

# SECURITY SISTEM KELISTRIKAN MOTOR INDUSTRI MENGUNAKAN MEDIA BLUETOOTH BERBASIS MIKROKONTROLER ATMEGA 328

Al-husain<sup>1</sup>  
Ferry Sudarto<sup>2</sup>  
Chandra Jayusman<sup>3</sup>

email : alhusain@raharja.info<sup>1</sup>, ferry.sudarto@raharja.info<sup>2</sup>,  
chandra.jayusman@raharja.info<sup>3</sup>

## ABSTRAKSI

Perkembangan ilmu pengetahuan dan teknologi telah mampu mendorong manusia dalam mengatasi dan menyelesaikan permasalahan yang ada di dalam kehidupan sehari-hari. Salah satu teknologi yang berkembang saat ini adalah Mikrokontroler. Mikrokontroler adalah sebuah chip yang dapat melakukan pemrosesan data secara digital sesuai dengan perintah bahasa pemrograman yang diberikan..Maka saat ini Mikrokontroler banyak diaplikasikan pada kehidupan sehari-hari maupun industri, salah satunya untuk "Security Sistem Kelistrikan Motor Industri Menggunakan Media Bluetooth Berbasis Mikrokontroler ATMEGA 328" ini merupakan sebuah alat sederhana yang dapat berfungsi secara otomatis berdasarkan input dari *handphone* Android yang dikirim melalui *Bluetooth* ke dalam IC Mikrokontroler. Mikrokontroler pada alat ini bertujuan untuk mengendalikan atau mengontrol masukan yang diterima dan diproses, sehingga menghasilkan output tujuan dari keamanan sistem kelistrikan motor industri ini yaitu untuk menciptakan suatu alat yang dapat membantu mengoperasikan suatu sistem keamanan pada mesin industri mengontrol panel dengan *handphone* secara otomatis menggunakan media *Bluetooth*, dengan begitu *Bluetooth* sebagai penghubung antara *smartphone* dengan perangkat yang akan menerjemahkan perintah yang dikirim oleh *smartphone* untuk dieksekusi Mikrokontroler.

**Kata kunci:** *Security* , Motor Industri , Mikrokontroler ATmega328, *Bluetooth*, *Handphone* Android.

## ABSTRACT

*Developments in science and technology have been able to encourage people to address and resolve the problems that exist in everyday life. One of the technologies developed at this time is a microcontroller. Microcontroller is a chip that can perform digital data processing in accordance with a command programming language that so givent currently Microcontroller widely applied in daily life and industry, one for "Electric Motor Industry Security System Using Bluetooth Media-Based Microcontroller ATMEGA 328 "this is a simple tool which can function automatically based on the input of Android phones are sent via Bluetooth to the Microcontroller IC. Microcontroller on this tool aims to control or control the input received and processed, resulting in the output destination of the motor electrical system security industry is to create a tool that can help operate a security system on industrial machinery control panel with automatic phone using Bluetooth media , with so Bluetooth as a liaison between a smartphone with a device that will translate commands sent by a smartphone to be executed Microcontroller.*

**Keywords :** *Security, Motor Industry, ATmega328 microcontroller, Bluetooth, Mobile Android*

## PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi pada masa sekarang terus diikuti oleh sebagian bahkan hampir semua kalangan. Tidak lepas dari hal yang di atas perkembangan sistem yang umumnya berbasis komputer dan sistem terkontrol yang menggunakan mikrokontroler sudah sangat maju. Dengan kemajuan teknologi tersebut pada era sekarang ini komunikasi bukan hanya digunakan untuk komunikasi antar sesama manusia saja, melainkan antara manusia dengan alat-alat kontrol, seperti sistem pengontrolan mesin industri dengan menggunakan *handphone* yang berbasis Android dengan menggunakan media *Bluetooth* yang berbasis Mikrokontroler. Perlunya pemahaman tentang komponen-komponen elektronika sangat dibutuhkan. Pada perancangan *Embedded System* ini juga digunakan untuk mengakses mesin industri dengan menggunakan *Handphone* yang berbasis Android.

Pada perkembangan teknologi sekarang sudah berbagai jenis sistem kontrol yang dapat kita rancang, khususnya pada penggunaan *Bluetooth* dan *handphone* Android untuk melakukan akses terhadap mesin dimana pengoperasian tersebut membutuhkan operator yang bertanggung jawab terhadap apa yang terjadi pada mesinnya. Adapun keuntungan yang didapatkan dengan menggunakan teknologi *handphone* yang berbasis Android adalah bagaimana sistem yang terdapat pada mesin industri dapat dihubungkan dengan *handphone* dengan melalui media *Bluetooth* yang berbasis Mikrokontroler ATmega328.

Oleh karena itu dengan menggunakan metode kode akses yang telah dikonfigurasi ke dalam pemrograman Mikrokontroler ATmega328, diharapkan dapat menjadi pengaman sehingga tidak sembarang orang dapat mengoperasikan mesin tersebut, apabila operator ingin mengoperasikan mesin tersebut baik itu

menghidupkan atau mematikan, yang bisa mengoperasikannya hanyalah operator yang telah mempunyai kode akses tersebut, sehingga pengontrolan terhadap mesin tersebut dapat tercapai dengan baik.

## PERMASALAHAN

Dalam menjalankan pengamanan terhadap mesin industri disuatu perusahaan dibutuhkan sebuah sistem untuk kendali mesin industri tersebut. Untuk menjalankan sistem tersebut dibutuhkan cara bagaimana menanamkan media *Bluetooth* kedalam sistem Mikrokontroler Atmega 328, agar dapat terkoneksi dengan *handphone*. Sehingga sistem tersebut dapat berjalan sebagaimana mestinya.

## LITERATURE REVIEW

Berikut ini adalah penelitian yang telah dilakukan dan memiliki korelasi yang searah dengan penelitian yang akan dibahas dalam Tugas Akhir ini diantaranya lain:

