

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian yang dilakukan Haritman (2014) yang berjudul Akses Pintu Gerbang Berbasis Arduino dan Android. Penelitian ini menghasilkan sebuah alat kontrol pintu berbasis arduino dan android untuk keamanan rumah yaitu pintu gerbang secara elektronik tanpa harus menggunakan kunci konvensional. Mikrokontroler arduino yang digunakan dapat mengendalikan alat sistem kerja dari kontrol pintu gerbang otomatis dengan bekerja sesuai urutan intruksi pemrograman menggunakan bahasa C. Cara kerja alat ini dengan menggunakan software Eclipse IDE pada PC atau *Smartphone* android yang diberi nama Gerbang Otomatis.apk. Dihubungkan dengan *Bluetooth* yang terdapat pada sistem jika ingin membuka pintu tekan “Buka” jika ingin menutup pintu tekan “Tutup”.

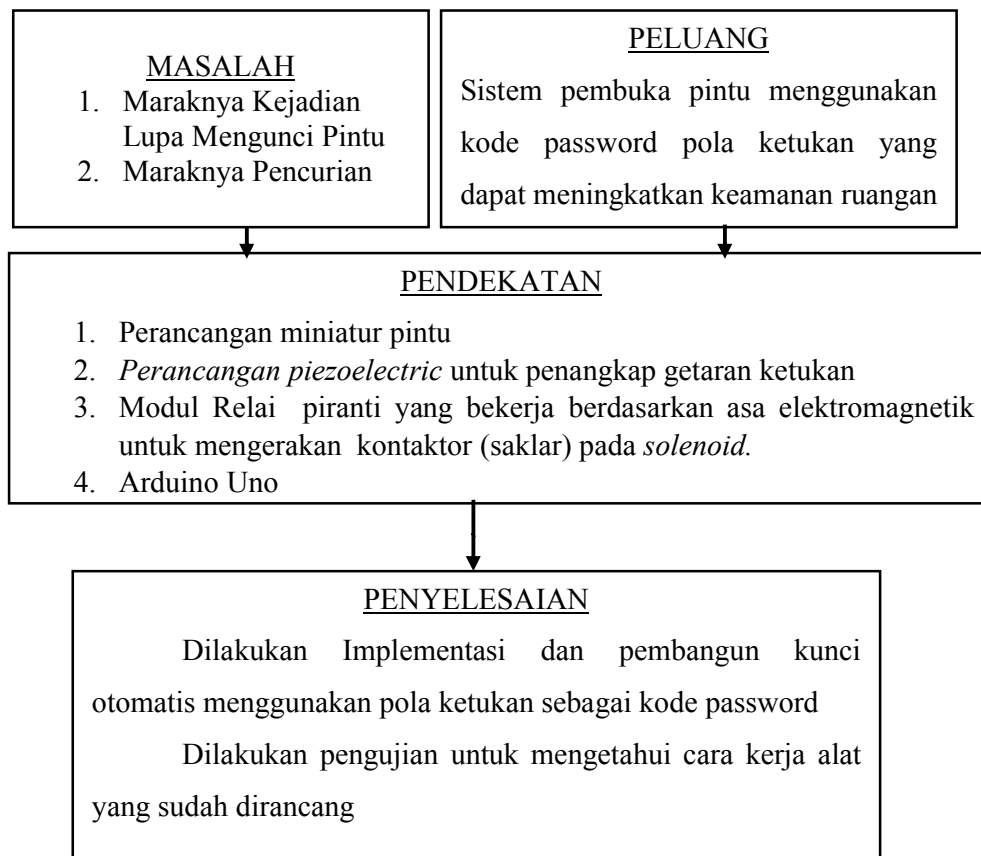
Penelitian yang dilakukan Arafat (2016) yang berjudul Sistem Pengamanan Pintu Rumah Berbasis Internet Of Things (IoT) dengan ESP8266 penelitian ini menghasilkan sebuah alat sistem pembuka pintu otomatis yang mampu dikendalikan oleh aplikasi android yang telah disesuaikan sebagai metode kontrol akses, alat yang dibuat juga dapat memeriksa keadaan pintu dan keadaan solenoid yang kemudian akan dikirimkan kepada pengguna sebagai notifikasi melalui aplikasi blynk. Cara kerja alat ini dengan menggunakan wifi sebagai penghubung sistem dengan smartphone pengguna, jika ingin membuka pintu tekan “Buka” jika ingin menutup tekan “Tutup” jika ada yang membuka pintu secara paksa, sistem akan mengirim pesan melalui SMS kepada smartphone pengguna.

