

BAB II KAJIAN PUSTAKA

2.1 Teori Dasar

Seperangkat konsep, definisi dan proposisi yang disusun secara sistematis adalah sebuah alur logika dalam sebuah teori. Menjelaskan (*explanation*), meramalkan (*prediction*), dan pengendalian (*control*), suatu gejala adalah tiga kegunaan teori (Sugiyono, 2014:54). Menurut (Sugiyono, 2014:58) suatu penelitian merupakan uraian sistematis tentang teori (dan bukan sekedar pendapat pakar atau penulis buku) dan hasil-hasil penelitian yang relevan dengan variabel yang diteliti.

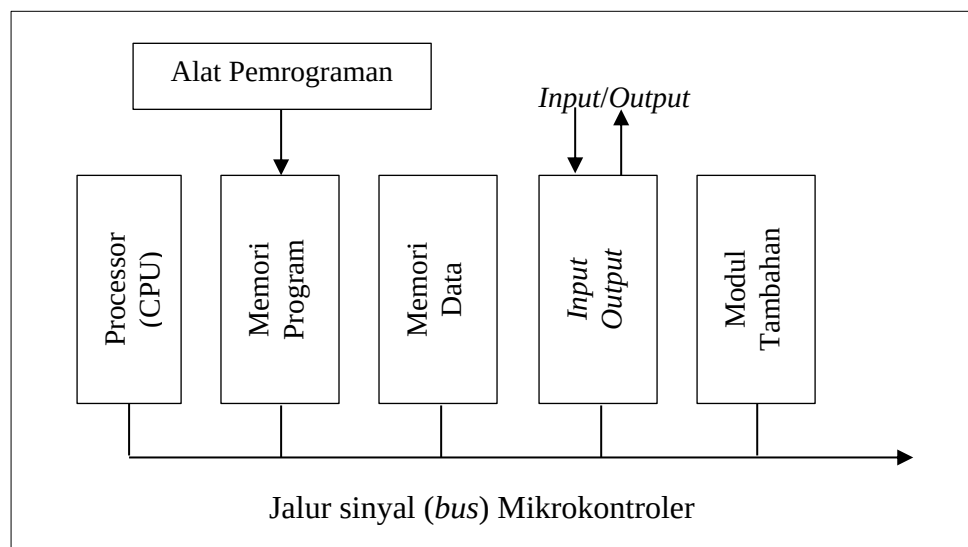
2.1.1 Mikrokontroler Arduino

Processor, *memory*, dan antarmuka yang bisa diprogram adalah bagian-bagian dalam Mikrokontroler. Ukuran Mikrokontroler lebih kecil dibandingkan computer pribadi dan komputer mainframe meskipun dibangun dengan elemen-elemen yang sama (Syahwil, 2014:53). Cara kerja Mikrokontroler adalah membaca dan menulis data yang dimasukan melalui sebuah alat elektronik digital yang dapat menghasilkan keluaran dan kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus (Syahwil, 2014:54).

Mikrokontroler pertama kali dikenalkan di Texas Instrument dengan seri TMS 1000 pada tahun 1970 yang merupakan mikrokontroler bit pertama. Mikrokontroler ini mulai dibuat sejak 1971. Pada tahun 1976 Intel mengeluarkan mikrokontroler

yang terkenal dengan nama 8048 yang merupakan mikrokontroler 8 bit yang termasuk keluarga MCS 48(Syahwil, 2014:57).

Arduino Uno adalah papan Mikrokontroler berbasis ATmega32 yang memiliki 14 pin digital *input/output* (dimana 6 pin dapat digunakan sebagai *output* PWM), 6 *input* analog, *clock speed* 16 MHz, koneksi USB, jack listrik, *header* ICSP, dan tombol reset (Syahwil, 2014:64). Menurut (Syahwil, 2014:58–59) jenis-jenis Mikrokontroler yang banyak digunakan secara umum diantaranya keluarga MCS51, AVR, PIC, Arduino dan ARM Cortex-M0. Menurut Dian Artanto (2009:10-11) sebuah Mikrokontroler umumnya memiliki komponen dasar sebagai berikut:



Gambar 2.1 Bagian Mikrokontroler
(Sumber: Dian Artanto, 2009:10)

Keterangan masing-masing bagian sebagai berikut:

1. Prosesor/CPU

Prosesor melakukan fungsi logika dan aritmetika mengikuti intruksi yang dibaca dari memori program. Prosesor ini juga akan membaca dan menuliskan data ke memori data dan ke modul *input/output*.

2. Memori Program

Memori program menyimpan intruksi untuk dibaca oleh prosesor. Prosesor hanya dapat membaca, tetapi tidak bisa menuliskan datanya ke memori program ini. Hanya alat pemrograman yang dapat menuliskan datanya ke memori. Data dalam memori ini tetap tersimpan sekalipun listrik mati.

3. Memori Data

Memori data menyimpan data dan variabel yang digunakan oleh prosesor. Prosesor dapat membaca dan menuliskan datanya ke memori data ini. Data dalam memori ini akan hilang bila tidak mendapat daya listrik.

4. Alat Pemrograman

Alat yang digunakan untuk memasukkan instruksi atau program ke dalam memori program Mikrokontroler.

5. *Input/Output*

Input/output bekerja untuk menghubungkan Mikrokontroler dengan piranti luar yang ditempatkan pada kaki-kaki Mikrokontroler.

6. Modul Tambahan

Berbagai fungsi tambahan disediakan oleh Mikrokontroler, seperti *counter/timer*, ADC, *comparator*, PWM, 12C, SPI dan lain-lain.

Menurut (Syahwil, 2014:57–58) pembagian mikrokontroler yang didasarkan pada kompleksitas intruksi-intruksi yang dapat diterapkan ada 2 yaitu, RISC dan CISC. RISC (*Reduced Instruction Set Computer*) yaitu instruksi yang terbatas dan fasilitas yang lebih banyak. CISC (*Complex Instruction Set Computer*) yaitu instruksi yang lebih lengkap dan fasilitas yang secukupnya.

Menurut (Syahwil, 2014:60) pembuatan Arduino dimulai pada tahun 2005 disitus perusahaan komputer Olivetti di Italia guna membuat perangkat untuk mengendalikan proyek desain interaksi siswa yang lebih murah. Dilanjut pada bulan Mei 2011, dan 300.000 Arduino terjual. Modul Arduino diciptakan oleh Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, David A. Mellis, dan Nicholas Zambetti.

Seluruh mikrokontroler yang diimplementasikan pada produk Arduino menggunakan ATmega keluarga AVR, dimana salah satunya seri ATmega 328 dengan jumlah fitur diantaranya *On-Chip System Debug*, 5 ragam tidur (*Mode Sleep*), 6 saluran ADC yang mendukung reduksi derau, ragam daya (*Power-save Mode*, *Power-Down*), dan Ragam siaga (*Standby Mode*) (Istiyanto, 2014:6). Menurut (Istiyanto, 2014:6) arsitektur mikrokontroler ATmega 328 yang sering digunakan pada *board* Arduino dan ATmega 120 menggunakan kristal 16 MHz sebagai pembangkit *clock*, memiliki blok memori flash untuk penyimpanan

instruksi program, SRAM untuk penyimpanan variabel data sementara dan EEPROM sebagai media penyimpanan data yang tetap tersimpan meskipun mikrokontroler dalam keadaan tidak dicatu.