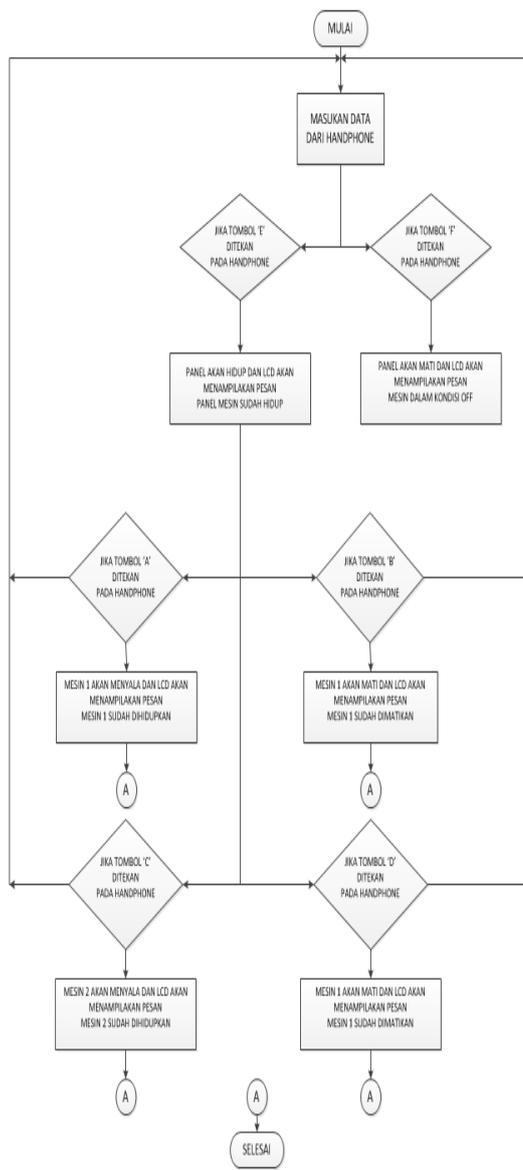
1. Penelitian yang dilakukan Reza Handaru dari Perguruan Tinggi Raharja, Indonesia pada tahun 2014 yang berjudul "Pemanfaatan Sensor Accelerometer pada Smartphone Android Sebagai Kendali Pagar Rumah Melalui Bluetooth". Penelitian ini membahas tentang mengendalikan pintu pagar rumah, yang dapat dilakukan dengan memanfaatkan Sensor Accelerometer. Dengan menggunakan aplikasi Android Sensor Accelerometer mengirimkan String tertentu kepada mikrokontroler ATmega 8 melalui komunikasi Bluetooth HC-05 untuk membuka pintu pagar rumah.
2. Penelitian Supriyadi dari Perguruan Tinggi Raharja, Indonesia pada tahun 2013 yang berjudul "Pemanfaatan Teknologi Bluetooth Untuk Indikator Posisi Suatu Benda". Pada penelitian ini bertujuan memanfaatkan media

- bluetooth untuk membantu mencari posisi suatu benda.
3. Penelitian yang dilakukan oleh Dhida Restu Giri Madya dari Perguruan Tinggi Raharja, Indonesia pada tahun 2014 yang berjudul “Pengendali pintu dan jendela mobil menggunakan smartphone berbasis atmega 328p di kelurahan Cibogo. Penelitian ini membahas tentang komunikasi serial antara arduino dengan smartphone android lewat media Bluetooth.
  4. Penelitian dalam sebuah jurnal yang telah dilakukan oleh Ferry Sudarto, M.Firman dan Sugeng Adi Atma dari Perguruan Tinggi Raharja, Indonesia pada tahun 2013 yang berjudul “Tongkat Ultrasonik untuk Tunanetra sebagai Deteksi Jarak Benda dengan Output Suara”. Diusulkan merancang tongkat ultrasonik untuk tunanetra dengan menggunakan teknologi berbasis mikrokontroler yang dapat mendeteksi keberadaan suatu objek. Untuk bisa mendeteksi jarak benda, tongkat ultrasonik dilengkapi oleh berbagai modul diantaranya adalah sensor Ultrasonik D-Sonar untuk mengukur jarak pengguna dengan benda didepannya, mikrokontroler AT89S51 sebagai memori program, dan ISD 2590 sebagai perekam suara untuk output. Gelombang ultrasonik ini akan dipancarkan dan sinyal yang mengenai suatu objek sebagian akan dipantulkan kembali. Sinyal pantul akan diterima oleh suatu penerima untuk kemudian diolah oleh mikrokontroler. Mikrokontroler akan mengontrol dan mengolahnya, sehingga dapat dihasilkan suatu output berupa suara. Dan sebagai pencatu tegangan untuk semua rangkaian digunakan battery.
  5. Penelitian dalam sebuah jurnal yang telah dilakukan oleh Titik

Muji Rahayu dari Perguruan Tinggi Raharja, Indonesia pada tahun 2010 yang berjudul “Perancangan Dan Pembuatan Penunjuk Arah Serta Deteksi Jarak Benda Untuk Tunanetra Dengan Output Suara Berbasis Mikrokontroler”. Diusulkan untuk merancang dan membuat alat penunjuk arah serta mendeteksi jarak benda untuk penderita tunanetra dengan menggunakan output suara berbasis mikrokontroler. Perancangan alat ini memanfaatkan teori tentang mata angin dan kecepatan gelombang bunyi di udara. Perancangan ini melalui dua tahap, yaitu tahap perancangan hardware dan software. Hardware yang digunakan dalam perancangan alat ini adalah kompas digital HM55B untuk menentukan arah mata angin, sensor Ultrasonik D-Sonar untuk mengukur jarak pengguna dengan benda di depannya, mikrokontroler AT89S51 sebagai memori program, dan ISD 2590 sebagai perekam suara untuk output. Software pada alat ini menggunakan bahasa pemrograman Assembler. Data yang diperoleh dari penelitian ini dianalisis dan dicari simpangannya. Pada perangkat penunjuk arah HM55B diperoleh simpangan rata-rata sebesar 3,65% dengan taraf ketelitian 96,35% dan pada perangkat pendeteksi jarak benda kesalahan relatifnya sebesar 1,92% dengan taraf ketelitian 98,08%.

## **PEMBAHASAN**

### **Flowchart Sistem**



Gambar 1. Flowchart Sistem Keseluruhan

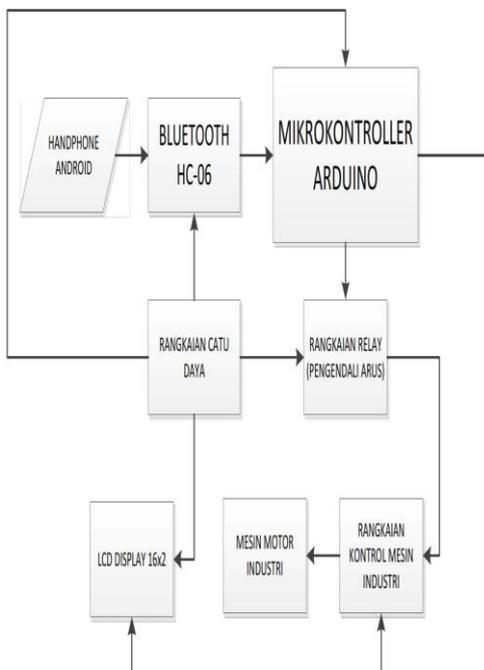
## IMPLEMENTASI

### Perancangan Hardware

Dalam membuat rancangan *hardware* menggunakan beberapa komponen diantaranya, Atmega328 sebagai sistem kontrol, power supply 12 volt dan 5 volt sebagai sumber tegangan, *bluetooth* HC – 06 sebagai media penghubung antara *handphone* dan mikrokontroler, *relay* berfungsi untuk menghidupkan dan mematikan arus tegangan pada rangkaian kontrol mesin industri, *LCD display* 16 x 2 sebagai media penampilan pesan, serta motor *DC* sebagai penggerak prototipe.