Penelitian yang dilakukan Yogie (2016) yang berjudul *Prototype* Penggerak Pintu Pagar Otomatis Berbasis Arduino ATMEGA 328P dengan Sensor Sidik Jari, penelitian ini menghasilkan *Prototype* penggerak pintu pagar otomatis menggunakan sensor sidik jari yang digunakan sebagai koding aktivasi penggerak motor, secara matematis motor DC yang digunakan pada penelitian dapat menggerakkan benda maksimal sebesar 69,02 kg karena konstruksi benda

yang kurang sempurna dan pergesekan antara grid dan rantai. Cara kerja alat ini dengan menggunakan sidik jari sebagai pembuka pintu pagar

2.2 Kerangka Pemikiran

Kerangka pemikiran adalah suatu diagram yang menjelaskan secara garis besar alur logika yang berjalannya sebuah penelitian. Dalam penulisan laporan yang berjudul Rancang Bangun Kunci Otomatis Menggunakan Pola Ketukan Berbasis Arduino, kerangka pemikiran dijelaskan pada gambar 2.1



Gambar 2.1 Bagan Kerangka Pemikiran

1. Masalah

Maraknya lupa mengunci pintu hal itu memberikan kesempatan bagi para pelaku pencurian untuk melakukan aksinya.

2. Peluang

Permasalahan lupa mengunci pintu dan maraknya pencurian menghasilkan sebuah peluang untuk mengatasinya, yaitu dengan pembuatan Sistem pembuka pintu menggunakan kode password pola ketukan yang dapat meningkatkan keamanan ruang.

3. Pendekatan

Pendekatannya dengan merancang miniatur pintu, merancang *Piezoelectric* untuk penangkap getaran ketukan dan menggunakan modul relai untuk mengerjakan sejumlah kontaktor (saklar) pada *solenoid* dan Arduino Uno sebagai otak dari sebuah sistem pintu otomatis.

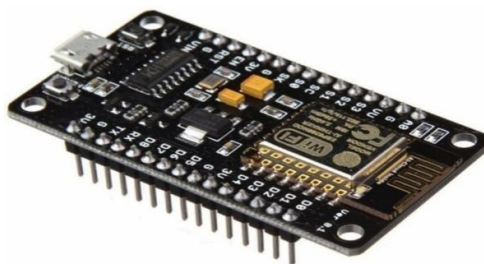
4. Penyelesaian

Penyelesaiannya dengan dilakukan pembangunan dari rangkaian – rangkaian yang sudah dibuat agar menghasilkan sebuah kunci otomatis yang dibuka dengan menggunakan pola ketukan sebagai kode *password* lalu melakukan pengujian untuk mengetahui cara kerja alat yang sudah dirancang.

2.3 Teori Pendukung

2.3.1 *Microcontroller*

Menurut wahyuni (2015) *Microcontroller* adalah suatu *chip* berupa IC (*Integrated Circuit*) yang dapat menerima sinyal *input*, mengolahnya dan memberikan sinyal *output* sesuai dengan program yang diisikan ke dalamnya. Sinyal *input* diperoleh dari sensor yang dipasang pada *microcontroller*, sedangkan sinyal *output* akan dikeluarkan berdasarkan *trigger* yang diperoleh *microcontroller*.



Gambar 2.2. *Microcontroller*

strain mekanis dalam merespon medan listrik yang teraplikasi pada bahan tersebut.

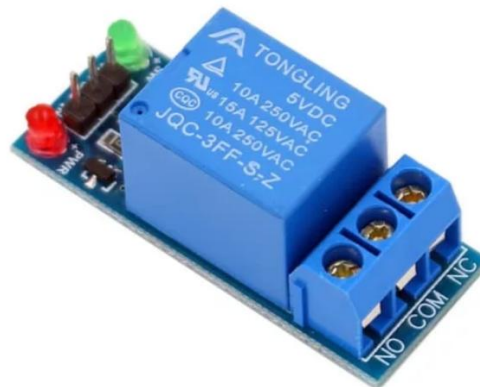


Gambar 2.4 *Piezoelectric*

Piezoelectric memiliki sifat reversibel, yakni efek langsung yaitu menimbulkan potensial listrik akibat tekanan mekanik dan efek tak langsung yaitu timbulnya tekanan akibat pemberian tegangan listrik yang mengakibatkan perubahan dimensi

2.3.4 Modul Relay

Menurut kholid (2018) *relay* merupakan saklar (*switch*) yang dioperasikan secara listrik dan memiliki dua bagian utama, yakni electromagnet (*coil*) dan mekanikal (seperangkat kontak saklar atau *switch*). Pada dasarnya *relay* merupakan sebuah saklar yang membuka dan menutup (*open* dan *close*) dengan tenaga listrik melalui *coil* yang terdapat didalamnya.