Menurut (Dinata, 2015:1) karya yang canggih dapat dihasilkan dengan menggunakan Arduino tanpa harus mengetahui bahasa pemrograman, karena Arduino adalah mikrokontroler yang dirancang untuk bias digunakan dengan mudah oleh para seniman dan desainer. Arduino sangatlah membantu dalam sebuah prototyping atau membuat sebuah proyek karena Arduino memberikan I/O yang fix sehingga proses pembuatan lebih efisien, serta Arduino dapat digabungkan dengan modul elektro dan pada desiner hanya perlu merancang *software* yang dapat mendayagunakan rancangan *hardware* yang ada (Dinata, 2015:3).

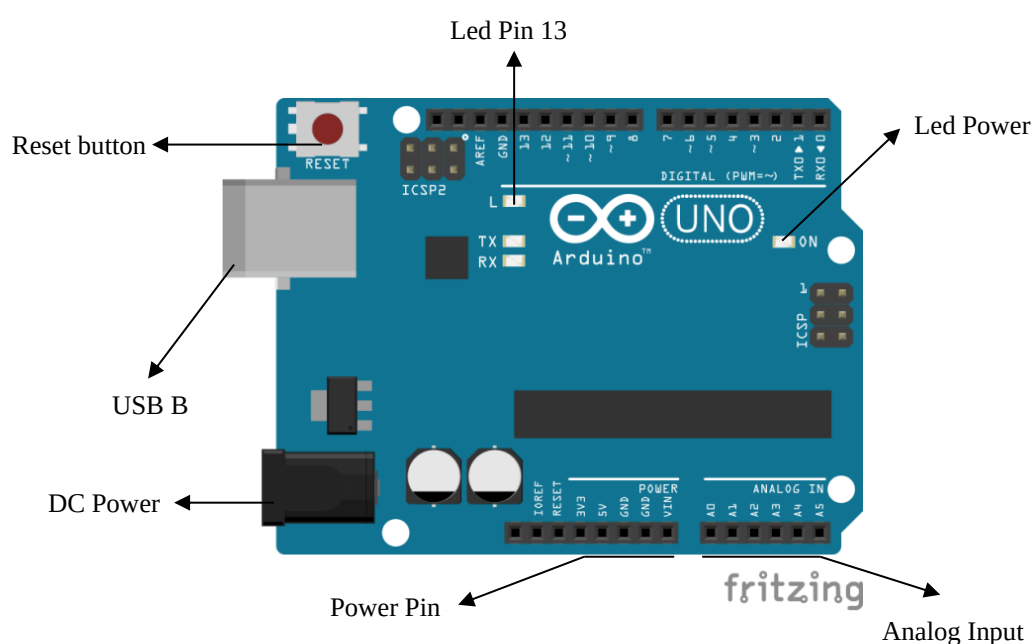
Arduino digunakan oleh peneliti karena berbagai kelebihan yang dimilikinya, diantaranya seperti yang dikemukakan oleh (Syahwil, 2014:61) yaitu:

1. Papan Arduino biasanya dijual relatif murah dibanding platform mikrokontroler pro yang lainnya.
2. Sederhana dan mudah pemrogramannya bagi seorang pemula dan cukup fleksibel bagi tingkat lanjut dalam menggunakannya.
3. Perangkat lunak Arduino IDE serta perangkat kerasnya dipublikasikan secara *open source*.
4. Tidak perlu perangkat *chip programmer* karena di dalamnya terdapat *bootloader* yang akan menangani *upload* program.

5. Sudah memiliki sarana komunikasi USB.
6. Bahasa pemrograman relatif mudah, karena *software* Arduino dilengkapi dengan kumpulan *library* yang cukup lengkap.
7. Memiliki modul siap pakai (*shield*) yang bisa ditancapkan pada *board* Arduino.

Secara umum menurut (Dinata, 2015:3) Arduino terdiri atas dua bagian utama, yaitu:

1. Bagian H//D berupa papan yang berisi I/O.
2. Bagian *software* yang meliputi *Integrated Development Enviroment* (IDE) untuk menulis program, serta Arduino yang memerlukan instalasi *driver* untuk menghubungkan dengan komputer, dimana di dalam IDE tersebut telah terdapat *library* untuk pengembangan program.



Gambar 2.2 Board Arduino Uno
(Sumber: Data Penelitian 2020)

Berikut ini adalah spesifikasi *Arduino Board* Uno menurut (Syahwil, 2014:64) yaitu:

Tabel. 2.1 Spesifikasi *Board* Arduino Uno

Mikrokontroler	ATMega 32
Tegangan Operasi	5V
Tegangan <i>Input</i> (Disarankan)	7-12V
Batas Tegangan <i>Input</i>	6-20-V
Pin Digital I/O	14 (dimana 6 pin <i>output</i> PWM)
Pin Analog <i>Input</i>	6
Arus DC per I/O Pin	40 mA
Arus DC untuk pin 3.3V	50 mA
Flash Memori	32 KB (ATMega 328, dimana 0.5 kb digunakan oleh <i>bootloader</i>)
SRAM	2 KB (ATMega 328)
EEPROM	1 KB (ATMega 328)
<i>Clock</i>	16 MHz

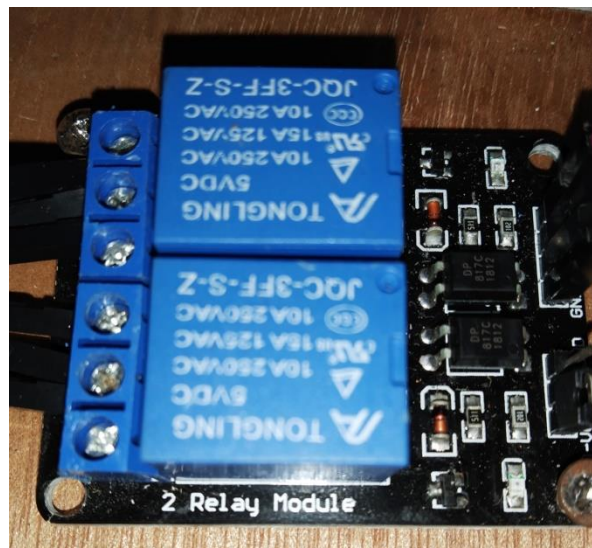
(Sumber: Syahwil, 2013)

Bahasa pemrograman yang digunakan dalam Arduino adalah bahasa C. Bahasa C telah membuat bermacam-macam sistem operasi dan compiler untuk banyak bahasa pemrograman, misalnya sistem operasi Unix, Linux dan lain-lain. Bahasa C menghasilkan file kode objek yang sangat kecil dan dieksekusi dengan sangat cepat, oleh karena itu Bahasa C sering digunakan pada sistem operasi dan pemrograman mikrokontroler (Syahwil, 2014:80).

