

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam pembuatan alat pendeteksi kebakaran berbasis Arduino, ada beberapa penelitian yang terkait sebagai dasar dari penelitian ini. Beberapa penelitian tersebut akan dijelaskan sebagai berikut:

Penelitian dari Tyas (2016) dengan judul Aplikasi Peminjaman Gedung Pertemuan Kantor Kelurahan Kutowinangan Lor Kota Salatiga berbasis SMS *Gateway* membahas tentang aplikasi SMS *gateway* yang dapat mengirimkan informasi gedung langsung ke *handphone* pengguna yang mempermudah pengguna dalam pemesanan gedung melalui SMS dan menghasilkan informasi status gedung melalui *handphone* secara langsung. Metode perancangan yang pertama adalah pengumpulan data yaitu observasi, wawancara, dan dokumentasi. Metode selanjutnya adalah pengembangan sistem menggunakan metode *Linear Sequential Model* model *waterwall* diantaranya analisi dan perencanaan sistem, perancangan sistem perangkat lunak, implementasi, pengujian menggunakan *blackbox*. Hasil dari perancangan sistem ini adalah menyajikan informasi jadwal peminjaman secara langsung via SMS, juga mencetak laporan peminjaman. Sistem ini diharapkan mampu membantu dan mempermudah proses pengelolaan peminjaman gedung serta meningkatkan pelayanan publik kepada warga masyarakat.

Penelitian dari Mulya (2015) dengan judul *Prototype* Monitoring Kebakaran Hutan Via *Website* Berbasis Arduino membahas tentang suatu sistem yang mampu memonitoring adanya indikasi kebakaran hutan. Teknologi *wireless* yang mampu mengirimkan data tanpa perlu menggunakan kabel diharapkan mampu menjadi salah satu perkembangan teknologi aplikatif yang dapat mendukung program kelestarian hutan. *Prototype* monitoring kebakaran hutan via *website* berbasis Arduino menggunakan sensor suhu LM35 untuk mendeteksi suhu, Arduino sebagai pengendali utama dan untuk memproses data, modem Wavecom FXT009 sebagai sumber jaringan *wireless*, buzzer sebagai penanda

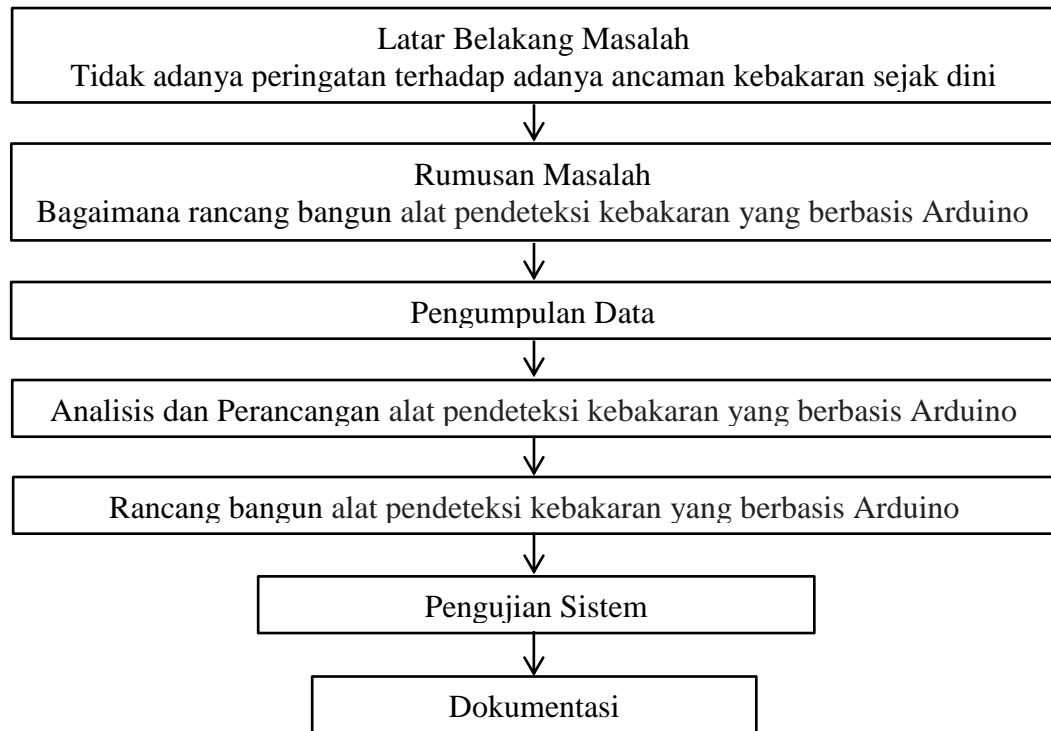
bunyi, dan website sebagai tampilannya. Hasil dari penelitian tersebut adalah dapat mendeteksi suhu di hutan, mengirimkan data melalui via internet nirkabel, mengirimkan data setiap 20 detik, menampilkan data pada *website* dan memberikan peringatan bila terjadi kebakaran hutan.