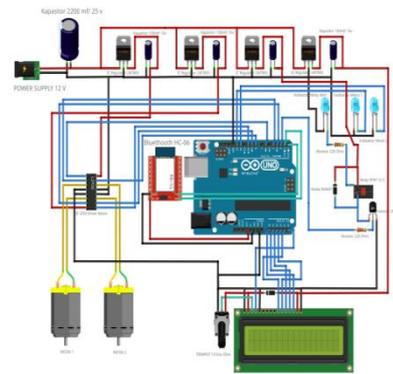
### Rangkaian sistem keseluruhan Perancangan Perangkat Keras (Hardware)

Perancangan perangkat keras, maka di gambarkan alur dan cara kerja perangkat keras pada rangkaian diagram blok pada gambar.2 bawah ini:



Gambar 2. Diagram blok rangkaian

Menurut Sumardi (2013:1), “Mikrokontroler merupakan suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data”.



Gambar 3. Skema Rangkaian Sistem Keseluruhan

### Rangkaian Sistem Minimum Atmega328

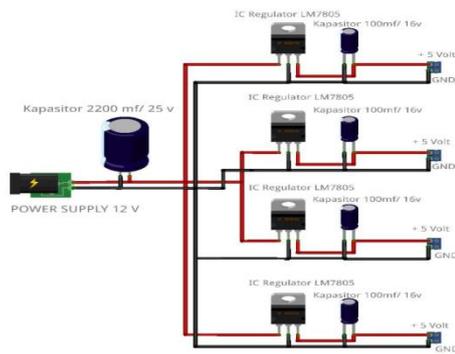
Agar Mikrokontroler Atmega328 dapat digunakan sebagai sistem kontrol, maka perlu dibuat sistem minimumnya. Rangkaian sistem minimum Atmega328 dapat bekerja secara baik dengan memberikan tegangan sebesar 12 volt lalu ditegangan tersebut diturunkan lagi menjadi 5 volt dengan menggunakan IC *Regulator* LM7805

### Rangkaian Catu Daya (Power Supply)

Agar alat yang dibuat dapat bekerja sesuai dengan fungsinya, maka diperlukan sumber tegangan listrik sebagai catu daya. Rangkaian catu daya

yang digunakan mendapatkan sumber tegangan dari adaptor *switching* dengan *output* 12 volt. Tegangan tersebut kemudian diturunkan menjadi 5 volt tegangan DC, melalui IC *regulator* LM7805. Arus yang masuk dari adaptor *switching* akan masuk melalui kapasitor yang bertujuan untuk mengurangi *noise* pada tegangan DC.

Setelah itu keluaran dari kapasitor tersebut masuk ke IC *regulator* yang fungsinya adalah untuk menstabilkan tegangan. IC *regulator* ini terdiri dari dua buah IC, yaitu LM7805 yang menghasilkan tegangan +5 volt. Keluaran dari IC regulator ini kemudian akan masuk kembali ke kapasitor agar tegangan DC yang dikeluarkan dapat lebih halus lagi (*smooth*).



Gambar 4. Rangkaian catu daya

Pada rangkaian catu daya ini menggunakan empat buah sumber output catu daya, yang digunakan terpisah untuk memberikan tegangan kerja pada masing-masing rangkaian. Adapun rangkaian yang menggunakan tegangan sebesar +5 Volt DC adalah, rangkaian kontrol L293, rangkaian motor DC, rangkaian Lcd Display dan rangkaian Bluetooth, sedangkan arus untuk tegangan relay sebesar 12 volt DC tidak perlu diturunkan lagi karena arus yang dimasuk sudah cukup.

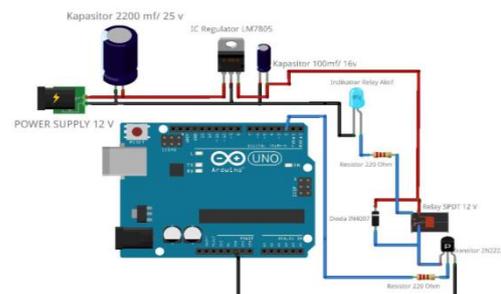
### Modul Bluetooth HC-06

Modul bluetooth HC-06 merupakan salah satu modul bluetooth yang dapat di temukan dipasaran dengan harga yang relative murah. Modul bluetooth HC-06 terdiri dari 6 pin konektor, setiap pin konektor memiliki fungsi-fungsi yang berbeda.

Gambar 5. Bluetooth HC-06

### Rangkaian Relay

Pada dasarnya penggunaan rangkaian *relay* dimaksudkan untuk menghidupkan dan mematikan arus tegangan kerja pada rangkaian kontrol mesin industry, sehingga arus yang mengalir dapat dihidupkan atau dimatikan sesuai dengan kebutuhan. Pada dasarnya cara kerja rangkaian *relay* akan bekerja ketika diberikan perintah dengan huruf “E” pada *handphone*, lalu dikirim ke Mikrokontroler melalui media Bluetooth, dan setelah diterima data yang terkirim diproses oleh Mikrokontroler, sehingga akan memberikan sinyal “HIGH” pada rangkaian *relay*, yang artinya rangkaian *relay* tersebut akan berada pada kondisi aktif dan rangkaian kontrol mesin industry akan mendapatkan arus, sehingga rangkaian kontrol mesin industry dapat bekerja sesuai dengan yang diinginkan, ketika akan mematikan arus pada rangkaian kontrol mesin industry tinggal menekan huruf “F”,



yang artinya mikrokontroler akan memberikan sinyal tidak aktif terhadap rangkaian *relay*, sehingga rangkaian kontrol mesin industri tidak mendapatkan arus lagi, karena rangkaian *relay* berada pada kondisi “LOW”. Gambar rangkaian relay dapat dilihat pada gambar berikut:

Gambar 6. Rangkaian Relay SPDT

Untuk memberikan tegangan kerja pada sebuah *relay* perlu dikonfigurasi terlebih dahulu pada program arduino. Dan untuk mendeklarasikan *relay* pada program arduino dapat dilihat seperti gambar berikut ini:

```

candra
*/
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5, A4, A3, A2, A1, A0);

int relay1 = 2;

int motor1Pin1 = 3;
int motor1Pin2 = 4;
int enablePin = 6;