Menurut (Istiyanto, 2014:52) bahasa Arduino memiliki struktur serupa dengan bahasa *Processing* dan juga kode sketch yang lebih sederhana dari standar bahasa C. Pada dasarnya dalam API (*Application Protocol Interface*) bahasa Arduino terdiri dari tiga bagian, yaitu variabel, *directive setup()*, dan *directive loop()*. API Arduino tidak membutuhkan deklarasi *directivemain()* seperti pada bahasa C tetapi hanya diperlukan blok fungsi *setup()* dan *loop()*.

2.1.2 Relay

Relay adalah saklar (*switch*) yang dioperasikan secara listrik dan merupakan komponen elektromekanikal yang terdiri dari dua bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar/*switch*). Relay menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dan dapat menghantarkan listrik yang bertegangan lebih tinggi (Muhamad Saleh, 2017:87).



Gambar 2.3 Relay
(Sumber: Data Penelitian 2020)

Relay adalah komponen elektronika berupa saklar elektronik yang digerakkan oleh arus listrik. Secara prinsip, relay merupakan tuas saklar dengan lilitan kawat pada batang besi (solenoid) didekatnya. Ketika solenoid dialiri arus listrik, tuas akan tertarik karena adanya gaya magnet yang terjadi pada solenoid sehingga kontak saklar akan tertutup (Turang, 2015:75).

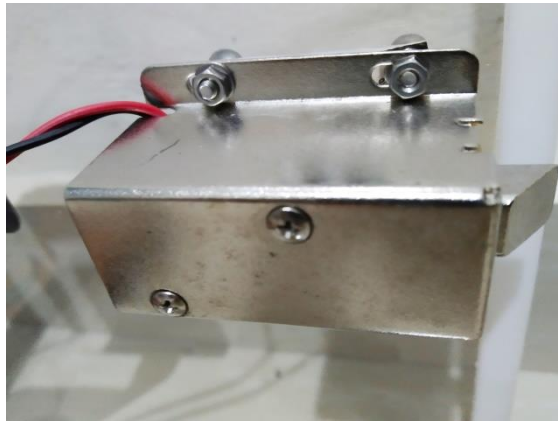
Menurut (Muhamad Saleh, 2017:88) relay termasuk ke dalam jenis saklar, oleh karena itu istilah *Pole* dan *Trow* yang dipakai oleh saklar juga berlaku untuk relay. *Pole* merupakan banyaknya kontak (*contact*) yang dimiliki oleh sebuah relay, sedangkan *Trow* ialah banyaknya kondisi yang dimiliki oleh sebuah kontak (*contact*). Menurut (Muhamad Saleh, 2017:88) penggolongan relay berdasarkan jumlah *pole* dan *trow*-nya adalah sebagai berikut:

1. *Single Pole Single Throw* (SPST) yaitu relay golongan ini memiliki empat terminal, dua terminal untuk saklar dan dua terminalnya untuk *Coil*.
2. *Double Pole Single Throw* (DPST) yaitu relay golongan ini memiliki enam terminal, diantaranya empat terminal yang terdiri dari dua pasang terminal saklar sedangkan dua terminal lainnya untuk *Coil*. Relay DPST dapat dijadikan dua saklar yang dikendalikan oleh satu *Coil*.
3. *Single Pole Double Throw* (SPDT) yaitu relay golongan ini memiliki lima terminal, tiga terminal untuk saklar dan dua terminalnya lagi untuk *Coil*.
4. *Double Pole Double Throw* (DPDT) yaitu relay golongan ini memiliki terminal sebanyak delapan terminal, diantaranya enam terminal yang merupakan dua pasang relay SPDT yang dikendalikan oleh satu (*single*) *Coil*. Sedangkan dua lainnya untuk *Coil*.

2.1.3 Solenoid Door Lock

Solenoid *Door Lock* adalah sebuah pengunci pintu yang mengaplikasikan sistem solenoid. Solenoid adalah sebuah kumparan eletromagnet yang dirancang

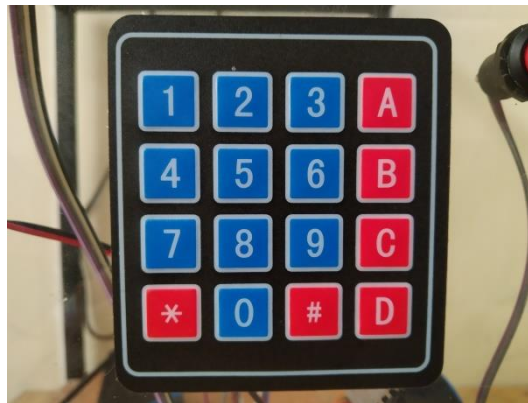
secara khusus. Cara kerja solenoid ini adalah pada saat arus listrik (Shandy. Dkk, 2015:3).



Gambar 2.4 Solenoid *Door Lock*
(Sumber: Data Penelitian 2020)

2.1.4 Keypad 4x4

Pada tugas akhir ini, peneliti menggunakan *keypad* 4x4 dalam memberikan masukan pada suatu perangkat elektronik yang masih membutuhkan interaksi manusia. *Keypad* ini juga berfungsi sebagai interface antara mesin dan manusia. *Keypad matriks* ini adalah tombol-tombol yang disusun secara matriks yang terdiri dari 8 pin untuk 16 tombol yang tersusun secara horizontal membentuk baris dan secara vertikal membentuk kolom.



Gambar 2.5 Keypad 4x4
(Sumber: Data Penelitian 2020)

2.1.5 *Push Button*

Menurut (Syahwil, 2014:37) pada umumnya saklar *push button* adalah tipe saklar yang hanya kontak sesaat saja saat ditekan dan setelah dilepas maka akan kembali lagi menjadi *Normaly Open* (NO), biasanya saklar tipe NO ini memiliki rangkaian penguncinya yang dihubungkan dengan konektor dan tipe NO digunakan untuk tombol On. *Push Button* ada juga yang bertipe *Normaly Close* (NC), biasanya digunakan untuk tombol off.



Gambar 2.6 Push Button
(Sumber: Data Penelitian 2020)

2.1.6 *Display LCD 16x2*

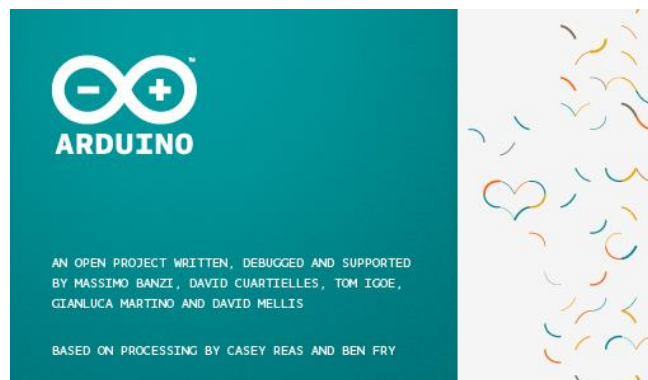
Menurut (Sinaulan, 2015:63) LCD atau *Liquid Cristal Display* adalah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf maupun grafik. Di pasaran tampilan LCD (lihat gambar 2.7) sudah tersedia dalam bentuk modul yaitu tampilan LCD beserta rangkaian pendukungnya termasuk ROM dan sebagainya. LCD mempunyai pin data, kontrol catu daya dan pengatur kontras tampilan.