Gambar 2.5 Modul *Relay*

Relay terdiri dari *coil* dan *contact*, *coil* merupakan sebuah gulungan kawat yang mendapat arus listrik. Sedangkan *contact* merupakan sejenis saklar yang penggerakannya tergantung ada atau tidaknya arus listrik pada *coil*.

2.3.5 Push Button

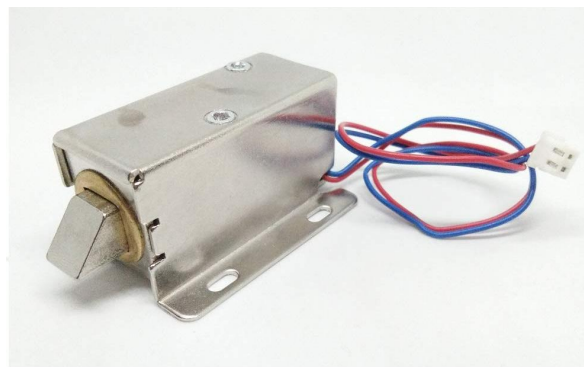
Menurut Iskandar, Akbar (2017) Pada *Push Button (PB)*, terdapat kontak - kontak yang berupa *normaly close (NC)* dan *Normaly open (NO)*, atau ada juga *push button* yang memiliki jumlah kontak yang lebih banyak.



Gambar 2.6 *Push Button*

2.3.6 Solenoid Door Lock

Menurut Iskandar, Akbar (2017) *Solenoid Door Lock* adalah alat elektronik yang dibuat khusus untuk pengunci pintu. Alat ini sering digunakan pada kunci pintu otomatis. *Solenoid* ini akan bergerak atau bekerja apabila diberi tegangan. Tegangan *solenoid* kunci pintu ini rata-rata yang di jual dipasaran 12 volt tapi ada juga yang 6 volt dan 24 volt.



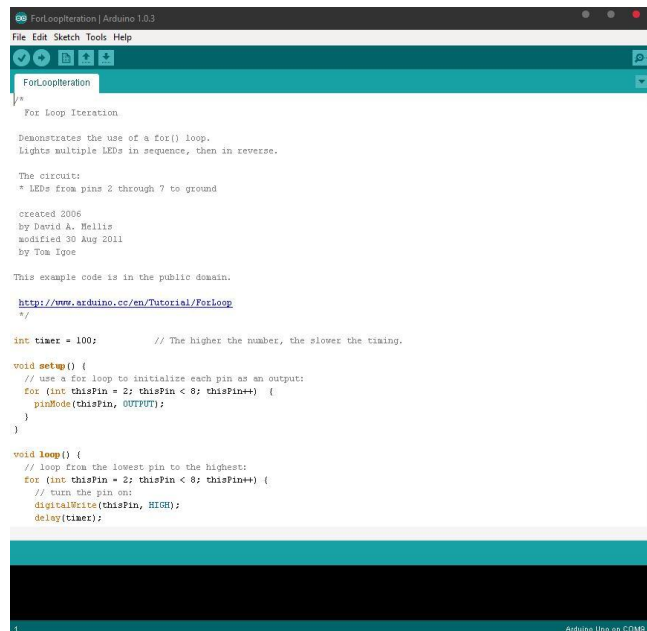
Gambar 2.7 *Solenoid Door Lock*

2.3.7 Arduino IDE

Menurut Rohmanu dan Widiyanto (2018) Arduino IDE (*integrated development environment*) adalah sebuah software untuk menulis program, mengkompilasi menjadi biner dan mengupload ke dalam *memory microcontoroller*.

Menurut Jauhari (2016) Arduino diciptakan untuk para pemula bahkan yang tidak memiliki basic bahasa pemrograman sama sekali karena menggunakan bahasa C++ yang telah dipermudah melalui *library*. Arduino IDE adalah sebuah program yang berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory microcontroller*. *Software IDE* terdiri dari tiga bagian:

