3Stat = false; L3Label = off; L3linkLabel = on;

L4Stat = false; L4Label = off; L4linkLabel = on;

}

void loop(){

word len = ether.packetReceive();

word pos = ether.packetLoop(len);

if(pos) {

//mendefinikan kondisi sesuai dengan parameter acuan yaitu laman web

if(strstr((char \*)Ethernet::buffer + pos, "GET /?ON1") != 0) {

L1Stat = true;

L1Label = on;

L1linkLabel = off;

} else if (strstr((char \*)Ethernet::buffer + pos, "GET /?OFF1") != 0) {

L1Stat = false;

L1Label = off;

L1linkLabel = on;

} else if (strstr((char \*)Ethernet::buffer + pos, "GET /?ON2") != 0) {

L2Stat = true;

L2Label = on;

L2linkLabel = off;

} else if (strstr((char \*)Ethernet::buffer + pos, "GET /?OFF2") != 0) {

L2Stat = false;

L2Label = off;

L2linkLabel = on;

} else if (strstr((char \*)Ethernet::buffer + pos, "GET /?ON3") != 0) {

L3Stat = true;

L3Label = on;

L3linkLabel = off;

} else if (strstr((char \*)Ethernet::buffer + pos, "GET /?OFF3") != 0) {

L3Stat = false;

L3Label = off;

L3linkLabel = on;

} else if (strstr((char \*)Ethernet::buffer + pos, "GET /?ON4") != 0) {

L4Stat = true;

L4Label = on;

L4linkLabel = off;

} else if (strstr((char \*)Ethernet::buffer + pos, "GET /?OFF4") != 0) {

L4Stat = false;

L4Label = off;

L4linkLabel = on;

}

//melakukan output sesuai dengan kondisi (status) pada variabel L1 hingga L4

digitalWrite(L1, L1Stat);

digitalWrite(L2, L2Stat);

digitalWrite(L3, L3Stat);

digitalWrite(L4, L4Stat);

//menampilkan antarmuka ke pengguna menggunakan fitur Buffer Filler

BufferFiller bfill = ether.tcpOffset();

bfill.emit\_p(PSTR("HTTP/1.0 200 OK\r\n"

"Content-Type: text/html\r\nPragma: no-cache\r\n\r\n"

"<html><head><meta name='viewport' content='width=200px'/>"

"<body style='background-color:#717272;'>"

"<title>Fransiskus IOT Project</title>"

//memuat file css yang ada di server (github saya)

"<link rel='stylesheet' type='text/css' href='https://rawgit.com/franzwahyu/test/master/tombol.css'>"

"</head>"

"<h3 style='text-align:center'>Kontrol Lampu Online</h3>"

"<div style='position:absolute;width:200px;height:450px;top:30%;left:50%;margin:-100px 0 0 -100px'>"

"<br><div style='text-align:center'>"

//menampilkan user interface tombol

"<div style='font:bold 14px verdana;text-align:center'>Lampu 1 $S</div>"

"<a href='/?$S1'><button class='button'>Turn $S!</button></a><br>"

"<br><div style='font:bold 14px verdana;text-align:center'>Lampu 2 $S</div>"

"<a href='/?$S2'><button class='button'>Turn $S!</button></a><br>"

"<br><div style='font:bold 14px verdana;text-align:center'>Lampu 3 $S</div>"

"<a href='/?$S3'><button class='button'>Turn $S!</button></a><br>"

"<br><div style='font:bold 14px verdana;text-align:center'>Lampu 4 $S</div>"

"<a href='/?$S4'><button class='button'>Turn $S!</button></a>"

"<br><br>"

"<div style='font:bold 12px arial'>Kontrol Lampu v1.0<br>By:Fransiskus Wahyu S<div>"

"</div></div></body></html>"

"<meta http-equiv='refresh' content='3;url=/'>"

),

//mendefinisikan isi format buffer filler (sesuai urutan $S dipanggil)

L1Label, L1linkLabel, L1linkLabel,

L2Label, L2linkLabel, L2linkLabel,

L3Label, L3linkLabel, L3linkLabel,

L4Label, L4linkLabel, L4linkLabel

);

ether.httpServerReply(bfill.position());