Gambar 2.7 LCD
(Sumber: Data Penelitian 2020)

2.2 *Tools/Software/Aplikai/System*

2.2.1 *Software Arduino IDE*

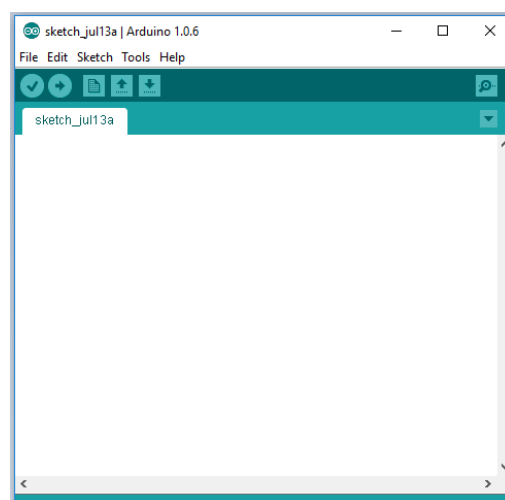


Gambar 2.8 Software Arduino
(Sumber: Data Penelitian 2020)

Dalam tugas akhir ini peneliti menggunakan *software* Arduino IDE dalam menjalankan program yang diperlukan. Menurut (Syahwil, 2014:39) untuk menulis program pada *board* Arduino dibutuhkan *software* Arduino IDE (*Integrated Development Enviroment*). IDE adalah sebuah *software* untuk menulis program, mengkompilasi menjadi kode biner dan meng-*upload* ke dalam *memory* mikrokontroler.

Menurut (Istiyanto, 2014:46) IDE Arduino merupakan aplikasi yang mencakup *editor*, *compiler* dan *uploader* dapat menggunakan semua seri modul keluarga Arduino, seperti Arduino Duemilanove, Uno, Bluetooth, Mega. Kecuali ada beberapa tipe *board* produksi Arduino yang memakai mikrokontroler di luar seri AVR, seperti mikroprosesor ARM.

Dari gambar 2.9 terlihat *button* (tombol) yang ada di IDE Arduino, menurut (Syahwil, 2014:41) *Software* IDE Arduino adalah *software* yang ditulis dengan menggunakan Java. Jendela utama IDE Arduino terdiri dari tiga bagian utama, yaitu:



Gambar 2.9 Jendela Utama IDE Arduino 1.0.6
(Sumber: Data Penelitian 2020)

1. Bagian atas, yakni *Toolbar*. Pada bagian atas juga terdapat menu *file*, *edit*, *sketch*, *tools* dan *help*.
2. Bagian tengah, yaitu tempat penulisan kode program atau sketch.
3. Bagian bawah berupa jendela pesan (*message windows*) atau tes konsul yang berisi status dan pesan *error*.

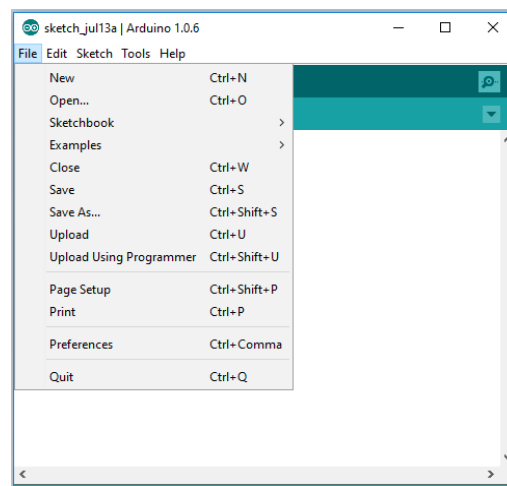


Gambar 2.10 Bagian *Toolbar* IDE Arduino

(Sumber: Data Penelitian 2020)

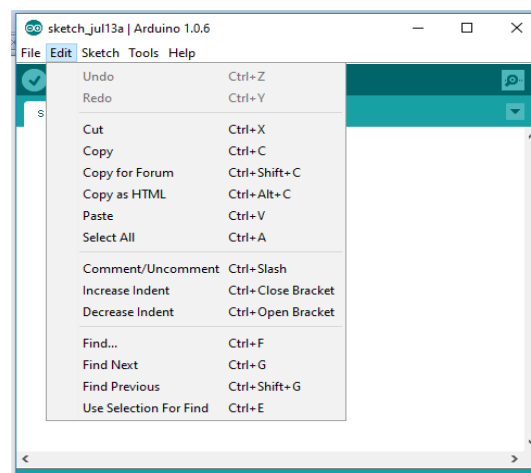
Berdasarkan gambar di atas, fungsi pada bagian *toolbar* tersebut adalah:

1. *Verify* : Mengirim ke *board* Arduino setelah diperiksa kesalahannya.
2. *Uploader* : Mengirim sketch pada *board* Arduino.
3. *New* : Membuat sebuah sketch baru.
4. *Open* : Membuat daftar sketch pada *sketchbook* untuk dibuka.
5. *Save* : Menyimpan kode atau sketch pada *sketchbook*.
6. *Serial Monitor* : Menampilkan data serial yang dikirimkan dari *board* Arduino.



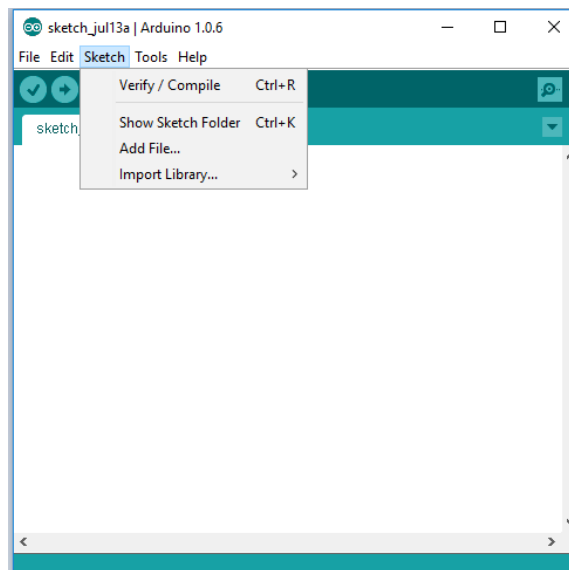
Gambar 2.11 Bagian Menu File IDE Arduino
(Sumber: Data Penelitian 2020)

Pada gambar 2.11 terdapat menu *File* yang terdiri dari *new*, *open*, *Sketchbook*, *example*, *save*, *save as* dan seterusnya.