Penelitian dari Sasmoko dan Mahendra (2017) dengan judul Rancang Bangun Sistem Pendeteksi Kebakaran Berbasis IOP dan SMS *Gateway* Menggunakan Arduino membahas tentang sistem yang dapat menginformasikan kebakaran hutan secara dini dipantau dari kantor pengawas untuk mengurangi atau menanggulangi agar kebakaran tidak semakin membesar dan dapat ditangani secara cepat. Penelitian tersebut bertujuan untuk mengurangi resiko kebakaran dan tingkat penyakit yang disebabkan oleh asap kebakaran hutan. Alat-alat yang mendukung dalam penelitian ini yaitu Arduino Uno R3, Sensor Asap MQ-7, Sensor Suhu LM35, Sensor Api, GSM/GPRS Shield SIM900. Prinsip kerja pada keadaan awal alat diaktifkan, sensor suhu akan menampilkan laporan suhu secara *real-time* pada komputer/laptop yang berada pada pos sebagai acuan keadaan hutan, yg pertama adalah jika suhu normal <35 C maka indikator warna hijau akan menyala, selanjutnya adalah jika sensor asap mendeteksi adanya asap maka indikator warna kuning yang akan tampil di status peringatan pada monitoring web, dan jika suhu mencapai angka >45 C maka indikator merah akan menyala sebagai peringatan bahaya kebakaran dan ketika sensor api mendeteksi adanya nyala api maka secara otomatis GSM/GPRS Shield SIM900 akan mengirimkan pesan SMS ke semua petugas pos jaga dan perwakilan penduduk setempat.

Penelitian dari Yendril, dkk. (2017) dengan judul Perancangan Sistem Pendeteksi Kebakaran Rumah Penduduk Pada Daerah Perkotaan Berbasis Mikrokontroler membahas tentang perancangan sistem pendeteksi kebakaran rumah penduduk pada daerah perkotaan yang berbasis mikrokontroler. Sistem ini bekerja berdasarkan prinsip penginderaan asap dan nyala api menggunakan *flame* sensor, mentransmisi data secara nirkabel (*wireless*), dan menampilkan lokasi kejadian pada Google Maps secara otomatis ke satuan pemadam kebakaran terdekat. Perangkat yang digunakan dalam penelitian antara lain sensor asap MQ-9, sensor api V2, sensor suhu LM35, Arduino Uno, dan modul wi-fi ESP8266.

2.2 Kerangka Pemikiran

Kerangka pemikiran adalah tahapan dalam rancang bangun alat pendeteksi kebakaran berbasis Arduino yang ditunjukkan pada pada Gambar 2.1.



Gambar 2.1. Kerangka Pemikiran

Penjelasan dari kerangka pemikiran yang ditunjukkan pada Gambar 2.1 adalah sebagai berikut:

1. Latar Belakang Masalah

Kebakaran semakin meningkat karena tidak adanya peringatan terhadap adanya ancaman kebakaran sejak dini. Dengan adanya alat pendeteksi kebakaran diharapkan dapat memperkecil resiko kebakaran dan mencegah api semakin besar. Dengan begitu penanganan terhadap kebakaran dapat dilakukan sejak dini.

2. Rumusan Masalah

Berdasarkan uraian pada latar belakang maka perumusan masalah dalam penelitian ini adalah bagaimana cara menciptakan sebuah alat pendeteksi kebakaran dengan sensor gas/asap dan sensor api menggunakan *SMS gateway* yang berbasis Arduino.

3. Pengumpulan data

Tahap pengumpulan data pada penelitian ini melalui observasi, wawancara dan studi literatur. Pengumpulan data bertujuan untuk mengetahui permasalahan dan kebutuhan informasi mengenai alat pendeteksi kebakaran berbasis Arduino.

4. Analisis dan perancangan sistem

Analisis dan perancangan sistem alat pendeteksi kebakaran berbasis Arduino bertujuan untuk mengetahui kesiapan data dan rancangan yang diharapkan dapat memberikan suatu informasi dengan efektif.

5. Rancang Bangun Sistem

Rancang bangun sistem adalah tahapan-tahapan yang akan dilakukan dalam perancangan alat pendeteksi kebakaran dengan sensor gas/asap dan sensor api menggunakan *SMS gateway* berbasis Arduino.