}

}

void setIP(){

ether.begin(sizeof Ethernet::buffer, mac);

#if STATIC

ether.staticSetup(myip, gwip);

#else

Serial.println("Searching DHCP");

if (!ether.dhcpSetup()) Serial.println("DHCP failed");

if (ether.dhcpSetup()) Serial.println("DHCP Success");

#endif

ether.printIp("IP : ", ether.myip);

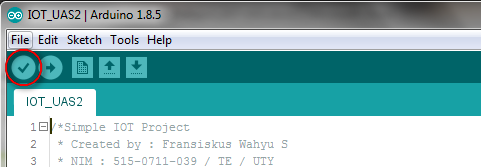
ether.printIp("GW : ", ether.gwip);

ether.printIp("MASK : ", ether.netmask);

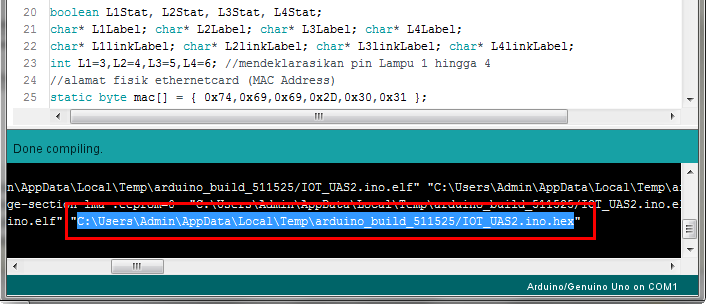
ether.printIp("DNS : ", ether.dnsip);

}

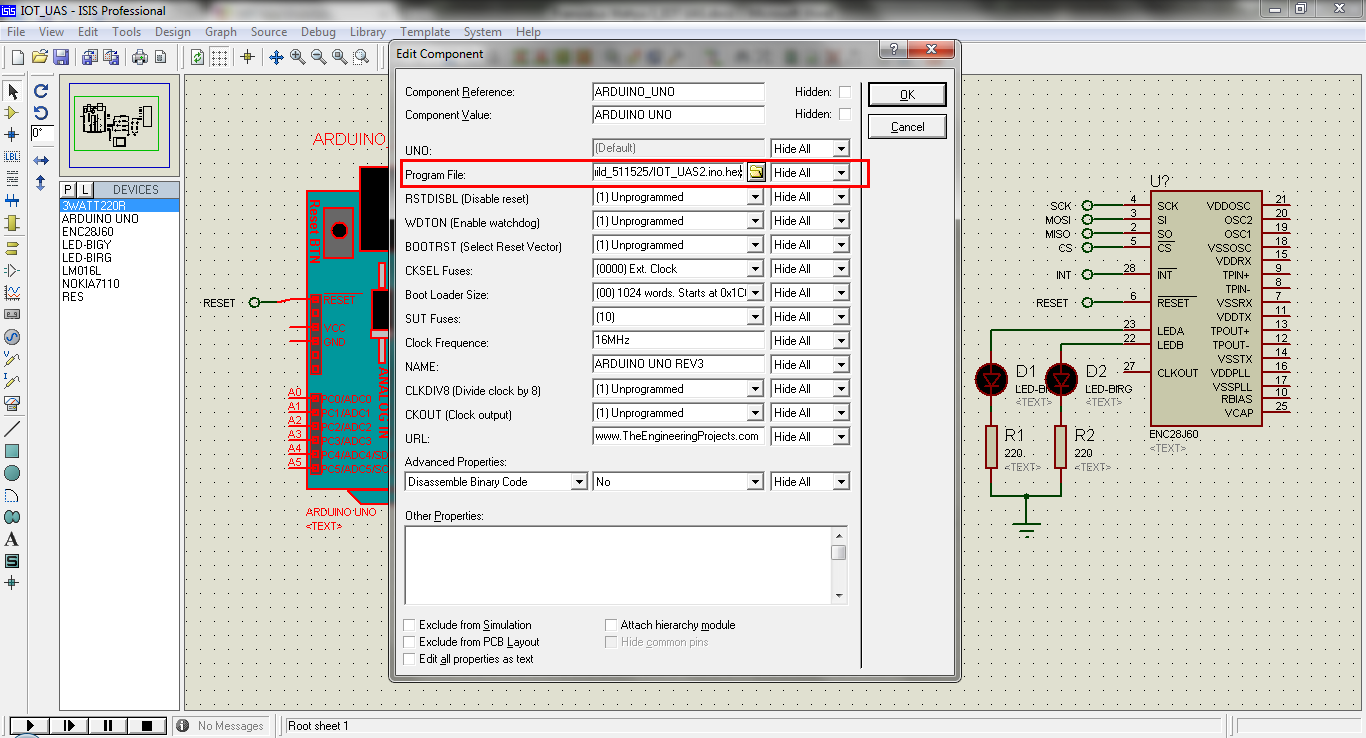
1. Melakukan compile program yang telah dibuat.