- Editor program*, untuk menulis dan mengedit program dalam bahasa *processing*. *Listing* program pada Arduino yang disebut *sketch*.
- Compiler*, modul yang berfungsi mengubah bahasa *processing* (kode program) ke dalam kode biner karena kode biner adalah satu-satunya bahasa program yang dipahami oleh *microcontroller*.
- Uploader*, modul yang berfungsi memasukan kode biner kedalam *memory microcontroller*.



```

ForLoopIteration | Arduino 1.8.3
File Edit Sketch Tools Help
ForLoopIteration
/*
 * For Loop Iteration.
 *
 * Demonstrates the use of a for() loop.
 * Lights multiple LEDs in sequence, then in reverse.
 *
 * The circuit:
 * - LEDs from pins 2 through 7 to ground
 *
 * Created 2006
 * by David A. Mellis
 * modified 30 Aug 2011
 * by Tom Igoe
 *
 * This example code is in the public domain.
 *
 * http://www.arduino.cc/en/Tutorial/ForLoop
 */

int timer = 100; // The higher the number, the slower the timing.

void setup() {
  // use a for loop to initialize each pin as an output:
  for (int thisPin = 2; thisPin < 8; thisPin++) {
    pinMode(thisPin, OUTPUT);
  }
}

void loop() {
  // loop from the lowest pin to the highest:
  for (int thisPin = 2; thisPin < 8; thisPin++) {
    // turn the pin on:
    digitalWrite(thisPin, HIGH);
    delay(timer);
  }
}

```

Gambar 2.8 Arduino IDE

2.3.8 Black Box Testing



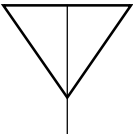
Menurut Dharma, Yadi (2018) *Black Box Testing* adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat suatu kotak hitam, kita hanya bisa melihat tampilan luarnya saja, tanpa tau ada apa dibalik bungkus hitamnya (hanya mengetahui *input output*). Pengujian *Black Box* berusaha menemukan kesalahan dalam kategori:

1. Fungsi – fungsi yang tidak benar
2. Kesalahan *Interface*
3. Kesalahan kinerja
4. Inisialisasi dan kesalahan terminasi

2.3.9 Block Diagram

Menurut Giblisco (2014) *Block diagram* menggambarkan perakitan secara umum dari sebuah perangkat elektronik atau sistem. Sebuah *block diagram* menyediakan versi sederhana dari sebuah perangkat dengan memisahkan komponen inti dan menunjukkan bagaimana perangkat tersebut dihubungkan. *Block diagram* biasanya terfokus pada *input* dan *output* dari sebuah sistem dan biasanya tidak memperhitungkan hasil *input* maupun *output*. Adapun simbol yang digunakan dalam *block diagram* ditunjukkan pada Tabel 2.1 berikut.

Tabel 2.1 Simbol dalam *Block Diagram*

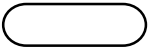
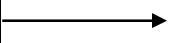
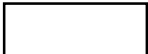
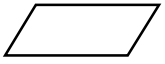
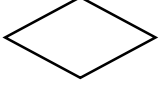
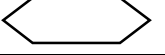



No	Simbol	Fungsi
1		Untuk menggambarkan modul atau fungsi utama dalam sebuah blok. Didalam simbol tersebut wajib ditulis nama modul atau fungsi.
2		Anak panah digunakan untuk menggambarkan arah aliran sinyal <i>input</i> maupun <i>output</i> dari atau ke sebuah <i>block</i> .
3		Untuk menggambarkan sebuah <i>integrated circuits</i> atau IC yang dibangun didalam sebuah <i>amplifier</i> khusus, biasanya antena atau pemancar sinyal lain.

2.3.10 Flowchart (Diagram Alir)

Menurut Barakbah & Karlita (2013) *Flowchart* adalah cara penulisan algoritma dengan menggunakan notasi grafis. *Flowchart* merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses beserta pernyataannya. Gambaran ini dinyatakan dengan simbol. Sehingga setiap simbol menggambarkan proses tertentu.

Adapun simbol-simbol yang digunakan dalam *flowchart* standar yang ditetapkan oleh ISO dan ANSI ditunjukkan pada Tabel 2.2 berikut.

Tabel 2.2 Simbol dalam *flowchart*

No.	Simbol	Nama	Fungsi
1		<i>Terminator</i>	Simbol awal (<i>start</i>) atau simbol akhir (<i>end</i>).
2		<i>Flow Line</i>	Simbol arah aliran atau penghubung.
3		Proses	Simbol perhitungan atau pengolahan angka.
4		<i>Input / Output</i>	Simbol untuk merepresentasikan pembacaan data (<i>read</i>) atau penulisan data (<i>write</i>).
5		<i>Decision / Pilihan</i>	Simbol untuk merepresentasikan suatu pernyataan pilihan, berisi suatu kondisi dengan <i>output</i> benar atau salah.
6		<i>Preparation</i>	Simbol pernyataan inisialisasi atau pemberian nilai awal.
7		<i>Predefined Process (Subprogram)</i>	Proses menjalankan subprogram / fungsi / prosedur.
8		<i>On Page Connector</i>	Simbol penghubung <i>flowchart</i> jika masih dalam satu halaman.
9		<i>Off Page Connector</i>	Simbol penghubung <i>flowchart</i> apabila sudah berganti halaman.