```

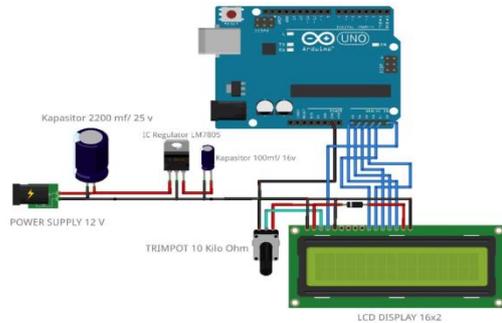
Gambar 7. Deklarasi pin 2 Mikrokontroler - relay

Gambar diatas adalah bagaimana cara mengkonfigurasi *relay* pada program arduino, dalam rangkaian sistem ini relay di pasang pada pin dua mikrokontroler

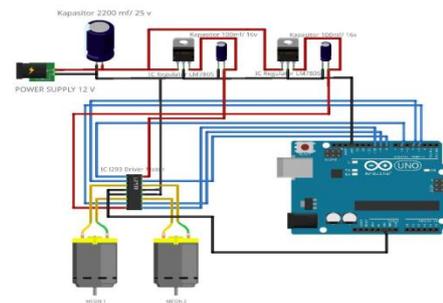
### Rangkaian Lcd Display 16x2

LCD karakter adalah suatu modul yang berfungsi sebagai display yang dapat menampilkan karakter alpha numeric yang memiliki 16 kolom dan 2 baris karakter. LCD ini memiliki warna dasar biru, dan karakter berwarna putih dengan menggunakan backlight. LCD ini berbasis HD44780 dengan supply tegangan sebesar 5V DC. Dengan menggunakan lcd karakter 16x2 display semua intruksi yang dilakukan, akan ditampilkan pada waktu yang telah

ditentukan, baik itu kondisi mesin, atau intruksi-intruksi yang akan dilakukan selanjutnya, gambar dibawah ini menunjukkan bagaimana sebuah lcd karakter 16x2 display dihubungkan dengan mikrokontroler.



Gambar 8. Rangkaian lcd karakter 16x2 display



Ketika lcd karakter 16x2 display sudah dihubungkan dengan sebuah

```

if (state == 'a') {
  digitalWrite(motor1Pin1, HIGH);
  digitalWrite(motor1Pin2, LOW);
  digitalWrite(indmesin1, HIGH);
  if (flag == 0) {

    lcd.display();
    lcd.setCursor (0,0);
    lcd.print(" MESIN 1 SUDAH ");
    lcd.setCursor (0,1);
    lcd.print(" DI HIDUPKAN ");
    delay(5000);
    lcd.clear();

    lcd.display();
    lcd.setCursor (0,0);
    lcd.print(" TEKAN TOMBOL B ");
    lcd.setCursor (0,1);
    lcd.print(" UNTUK MATIKAN ");
    delay(5000);

    lcd.display();
    lcd.setCursor (0,0);
    lcd.print("MASUKAN PERINTAH");
    lcd.setCursor (0,1);
    lcd.print("UNTUK MENGAKSES");
    delay(500);

    flag=1;

```

Mikrokontroler, lcd karakter 16x2 display tidak bisa langsung digunakan

```

else if (state == 'b') {
  digitalWrite(motor1Pin1, LOW);
  digitalWrite(motor1Pin2, LOW);
  digitalWrite(indmesin1, LOW);
  if (flag == 0) {

    lcd.display();
    lcd.setCursor (0,0);
    lcd.print(" MESIN 1 SUDAH ");
    lcd.setCursor (0,1);
    lcd.print(" DI MATIKAN ");
    delay(5000);
    lcd.clear();

    lcd.display();
    lcd.setCursor (0,0);
    lcd.print(" TEKAN TOMBOL A ");
    lcd.setCursor (0,1);
    lcd.print(" UNTUK HIDUPKAN ");
    delay(5000);

    lcd.display();
    lcd.setCursor (0,0);
    lcd.print("MASUKAN PERINTAH");
    lcd.setCursor (0,1);
    lcd.print("UNTUK MENGAKSES");
    delay(500);

    flag = 1;

```

untuk menampilkan pesan, melainkan Mikrokontroler harus diupload ke dalam program sehingga dapat menampilkan karakter.

*Listing* program pada Gambar 9. akan terus dijalankan selama masih terdapat arus listrik, *listing* program diatas akan ditampilkan ketika pada Mikrokontroler mendapat input berupa huruf 'a' dari *handphone* merupakan perintah untuk menghidupkan mesin satu, dan ketika Mikrokontroler mendapat input huruf 'b' dari *handphone* lcd karakter16x2 display akan menampilkan pesan.

Gambar 9. Listing Program Input Huruf "a"

Gambar 9. Listing Program Input Huruf "b"

Perintah pada gambar 9. akan ditampilkan ketika menekan huruf 'b' yang merupakan perintah untuk mematikan mesin satu.

Gambar 10. Rangkain *driver* motor L293

Rangkaian di atas baru akan bekerja ketika mendapat *input*-an dari *handphone*, huruf yang digunakan adalah huruf 'a' untuk menghidupkan mesin satu, huruf 'b' untuk mematikan mesin satu, huruf 'c' untuk menghidupkan mesin dua, huruf 'd' untuk mematikan mesin dua, sedangkan untuk menghidupkan panel listrik menggunakan huruf 'e' dan huruf 'f' untuk mematikan panel listrik. Cara kerja rangkaian di atas adalah dengan

memberikan tegangan 5 volt sebagai *Vcc* pada pin 16 dan 5 volt pada pin delapan untuk tegangan motor, maka IC L293 siap digunakan. Jika terdapat tegangan *input* satu dan *input* dua maka dengan memberikan logika *HIGH* pada *enable1* maka *output* 1 dan *output* 2 akan aktif. Sedangkan *enable1* berlogika rendah, meskipun terdapat tegangan pada *input* 1 dan *input* 2, *output* tetap nol (tidak aktif). Hal ini juga berlaku untuk *input* 3, 4 dan *output* 3, 4 serta *enable* 2. Konfigurasi pin IC L293 di atas, rangkaian di atas dapat digunakan untuk mengontrol dua motor *DC* sekaligus, dan juga dapat digunakan untuk mengontrol motor *DC* secara *continue* dan dengan teknik *PWM* (*Pulse-Width Modulation*). Adapun penulisan *listing* program untuk driver L293 dapat ditulis seperti terlihat pada gambar berikut.

```

candra $
~/
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5, A4, A3, A2, A1, A0);

int relay1 = 2;

int motor1Pin1 = 3;
int motor1Pin2 = 4;
int enable1Pin = 6;

int motor2Pin1 = 8;
int motor2Pin2 = 9;
int enable2Pin = 10;

int indmesin1 = 13;
int indmesin2 = 12;
int state;
int flag=0;
void setup() {
  pinMode(motor1Pin1, OUTPUT);
  pinMode(motor1Pin2, OUTPUT);
  pinMode(enable1Pin, OUTPUT);
  pinMode(motor2Pin1, OUTPUT);
  pinMode(motor2Pin2, OUTPUT);
  pinMode(enable2Pin, OUTPUT);
  digitalWrite(enable1Pin, HIGH);
  digitalWrite(enable2Pin, HIGH);
}

```

Gambar 11.