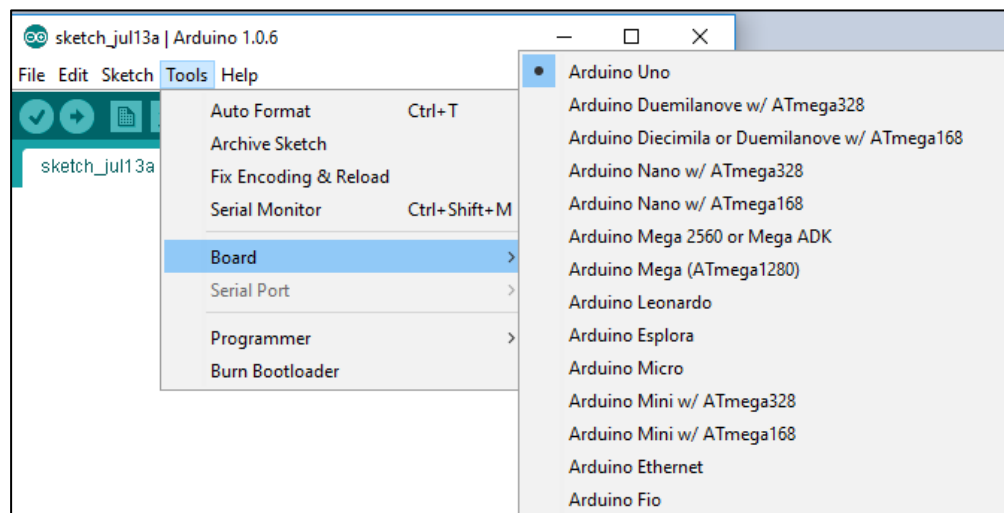
Gambar 2.12 Bagian Menu *Edit* IDE Arduino
(Sumber: Data Penelitian 2020)

Pada gambar 2.12 terdapat bagian menu *Edit* yang terdiri dari *cut*, *copy*, *copy for forum*, *copy as HTML*, *paste*, *select all* dan seterusnya.



Gambar 2.13 Bagian Menu Sketch IDE Arduino
(Sumber: Data Penelitian 2020)

Pada gambar 2.13 terdapat menu sketch yang terdiri dari *verify/compile*, *show sketch folder*, *add file*, dan *add library*.



Gambar 2.14 Bagian Menu Tools IDE Arduino
(Sumber: Data Penelitian 2020)

Gambar 2.14 adalah menu *tools* yang terdiri dari tipe *board* yang dipakai untuk mengirim program, seperti *board* Arduino Uno, Arduino Duemilalove, Arduino Mega, dan seterusnya.

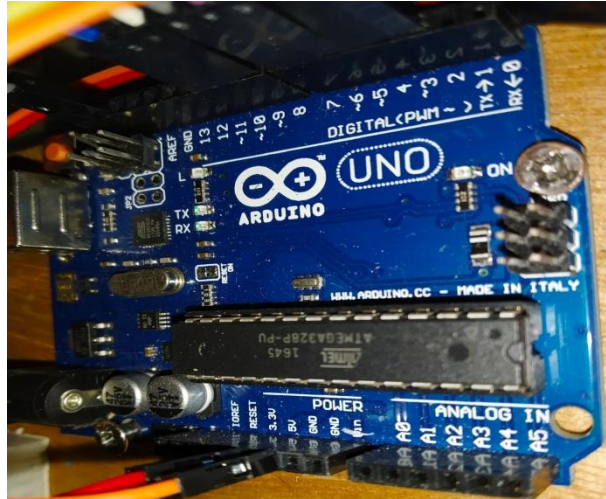
Menurut (Dinata, 2015:11–12) IDE Arduino adalah *software* yang sangat canggih, ditulis menggunakan Java. IDE Arduino terdiri atas:

1. *Editor* program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.
2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *processing*) menjadi kode biner.
3. *Uploader*, sebuah modul yang membuat kode biner dari komputer ke dalam memory di dalam papan Arduino.

Menurut (Istiyanto, 2014:47) Proses kompilasi IDE Arduino diawali dengan proses pengecekan kesalahan sintaksis sketch, kemudian memanfaatkan pustaka *processing* dan *avr-gcc* sketch dikompilasi menjadi berkas *object*, lalu berkas-berkas *object* digabungkan oleh pustaka Arduino menjadi berkas biner. Berkas biner ini diunggah ke cip mikrokontroler via kabel USB, serial port DB9, atau serial *bluetooth*.

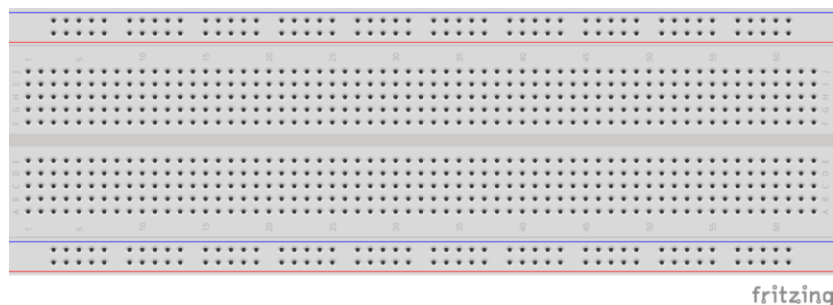
Arduino Uno merupakan mikrokontroler standar dari Arduino. *Board* Uno-R3 merupakan *board* UNO terbaru. Selain memiliki fitur-fitur pada versi sebelumnya, UNO R3 kini menggunakan ATmega16U2 untuk konverter serialnya. Penggunaan ATmega16U2 ini membuat kecepatan transfer menjadi lebih cepat, dan tentu memory yang lebih banyak. Tidak dibutuhkan *driver* tambahan untuk Linux

maupun Mac (namun bagi pengguna Windows akan membutuhkan info terupdate) (Istiyanto, 2014:20).



Gambar 2.15 Board Arduino Uno
(Sumber: Data Penelitian 2020)

Komponen elektronika selanjutnya yang digunakan dalam penelitian tugas akhir ini adalah *breadboard* (papan rangkaian). *Breadboard* adalah *board* yang digunakan untuk membuat rangkaian sementara dengan tujuan uji coba atau prototipe tanpa harus menyolder. *Breadboard* umumnya terbuat dari plastik dengan banyak lubang di atasnya. Lubang-lubang pada *breadboard* diatur sedemikian rupa membentuk pola sesuai dengan pola jaringan koneksi di dalamnya (Syahwil, 2014:21).