6. Pengujian Sistem

Pengujian sistem merupakan tahap setelah sistem berhasil dibuat. Pengujian sistem bertujuan untuk mengetahui kelebihan, kelemahan serta mengetahui kelayakan suatu sistem untuk digunakan. Pengujian sistem dalam penelitian ini menggunakan metode *Black Box* testing yang dilakukan terhadap alat pendeteksi kebakaran berbasis Arduino.

7. Dokumentasi

Tahap terakhir adalah dokumentasi, dimana dokumentasi adalah aktivitas atau proses penyediaan dokumen-dokumen dengan menggunakan bukti yang akurat berdasarkan pencatatan pada penelitian rancang bangun alat pendeteksi kebakaran berbasis Arduino yang telah dilakukan.

2.3 Landasan Teori

Dalam rancang bangun alat pendeteksi kebakaran menggunakan *SMS gateway* yang berbasis Arduino terdapat beberapa landasan teori yang digunakan dalam penelitian ini dijelaskan sebagai berikut:

2.3.1 Rancang Bangun

Rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan (Pressman, 2012). Rancang sistem adalah penentuan proses dan data yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik (Ladjamudin, 2005). Sedangkan pengertian bangun atau pembangunan sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian (Pressman, 2012).

Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil analisis ke dalam bentuk suatu perangkat yang menciptakan sistem yang diciptakan untuk menyelesaikan masalah yang sedang dihadapi.

2.3.2 Penyebab Terjadinya Kebakaran

Kebakaran dapat terjadi karena manusia, peristiwa alam, penyalahan sendiri dan unsur kesengajaan (Triyono, 2001).

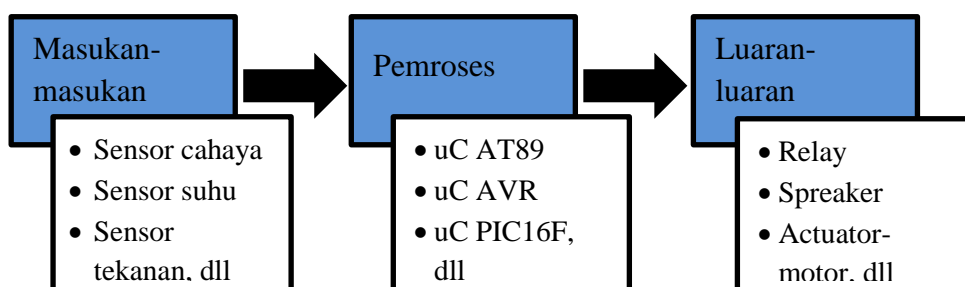
1. Kebakaran karena manusia bersifat kelalaian, seperti:
 - a. Kurangnya pengertian, pengetahuan tentang penanggulangan bahaya kebakaran.
 - b. Kurang hati-hati dalam menggunakan alat atau bahan yang dapat menimbulkan api.
 - c. Kurangnya kesadaran pribadi atau tidak disiplin.
2. Kebakaran karena bersifat peristiwa alam terutama menyangkut cuaca dan gunung berapi seperti sinar matahari, letusan gunung berapi, gempa bumi, petir, angin dan topan.
3. Kebakaran karena penyalahan sendiri, sering terjadi pada gudang-gudang bahan kimia dimana bahan-bahan tersebut bereaksi dengan udara, air dan juga dengan bahan-bahan lainnya yang mudah meledak atau terbakar.
4. Kebakaran karena unsur kesengajaan, untuk tujuan-tujuan tertentu, misalnya:
 - a. Sabotase untuk menimbulkan huru-hara, kebanyakan dengan alasan politis.

- b. Mencari keuntungan pribadi karena ingin mendapatkan ganti rugi melalui asuransi kebakaran.
- c. Untuk menghilangkan jejak kejahatan dengan cara membakar dokumen atau bukti–bukti yang dapat memberatkannya.
- d. Untuk jalan taktis dalam pertempuran dengan jalan bumi hangus.