Program yang digunakan untuk mengontrol driver L329

*Listing* program yang ditunjuk pada garis hitam pertama adalah menghubungkan pin 3,4,6,8,9,10 yang terdapat pada Mikrokontroler dan pin

1,2,7,9,10,15 yang terdapat pada IC driver L293, sedangkan listing program yang kedua adalah bagaimana mendeklarasikan sebuah IC driver L293 sebagai keluaran dari perintah yang terdapat pada Mikrokontroler.

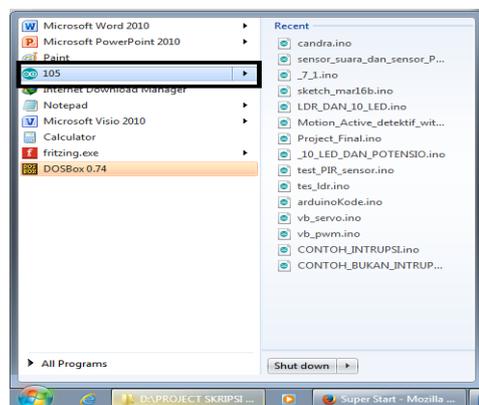
### Perancangan Software

Perancangan perangkat lunak, adalah melakukan penulisan *listing* program ke dalam suatu *Software* Arduino 1.0 dengan menggunakan bahasa pemrograman C, dimana perintah-perintah program tersebut akan di eksekusi oleh *hardware* atau sistem yang di buat.

### Penulisan *listing* program bahasa C

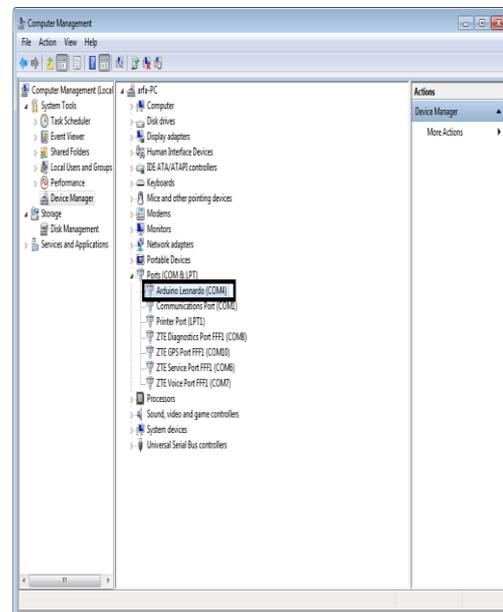
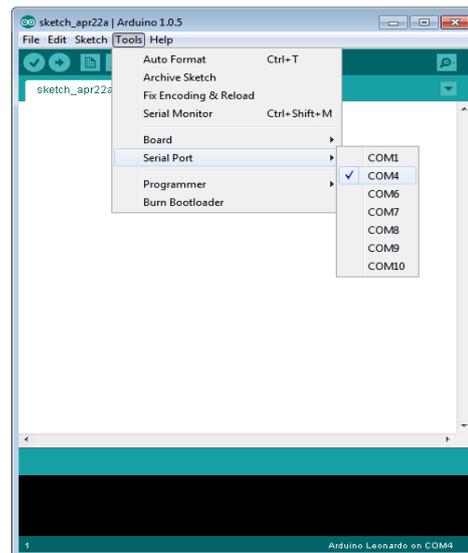
Pada perancangan perangkat lunak akan menggunakan program Arduino 1.0 digunakan untuk menuliskan *listing* program dan menyimpannya dengan file yang berekstensi .pde, dan *bootloader* Arduino Uno sebagai media yang digunakan meng-*upload* program ke dalam mikrokontroler, sehingga mikrokontroler dapat bekerja sesuai dengan yang diperintahkan.

Adapun langkah-langkah untuk memulai menjalankan *software* Arduino 1.0 dapat dilihat seperti pada gambar 12. sebagai berikut :



Gambar 12. Memulai Ide arduino

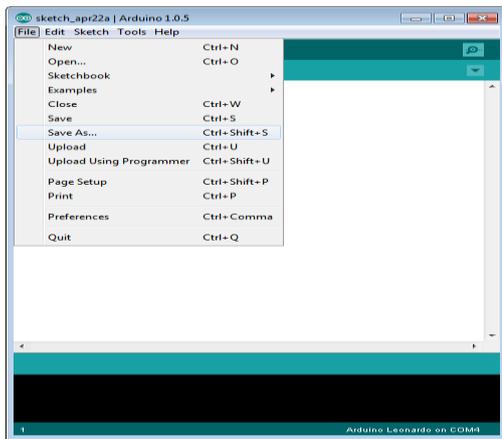
Setelah *form* utama program Arduino 1.0. ditampilkan, maka langkah selanjutnya adalah *mengkonfigurasi* pengalamatan *port* koneksi yang ada pada *device manager*.



Gambar 13. Konfigurasi *port* melalui *device manager*

sebagai berikut :

Setting koneksi *port* pada Arduino 1.0 dilakukan agar pada saat program di *upload* tidak terjadi *error* karena kesalahan pada pengalamatan *port* yang sebelumnya

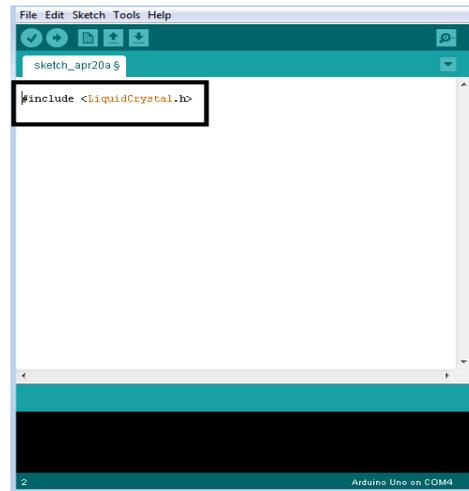
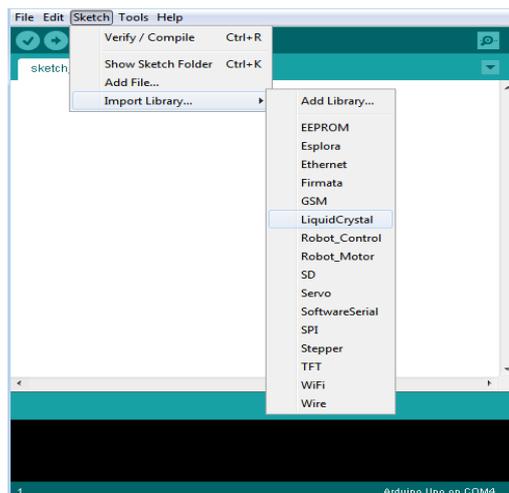


di *setting* juga melalui *device manager*.