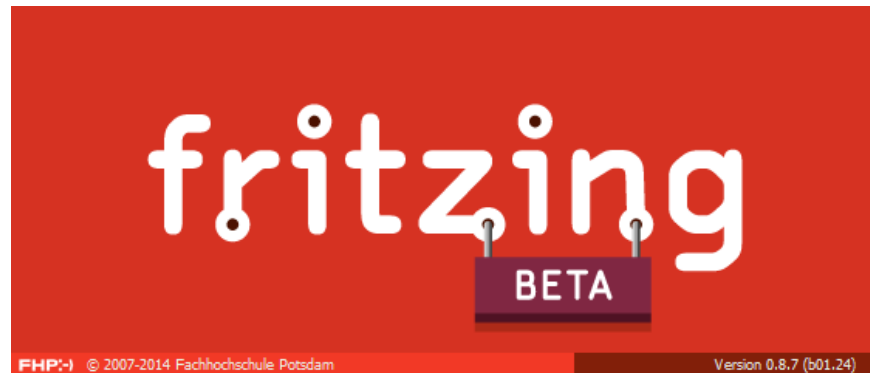
Gambar 2.16 Breadboard
(Sumber: Data Penelitian 2020)

Gambar 2.16 merupakan *breadboard* dengan 400 titik koneksi. Medium *breadboard* ini bisa juga disebut *half* (setengah) *breadboard*, karena ukurannya kurang lebih dari ukuran *large/full breadboard* dengan 830 titik koneksi (Syahwil, 2014:22)

Menurut (Syahwil, 2014:65) Arduino dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya eksternal (*non-USB*) dapat berasal dari adapter AC-ke-DC atau baterai. Adapter ini dapat dihubungkan dengan menancapkan power jack, dapat juga dihubungkan pada power pin (Gnd dan Vin). *Board* Arduino Uno dapat beroperasi pada pasokan eksternal dari 6 sampai 20 volt jika disuplai kurang 7 V. Meskipun, pin 5V dapat disuplai kurang dari 5 volt, *board* mungkin tidak stabil. Jika menggunakan tegangan lebih dari 12V, regulator tegangan bisa panas dan merusak *board*. kisaran yang disarankan adalah 7 sampai 12 volt.

Menurut (Istiyanto, 2014:61) setiap pin I/O digital dapat berfungsi sebagai masukan (*input*) atau keluaran (*output*) logika digital dengan konfigurasinya diatur melalui kode program. Pin-pin tersebut bisa diakses dengan cukup mendefinisikan nomor pin target, tidak seperti bahasa assembler, karena bahasa Arduino adalah bahasa *Processing*.

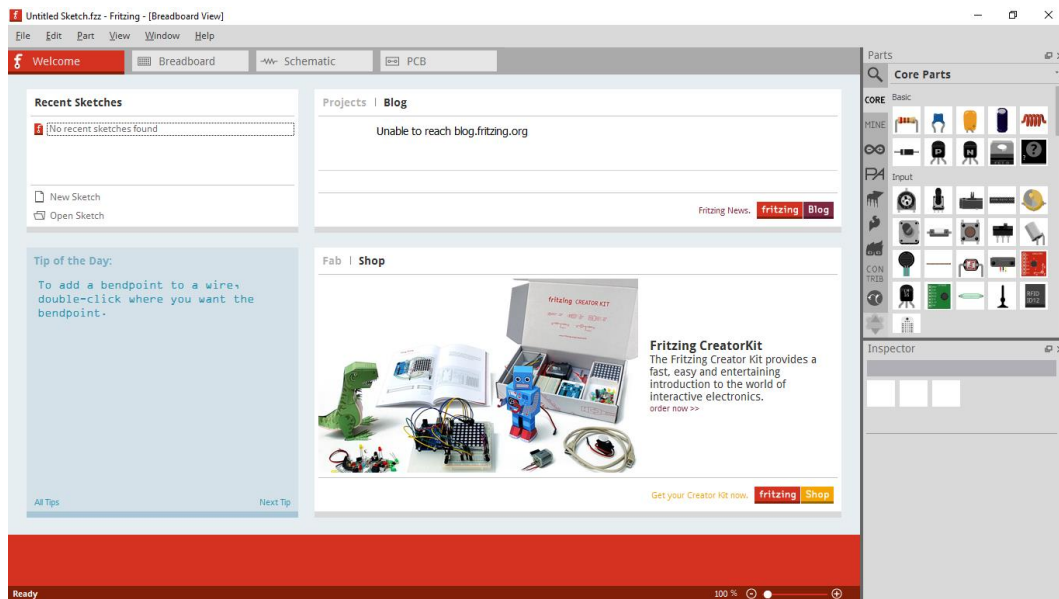
2.2.2 Software Fritzing



Gambar 2.17 *Software Fritzing*
(Sumber: Data Penelitian 2020)

Fritzing adalah sebuah *software* gratis dan sebuah aplikasi *open source* yang dibuat oleh komunitas *online*. *Fritzing* (Ver 0.9 ke atas) dapat dipakai untuk menggambar PCB untuk diproduksi massal. *Fritzing* juga dapat dipakai untuk dokumentasi dan melakukan pemeriksaan desain rangkaian yang akan dibuat. *Fritzing* cukup mudah digunakan dan praktis, karena itu banyak digunakan oleh pengembang modul mikrokontroler Arduino, papan tunggal raspberry-Pi dan sejenisnya (Andrianto & Darmawan, 2016:179)

Menurut (Andrianto & Darmawan, 2016:181) *Fritzing* (Ver 0.9 x) mempunyai 5 menu tab, diantaranya ada tiga menu pilihan (yang berkaitan satu dengan yang lainnya) untuk mendesain rangkaian menggunakan *fritzing* membantu dalam visualisasi koneksi secara fisik, sehingga memudahkan perancangan, pemeriksaan koneksi rangkaian maupun tampilan *layout*. Gambar 2.18 ini merupakan tampilan awal *fritzing* saat pertama kali dibuka.

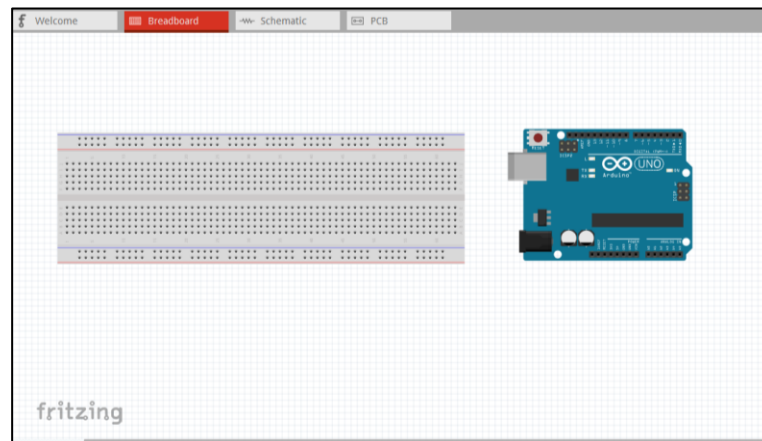


Gambar 2.18 Layout Aplikasi *Fritzing*
(Sumber: Data Penelitian 2020)