1. Melakukan attach file (upload) hasil compile (.hex) ke arduino uno di Proteus, kemudian melakukan running simulasi di Proteus. File hasil compile Arduino akan tersimpan pada directory temp di Windows, alamat file tersebut dapat dilihat pada Arduino IDE seperti gampar dibawah, untuk menyalin alamat tersebut gunakan perintah ctrl+c.

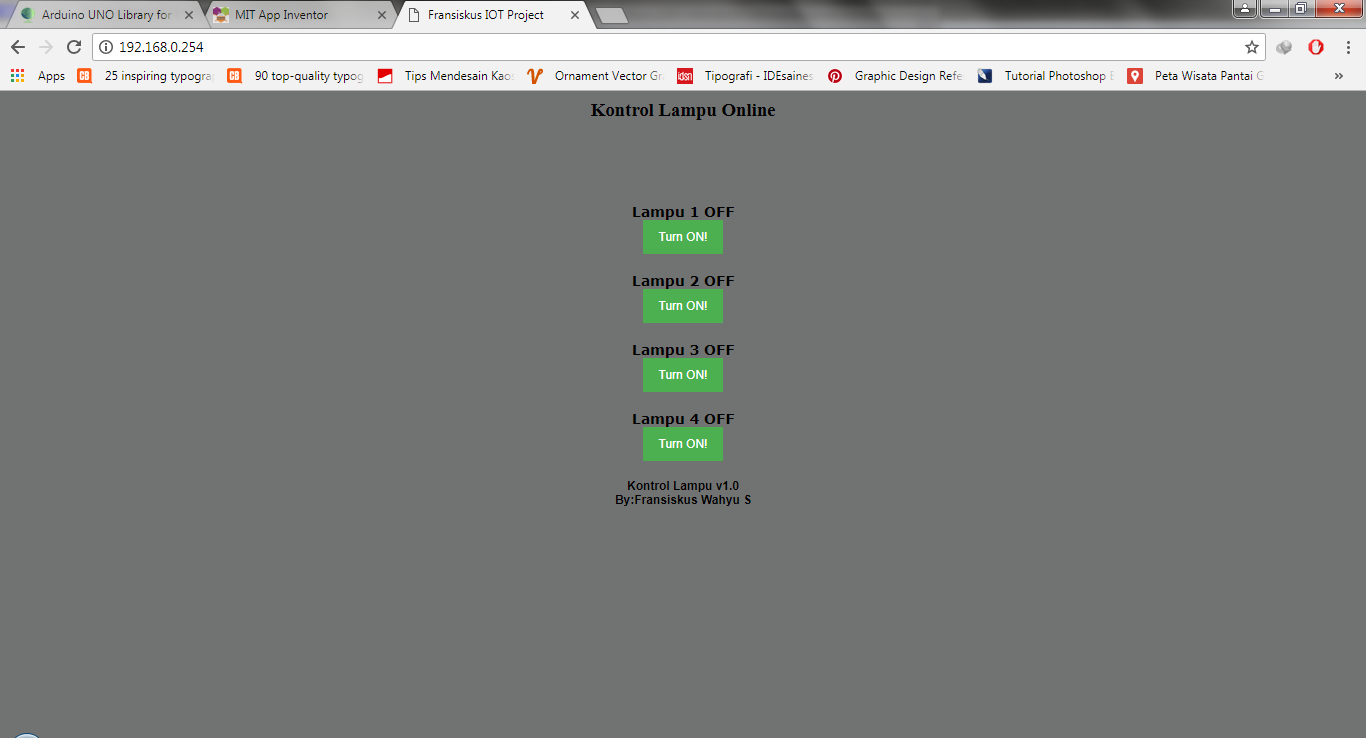


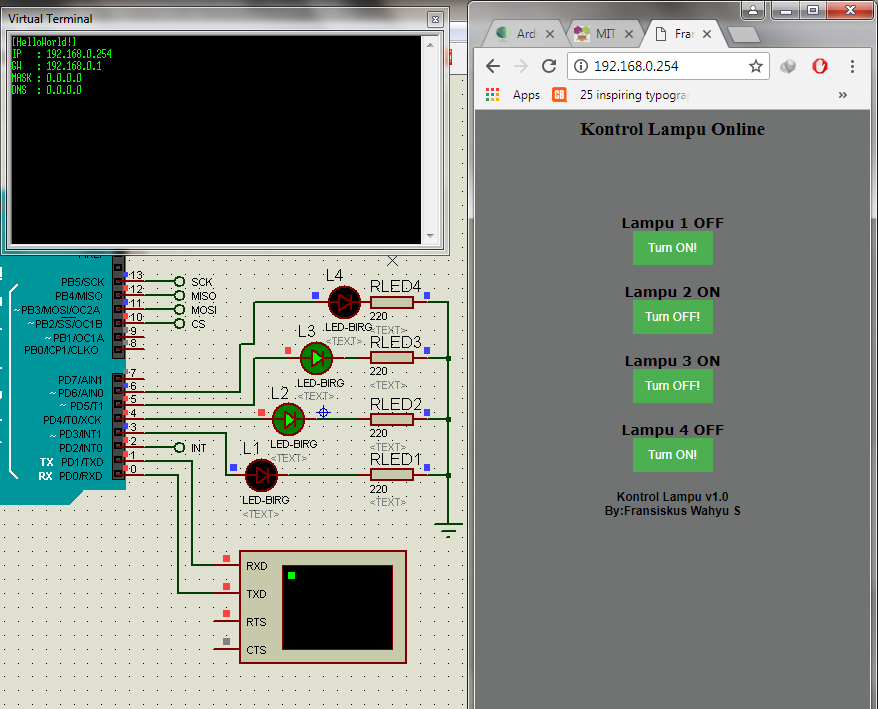
Kemudian melakukan edit component pada Arduino UNO (double click pada komponen) kemudian menambahkan file .hex tersebut ke kolom program file (ctrl+v), kemudian klik OK.

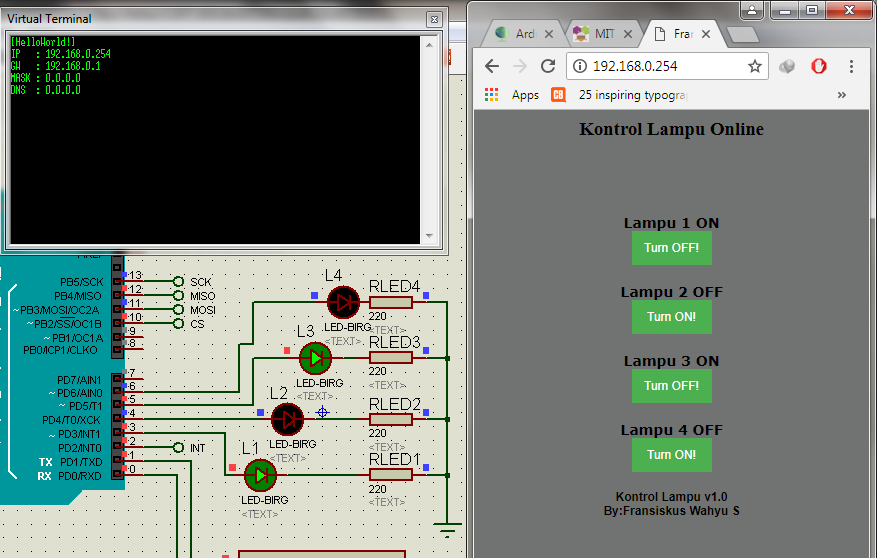


Setelah program file ditambahkan, jalankan simulasi menggunakan tombol play.