Gambar 14. Menyimpan file program pada Arduino 1.0

Menyimpan *listing* program yang sudah dibuat dengan nama berekstensi *.pde* dalam penelitian ini nama *file* yang akan disimpan dengan nama Project\_Final.pde.

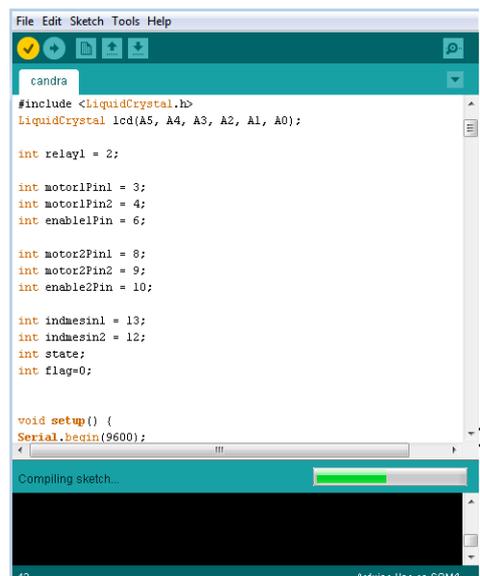
Setelah melakukan penyimpanan *file* program selanjutnya tahap penulisan *listing* dimulai dari mengimpor *library* dan dapat di lihat pada gambar 15



Gambar 15.

Mengimpor *library* pada *header* Arduino 1.0.

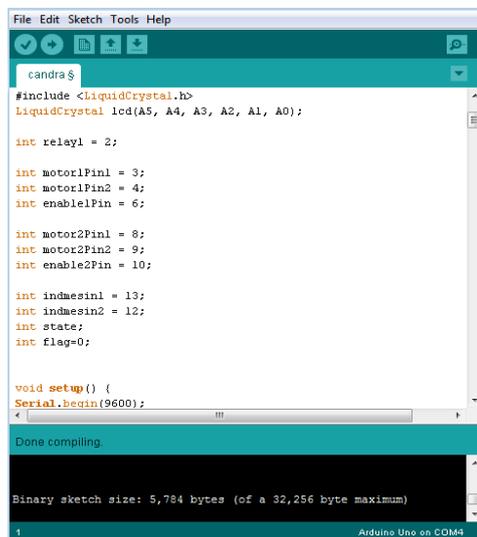
Pemrograman Mikrokontroler ATmega328 yang menggunakan *bootloader* Arduino Uno sebagai media untuk menanamkan program dan Arduino 1.0. Setelah langkah pada gambar di atas dilakukan, agar sistem dapat bekerja sesuai dengan yang diinginkan, selanjutnya lakukan



penulisan *listing* program secara keseluruhan.

Gambar 16. Proses kompilasi *listing* program

Setelah *listing* program ditulis semua, langkah selanjutnya proses kompilasi untuk mengecek apakah *listing* program yang ditulis terjadi kesalahan atau tidak, proses kompilasi dapat dilihat pada gambar 17 diatas.



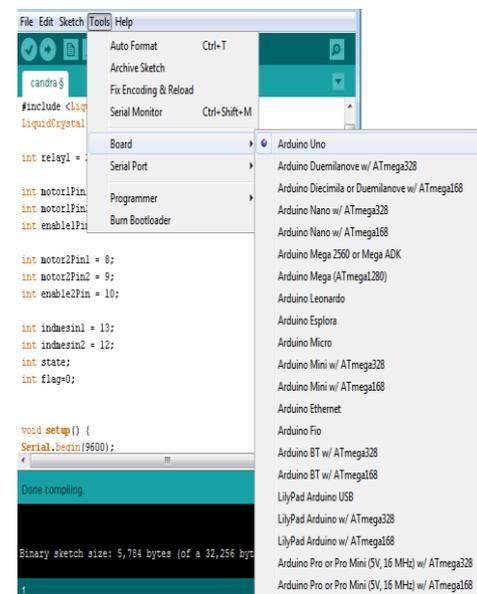
Gambar 17. Hasil kompilasi *listing* program

Pada gambar 17. menunjukan hasil dari kompilasi *listing* program dan hasil dari proses kompilasi tidak terjadi *error*, artinya proses penulisan *listing* program sudah benar, hasil dari kompilasi inilah yang nantinya akan ditanamkan ke dalam sistem Mikrokontroler ATmega328.

### Pengisian program ke dalam IC ATmega328 pada *Bootloader* uno

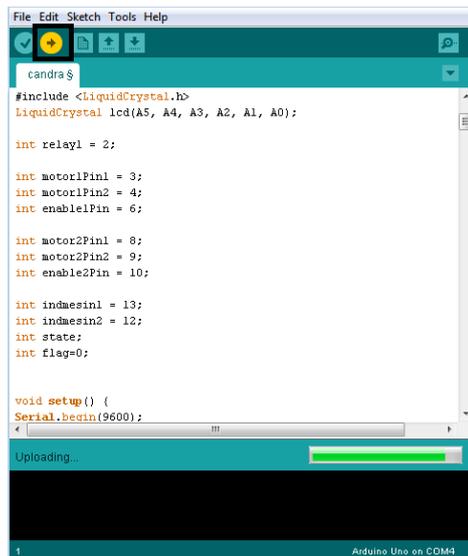
Mikrokontroler bisa bekerja jika di dalamnya sudah dimasukkan *listing* program, program yang akan dimasukan kedalam Mikrokontroler ATmega328 yaitu program aplikasi yang dibuat dengan aplikasi Arduino 1.0. Untuk melakukan pengisian program menggunakan perangkat keras (*Hardware*) dan perangkat lunak (*Software*).