Pada gambar 2.18 terdapat beberapa menu yang ditampilkan pada tampilan utama aplikasi *fritzing*, yaitu sebagai berikut:

1. Menu *Breadboard*

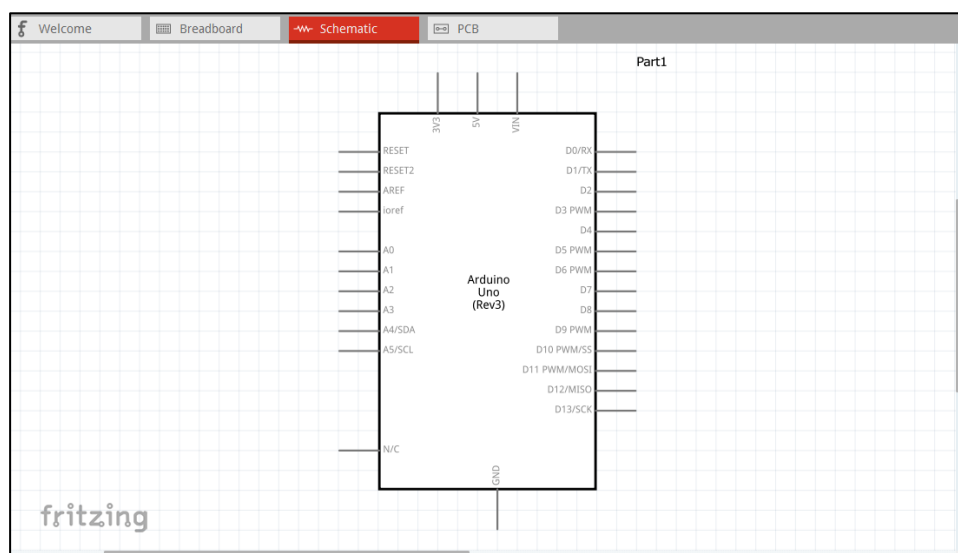
Menu ini digunakan untuk merancang rangkaian dengan menggunakan papan *breadboard* sebagai tempat peletakan komponen-komponen yang digunakan, menu ini sangat membantu dalam visualisasi koneksi secara fisik, sehingga memudahkan perancangan, pemeriksaan koneksi rangkaian maupun untuk tampilan *layout*.



Gambar 2.19 Menu *Breadboard*
(Sumber: Data Penelitian 2020)

2. Menu *Schematic*

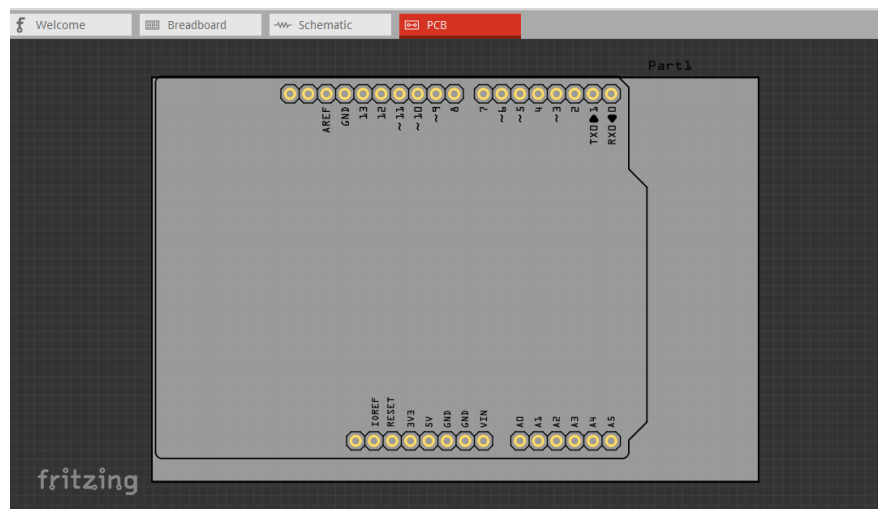
Menu yang dipakai untuk merancang rangkaian dengan menggunakan simbol-simbol komponen elektronika untuk melakukan pengkoreksiannya. Menu ini merupakan menu perancangan rangkaian yang umumnya digunakan dalam berbagai macam perangkat lunak lainnya.



Gambar 2.20 Menu *Schematic*
(Sumber: Data Penelitian 2020)

3. Menu PCB

Menu ini memperlihatkan hubungan antara komponen dalam sebuah papan PCB (satu muka). *Fritzing* juga memberikan fasilitas keluaran *file gerber* dan *file PDF* untuk memudahkan pembuatan PCB secara masal, untuk keperluan produsen PCB.



Gambar 2.21 Menu PCB
(Sumber: Data Penelitian 2020)

2.3 Penelitian Terdahulu

Untuk melaksanakan tugas akhir ini peneliti akan menyampaikan lima penelitian terdahulu yang relevan dengan permasalahan yang sedang diteliti tentang sistem pengunci pintu otomatis berbasis Arduino menggunakan *password* yaitu sebagai berikut:

1. Judul : Rancang Bangun Sistem *Alarm* dan Pintu Otomatis dengan Sensor Gas Berbasis Arduino

Nama Jurnal : Jurnal Teknologi Elektro

Penulis Jurnal : Supegina & Wahyudi

ISSN/Vol./No./Tahun : 2086-9479/4/No.2/2013

Isi Jurnal : Uraian menurut peneliti mengenai jurnal ini yaitu permasalahan utama yang diangkat oleh penulis terkait dengan kecemasan masyarakat terhadap pengalihan kompor minyak ke kompor gas. Sehingga masyarakat merasa was-was akan bahaya kebocoran apabila terjadi kebocoran. Oleh karena itu penulis merancang sebuah sistem *alarm* yang dapat mendeteksi kebocoran dan pintu otomatis dengan menggunakan Arduino. Sistem ini dibuat dengan menggunakan sensor gas MQ 6 khusus untuk mendeteksi gas LPG serta menggunakan *Exhaust fan* dan menggunakan Arduino untuk membuka pintu

2. Judul : Rancang Bangun Akses Kontrol Pintu Gerbang Berbasis Arduino dan Android.

Nama Jurnal : Jurnal Electrans

Penulis Jurnal : Silvia, Haritman, & Muladi

ISSN/Vol./No./Tahun : 1412-3762/13/1/2014

Isi Jurnal : Uraian yang dapat peneliti jelaskan untuk jurnal ini adalah penulis bermaksud memanfaatkan perkembangan teknologi untuk mempermudah pengguna salah satunya dengan perancangan sebuah sistem pintu gerbang otomatis. Sistem ini dirancang dengan menggunakan Arduino dan juga Android. Hasil dari perancangan ini adalah pintu gerbang otomatis

ini dapat dibuka dan ditutup dalam jarak 11meter dengan waktu respon maksimum 1 detik dalam keadaan ruang terbuka.