1. Melakukan uji akses melalui Web Browser, dan melakukan pengamatan secara realtime pada Proteus. Jika berhasil maka lampu (LED) dapat dikontrol melalui Web.
   1. Tampilan antarmuka Web.

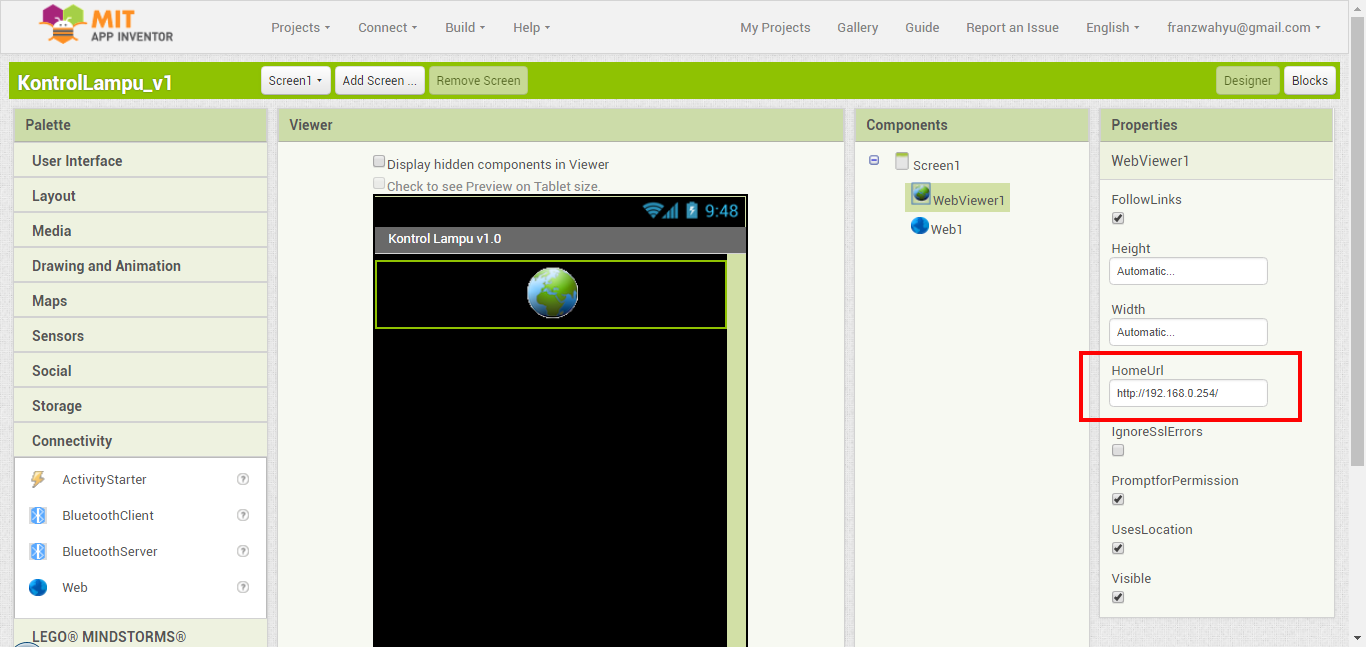


* 1. Kondisi lampu 1 OFF, lampu 2 ON, lampu 3 ON, lampu 4 OFF.
  2. Kondisi lampu 1 ON, lampu 2 OFF, lampu 3 ON, lampu 4 OFF



1. Membuat interface aplikasi mobile Android, dalam percobaan ini saya menggunakan MIT AppInventor untuk membuat aplikasi mobile, online software ini dapat diakses melalui browser dengan alamat : <http://ai2.appinventor.mit.edu/>.

Free online software ini dapat digunakan dengan cukup mudah, yaitu dengan drag and drop palette/komponen yang diperlukan, dalam percobaan ini saya hanya menambahkan WebViewer dan connectivity Web, agar web yang tertanam pada arduino dapat diakses melalui aplikasi mobile ini.



Pada properties WebViewer arahkan HomeUrl-nya ke IP dari Arduino kita.

1. Melakukan pengujian aplikasi pada device Android, pastikan android terhubung ke jaringan wifi (sesuai dengan topologi).

Demo terdapat pada youtube : (<https://youtu.be/JR7k-kUb1v4>)