Arduino sebagai media untuk memasukan program ke dalam mikrokontroler ATmega328, maka program yang ditulis pada Arduino 1.0 dapat langsung dimasukan ke dalam Mikrokontroler ATmega328. Langkah selanjutnya sebelum *listing* program dimasukan ke dalam Mikrokontroler, yang perlu diperhatikan yaitu jenis *board* yang akan digunakan pada saat memasukan *listing* program, proses pemilihan *board* yang digunakan untuk memasukan *listing* program dapat dilihat pada gambar 18. sebagai berikut:



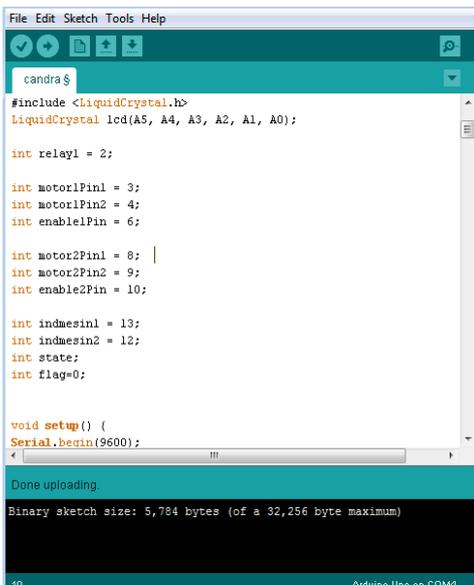
Gambar 18. Pemilihan Arduino board

Setelah jenis board sudah dipilih, langkah selanjutnya adalah memasukan program ke dalam Mikrokontroler dengan menggunakan *internal clock*, arti dari *internal clock* adalah dengan memanfaatkan board Arduino sebagai board untuk berkomunikasi dengan komputer, dan Mikrokontroler yang ada pada arduino board tersebut dilepas, agar IC ATmega328 yang akan digunakan dapat terbaca oleh Arduino



board.

Pada tampilan pemrograman Arduino 18. diatas, dilakukan dengan mengklik tombol *upload* yang ada pada Arduino 1.0., pada saat mengupload



*listing* program secara otomatis akan menampilkan pesan bahwa proses *upload* program tidak terjadi *error* atau sukses. Proses *upload listing* program yang tidak terjadi *error* .

Gambar 19. Proses *upload listing* program sukses

Setelah langkah *upload listing* program selesai, maka sistem Mikrokontroler ATmega328 yang berjudul “**Security sistem Kelistrikan Motor Industri Menggunakan Media Bluetooth Berbasis Mikrokontroler ATmega328**” sudah siap digunakan.

```

File Edit Sketch Tools Help
candra $
***KONSENTRASI = CCIT
*/
#include <LiquidCrystal.h>
LiquidCrystal lcd(A5, A4, A3, A2, A1, A0);

int relay1 = 2;
int motor1Pin1 = 3;
int motor1Pin2 = 4;
int enable1Pin = 6;

int motor2Pin1 = 8;
int motor2Pin2 = 9;
int enable2Pin = 10;
int indmesin1 = 13;
int indmesin2 = 12;
int state;
int flag=0;

void setup() {
  Serial.begin(9600);
  pinMode(motor1Pin1, OUTPUT);
  pinMode(motor1Pin2, OUTPUT);
  pinMode(enable1Pin, OUTPUT);
  pinMode(motor2Pin1, OUTPUT);
  pinMode(motor2Pin2, OUTPUT);
  pinMode(enable2Pin, OUTPUT);
  digitalWrite(enable1Pin, HIGH);
  digitalWrite(enable2Pin, HIGH);

  pinMode(relay1, OUTPUT);
  pinMode(indmesin1, OUTPUT);
  pinMode(indmesin2, OUTPUT);
}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
20 Arduino Uno on COM4

```

```

File Edit Sketch Tools Help
candra $
delay(5000);

lcd.display();
lcd.setCursor(0,0);
  lcd.print("MASUKAN PERINTAH");
  lcd.setCursor(0,1);
  lcd.print("UNTUK MENGAKSES");
  delay(500);
flag=1;
}
}

else if (state == 'b') {
  digitalWrite(motor1Pin1, LOW);
  digitalWrite(motor1Pin2, LOW);
  digitalWrite(indmesin1, LOW);
  if (flag == 0) {

    lcd.display();
    lcd.setCursor(0,0);
    lcd.print(" MESIN 1 SUDAH ");
    lcd.setCursor(0,1);
    lcd.print(" DI MATIKAN ");
    delay(5000);
    lcd.clear();

    lcd.display();
    lcd.setCursor(0,0);
    lcd.print(" TEKAN TOMBOL A");
    lcd.setCursor(0,1);
    lcd.print(" UNTUK HIDUPKAN ");
  }
}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
68 Arduino Uno on COM4

```

```

File Edit Sketch Tools Help
candra $

lcd.begin(16, 2);
lcd.setCursor(0,0 );
lcd.print("NAMA :CANDRA JATUSMAN STMIK Raharja");
lcd.setCursor(0, 1);
lcd.print("NIM :1033464312 Tahun 2013 / 2014");
delay(500);
for (int positionCounter = 0; positionCounter < 40; positionCounter++) {
  lcd.scrollDisplayLeft();
  delay(400);
}
}

void loop() {
if(Serial.available() > 0){
state = Serial.read();
flag=0;
}

if (state == 'a') {
digitalWrite(motor1Pin1, HIGH);
digitalWrite(motor1Pin2, LOW);
digitalWrite(indmesin1, HIGH);
if(flag == 0){
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" MESIN 1 SUDAH ");
  lcd.setCursor(0,1);
  lcd.print(" DI HIDUPKAN ");
  delay(5000);
  lcd.clear();
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" TEKAN TOMBOL B ");
  lcd.setCursor(0,1);
  lcd.print(" UNTUK MATIKAN ");
}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
68 Arduino Uno on COM4

```

```

File Edit Sketch Tools Help
candra $
delay(5000);

lcd.display();
lcd.setCursor(0,0);
  lcd.print("MASUKAN PERINTAH");
  lcd.setCursor(0,1);
  lcd.print("UNTUK MENGAKSES");
  delay(500);

  flag = 1;
}
}

else if (state == 'c') {
digitalWrite(motor2Pin1, HIGH);
digitalWrite(motor2Pin2, LOW);
digitalWrite(indmesin2, HIGH);
if(flag == 0){

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" MESIN 2 SUDAH ");
  lcd.setCursor(0,1);
  lcd.print(" DI HIDUPKAN ");
  delay(5000);
  lcd.clear();

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" TEKAN TOMBOL D");
  lcd.setCursor(0,1);
  lcd.print(" UNTUK MATIKAN ");
}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
68 Arduino Uno on COM4

```

```

File Edit Sketch Tools Help
candra $
lcd.display();
lcd.setCursor(0,0);
  lcd.print("MASUKAN PERINTAH");
lcd.setCursor(0,1);
  lcd.print("UNTUK MENGAKSES");
delay(500);
flag=1;
}
}
else if (state == 'e') {
digitalWrite(relay1, HIGH);
if(flag == 0){

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" P A N E L  M E S I N  ");
  lcd.setCursor(0,1);
  lcd.print(" S U D A H  H I D U P  ");
  delay(5000);
  lcd.clear();

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" T E K A N  T O M B O L  F  ");
  lcd.setCursor(0,1);
  lcd.print(" U N T U K  M A T I K A N  ");
  delay(5000);

  lcd.display();
}

}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
68 Arduino Uno on COM4