3. Judul Jurnal : *Modeling & Testing of Automatic Pneumatic Sliding Door Using Sensors & Controllers.*

Nama Jurnal : *International Journal of Scientific and Research Publications.*

Penulis Jurnal : Mohapatra & Anand

ISSN/Vol./No./Tahun : 2250-3151/4/10/201

Isi Jurnal : Pintu Pneumatik Otomatis menggunakan sensor IR berfungsi untuk mengotomatiskan mekanisme operasi pintu menggunakan Pneumatik, Mikrokontroler Arduino dan teknologi sensor inframerah. Metodologi yang diterapkan dalam proyek ini dibagi menjadi tiga bagian, pertama merancang dan membuat pintu dengan dimensi yang dihitung, kedua mengembangkan program di internet Arduino untuk operasi pintu dan ketiga antarmuka yang berbeda komponen untuk bekerja bersama secara kohesig. Ketika suatu objek masuk atau keluar dari kisaran sensor, sinyal dikirim ke Arduino yang mengotrol sirkuit pneumatik untuk membuka atau menutup pintu. Pentingnya sistem ini adalah otomatisasi pintu yang dapat dikustomisasi sesuai dengan industri, komersial atau domestik persyaratan. Berdasarkan hasil yang diperoleh *prototype* kecil dirancang dan kode yang cocok dikembangkan dengan mempertimbangkan cahaya sekitar.

4. Judul Jurnal : Implementasi Sistem Kunci Pintu Otomatis Untuk *Smart Home* Menggunakan SMS Gateway.

Nama Jurnal : *E-Proceeding of Engineering*

Penulis Jurnal : Shandy et al.

ISSN/Vol./No./Tahun : 2355-9365/2/2/2015

Isi Jurnal : Salah satu fitur yang masih digunakan sampai sekarang adalah fitur *Short Message Service* (SMS). Disamping perkembangan aplikasi *chatting* saat ini, fitur SMS sudah mulai ditinggalkan tetapi ternyata fitur ini masih bisa dipergunakan untuk kebutuhan yang lainnya. Pada tugas akhir ini penulis membuat salah satu sistem keamanan rumah berupa sistem kunci pintu otomatis dan menggabungkannya dengan menggunakan fitur pada ponsel yaitu SMS. Sistem ini juga diterapkan fitur untuk memberikan *notifikasi* melalui ponsel dan bisa mengunci pintu dari jarak jauh hanya dengan mengirimkan SMS, apabila pemilik meninggalkan rumahnya tetapi lupa untuk mengunci pintu rumahnya, karena hal ini merupakan salah satu penyebab terjadinya pencurian. Pada sistem ini pun akan dibuat sistem terpusat yang ada di dalam rumah sehingga pemilik tidak perlu menggunakan SMS jika sedang berada di dalam rumah tetapi ingin mengunci pintu. Dengan dilakukan ujicoba yang dilakukan berdasarkan desain sistem kunci pintu dan skenario uji coba yang sudah ditentukan, penulis mendapat kesimpulan bahwa dengan desain sistem ini, sistem pusat dapat bekerja dengan baik dalam mengunci ataupun membuka kunci pintu,

sedangkan fitur sms, *user* akan mendapatkan *notifikasi* pada saat pintu dibuka secara paksa dalam rentang waktu 5-8 detik.

5. Judul Jurnal : *Automatic Opening and Closing Door*

Nama Jurnal : *International Journal of Applied and Pure Science and Agriculture*

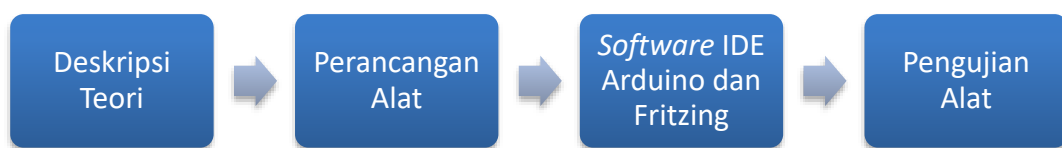
Penulis Jurnal : Panchal, Shinde, Deval, Reddy, & Adeppa

ISSN/Vol./No./Tahun : 2394-5532/02/05/2016

Isi Jurnal : Proyek ini terutama bertujuan merancang sistem akses keamanan otomatis sepenuhnya untuk aplikasi domestik dan industri. Untuk semua modul yang terlibat dalam proyek. Perangkat pengendali seluruh sistem adalah mikrokontroler PIC. Pembaca decoder DTMF, motor stepper, modem GSM, layar LCD dihubungkan ke mikrokontroler. Setiap kali ada panggilan ke telepon dalam sistem, itu akan diwajibkan secara otomatis dan kata sandi dimasukkan melalui telepon papan tombol. Dekoder DTMF mendapatkan kata sandi dan FBI sebagai *input* ke mikrokontroler. Mikrokontroler memvalidasi kata sandi. Jika kata sandi itu valid, ia akan membuka pintu yang terhubung ke motor stepper dihubungkan ke controller. Status pintu ditampilkan pada layar LCD. Ketika ada salah memasukkan kata sandi, sistem memberi peringatan secara otomatis dalam bentuk pesan SMS ke otoritas masing-masing. Mikrokontroler yang digunakan dalam proyek ini diprogram menggunakan bahasa C.

2.4 Kerangka Pikir

(Sugiyono, 2014:60) mengemukakan bahwa, kerangka berfikir merupakan model konseptual tentang bagaimana teori berhubungan dengan berbagai faktor yang telah diidentifikasi sebagai masalah yang penting.



Gambar 2.22 Kerangka Pikir
(Sumber: Data Penelitian 2020)

Gambar 2.22 merupakan kerangka pikir yang digunakan peneliti dalam membuat tugas akhir ini. Dimana setiap teori yang berhubungan dengan permasalahan akan dirumuskan terlebih dahulu, sehingga peneliti dapat mengetahui segala aspek-aspek yang mendukung dalam penelitian ini. Setelah semua teori yang dibutuhkan dideskripsikan, peneliti kemudian membuat rancangan alat yang akan diimplementasikan. Untuk *prototype* alat tersebut peneliti selanjutnya membuat rancangan miniatur kotak berpintu dengan menggunakan google sketchup. Setelah kotak berpintu dirancang, peneliti selanjutnya membuat rangkaian IDE Arduino dengan menggunakan *software* fritzing, sehingga *prototype* yang bisa diuji terlebih dahulu rangkaian serta kabel-kabel yang ada bisa dijalankan. Setelah rangkaian perangkat keras Arduino selesai dibuat dengan menggunakan fritzing. Peneliti kemudian membuat program, di mana program tersebut akan digunakan untuk menjalankan perangkat keras yang tadi telah disusun. Setelah perangkat keras dan

program IDE Arduino siap ditampilkan, peneliti mulai menyusun rangkaian yang ada sesuai dengan rangkaian yang telah disusun di fritzing sebelumnya. Rangkaian yang telah disusun dengan menggunakan *software* IDE Arduino kemudian dilakukan pengujian alat yang bertujuan untuk melihat tingkat keberhasilan dalam perancangan alat tersebut.