```

```

File Edit Sketch Tools Help
candra $
  lcd.setCursor(0,0);
  lcd.print("MASUKAN PERINTAH");
  lcd.setCursor(0,1);
  lcd.print("UNTUK MENGAKSES");
  delay(500);

flag=1;
}
}
else if (state == 'f') {
digitalWrite(relay1, LOW);
digitalWrite(motor1Pin1, LOW);
digitalWrite(motor1Pin2, LOW);
digitalWrite(motor2Pin1, LOW);
digitalWrite(motor2Pin2, LOW);
digitalWrite(indmesin1, LOW);
digitalWrite(indmesin2, LOW);
if(flag == 0){

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" M E S I N  D A L A M  ");
  lcd.setCursor(0,1);
  lcd.print(" K O N D I S I  O F F  ");
  delay(5000);
  lcd.clear();

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print("MASUKAN PERINTAH");
  lcd.setCursor(0,1);
  lcd.print("UNTUK MENGAKSES");
  delay(500);

}

}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
218 Arduino Uno on COM4

```

```

File Edit Sketch Tools Help
candra $
flag=1;
}
}
else{
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" T E K A N  T O M B O L  E  ");
  lcd.setCursor(0,1);
  lcd.print(" U N T U K  H I D U P K A N  ");
  delay(5000);
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" P A N E L  ");
  lcd.setCursor(0,1);
  lcd.print(" ");
  delay(3000);

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" T E K A N  T O M B O L  A  ");
  lcd.setCursor(0,1);
  lcd.print(" U N T U K  H I D U P K A N  ");
  delay(5000);
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" M E S I N  1  ");
  lcd.setCursor(0,1);
  lcd.print(" ");
  delay(3000);

  lcd.display();
}

}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
219 Arduino Uno on COM4

```

```

File Edit Sketch Tools Help
candra $
  lcd.print(" T E K A N  T O M B O L  A  ");
  lcd.setCursor(0,1);
  lcd.print(" U N T U K  H I D U P K A N  ");
  delay(5000);
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" M E S I N  1  ");
  lcd.setCursor(0,1);
  lcd.print(" ");
  delay(3000);

  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" T E K A N  T O M B O L  C  ");
  lcd.setCursor(0,1);
  lcd.print(" U N T U K  H I D U P K A N  ");
  delay(5000);
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print(" M E S I N  2  ");
  lcd.setCursor(0,1);
  lcd.print(" ");
  delay(3000);

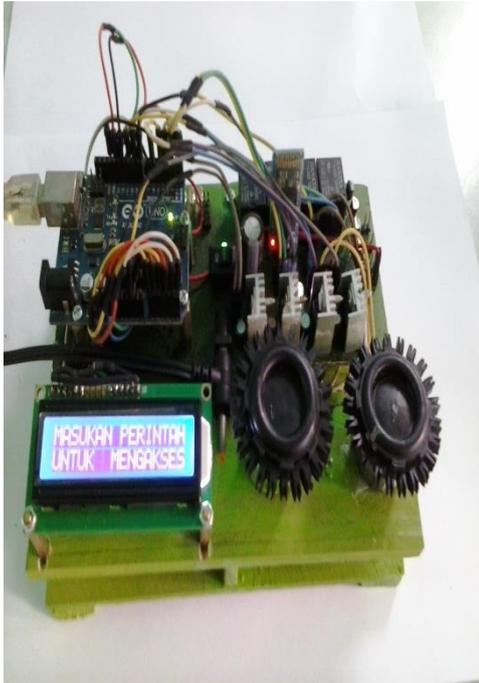
  lcd.display();
  lcd.setCursor(0,0);
  lcd.print("MASUKAN PERINTAH");
  lcd.setCursor(0,1);
  lcd.print("UNTUK MENGAKSES");
  delay(5000);
}

}

Done Saving.
Binary sketch size: 5,784 bytes (of a 32,256 byte maximum)
219 Arduino Uno on COM4

```

Gambar 20. Listing Program Keseluruhan



Gambar 21 . Bentuk dari *Prototype* Alat

### Gambar *Prototype* Alat

Bentuk dari alat dibawah ini adalah *prototype* dari sebuah sistem keamanan kelistrikan motor industri dengan menggunakan hak akses media *Bluetooth* pada sebuah alat komunikasi handphone berbasis Mikrokontroler ATmega 328 yang sedang dioperasikan atau disimulasikan pada PT. Utama Raya Motor Industri .

### Kesimpulan

Berdasarkan hasil analisis dan perumusan masalah yang telah dilakukan, maka penulis dapat menyimpulkan:

- Pada konsep kerja dari aplikasi ini adalah dengan memanfaatkan input dari *handphone* Android yang dikirimkan melalui media

*Bluetooth* dengan menggunakan aplikasi *Blueterm* ke dalam Mikrokontroler ATmega 328 yang selanjutnya diproses untuk menghidupkan panel pada mesin / motor.

- Komunikasi media *Bluetooth* terhadap Mikrokontroler adalah dengan menggunakan modul *Bluetooth* HC-06 yang kemudian dihubungkan melalui pin RX/TX ( Mengirim / Menerima Data secara Serial ) , sehingga *handphone* dapat terkoneksi atau berkomunikasi secara baik pada Mikrokontroler . Dalam aplikasi ini pengontrolan dilakukan dengan menggunakan tombol *input ON/OFF* dimana pengontrolan tersebut akan di eksekusi `juga dengan menggunakan aplikasi media jaringan *Bluetooth* tersebut.

### Daftar Pustaka

Sudarma, D. 2013. Rancang Bangun Kendali Lampu on/off Dengan Smartphone Android Via Bluetooth. Jurnal Pontianak Universitas Tanjungpura.

<http://arduino.cc/en/Main/arduinoBoardMicro>, diakses pada tanggal 11 Desember 2015.

<http://www.adafruit.com/datasheets/UG-2864HSWEG01.pdf>, pada tanggal 11 Desember 2015.

<http://www.adafruit.com/datasheets/SSD1306.pdf>, pada tanggal 11 Desember 2015.

<http://www.alldatasheet.com/datasheet-pdf/pdf/3042/MOTOROLA/HC05.html>, pada tanggal 11 Desember 2015.

<http://basic4ppc.com/android/why.html>, pada tanggal 11 Desember 2015.

<http://www.famosastudio.com/serial-port-bluetooth-module-slave-hc-06.html>,  
pada tanggal 9 Februari 2015.