2.3.3 Mikrokontroler

Menurut Putra (2010), jika kita bicara tentang Mikrokontroler, maka tidak terlepas dari pengertian atau definisi tentang Komputer. Hal ini disebutkan dengan kesamaan-kesamaan antara Mikrokontroler dengan Komputer (atau Mikrokomputer), antara lain:

- a. Sama-sama memiliki unit pengolah pusat atau yang lebih dikenal dengan CPU (*Central Processing Unit*).
- b. CPU tersebut sama-sama menjalankan program dari suatu lokasi atau tempat, biasanya dari ROM (*Read Only Memory*) atau RAM (*Random Access Memory*).
- c. Sama-sama memiliki RAM yang digunakan untuk menyimpan data-data sementara atau yang lebih dikenal dengan variable-variabel.
- d. Sama-sama memiliki beberapa luaran dan masukan (I/O) yang digunakan untuk melakukan komunikasi timbal-balik dengan dunia luar, melalui sensor (masukan) dan *actuator* (luaran), perhatikan bagan yang ditunjukkan pada Gambar 2.2 berikut:



Gambar 2.2. Bagan Masukan, Pemrosesan Hingga Luaran (Putra, 2010).

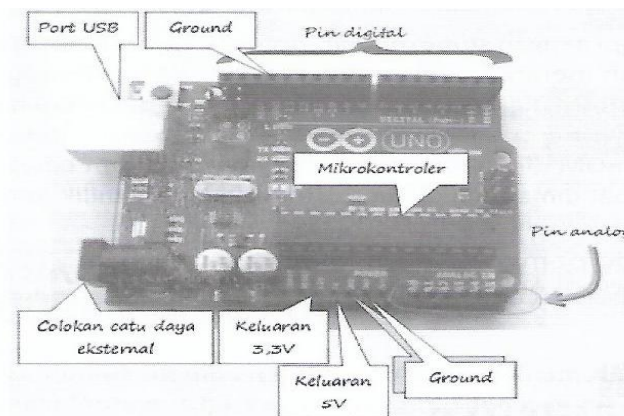
Sedangkan yang membedakan antara Mikrokontroler dengan Komputer atau Mikrokomputer yaitu Mikrokontroler adalah versi mini dan untuk aplikasi khusus dari Mikrokomputer atau Komputer.

2.3.4 Arduino

Arduino adalah nama keluarga papan mikrokontroler yang awalnya dibuat oleh perusahaan Smart Projects. Salah satu tokoh penciptanya adalah Massimo Banzi. Papan ini merupakan perangkat keras yang bersifat *open source* sehingga boleh dibuat oleh siapa saja. Arduino dibuat dengan tujuan memudahkan eksperimen atau perwujudan berbagai peralatan yang berbasis mikrokontroler (Kadir, 2018).

2.3.5 Arduino uno

Arduino Uno berukuran sebesar kartu kredit. Walaupun berukuran kecil seperti itu, papan tersebut mengandung mikrokontroler dan sejumlah *input/output* (I/O) yang memudahkan pemakai untuk menciptakan berbagai proyek elektronika yang dikhususkan untuk menangani tujuan tertentu. Bagian-bagian di Arduino Uno yang perlu diketahui terlebih dahulu ditunjukkan pada Gambar 2.3 berikut:



Gambar 2.3. Arduino Uno (Kadir, 2018).

Penjelasan bagian masing-masing pada Arduino Uno yang ditunjukkan pada Gambar 2.3 adalah sebagai berikut:

1. *Port* USB digunakan untuk menghubungkan Arduino Uno dengan komputer, melalui sepasang kabel USB.
2. Colokan catu daya eksternal digunakan untuk memasok sumber daya listrik diberi untuk Arduino Uno dihubungkan ke komputer melalui kabel USB, pasokan daya listrik diberi oleh komputer.
3. Pin digital mempunyai label 0 sampai dengan 13. Disebut pin digital karena mempunyai isyarat digital, yakni berupa 0 atau 1. Dalam praktik, nilai 0 dinyatakan dengan tegangan 0V dan nilai 1 dinyatakan dengan tegangan 5V.

4. Pin analog berarti bahwa pin-pin ini mempunyai nilai yang bersifat analog (nilai yang berkesinambungan). Dalam program, nilai setiap pin analog yang berlaku sebagai masukan (hasil dari sensor) berkisar antara 0 sampai dengan 1023.
5. Mikrokontroler yang digunakan di Arduino Uno adalah Atmega328.
6. Ada dua pin yang dapat digunakan untuk memasok catu daya ke komponen elektronis yang digunakan dalam menangani proyek, misalnya sensor gas, sensor jarak, dan relay. Tegangan yang tersedia adalah 3,3V dan 5V. Komponen-komponen elektronis yang diberi tegangan oleh Arduino Uno adalah yang memerlukan arus kecil. Sebagai contoh, Motor DC yang menarik arus lebih dari 500mA harus menggunakan daya tersendiri.

Arduino Uno dilengkapi dengan dengan *static random-access memory* (SRAM) berukuran 2KB untuk memegang data, *flash memory* berukuran 32KB, dan *erasable programmable read-only memory* (EEPROM). SRAM digunakan untuk menampung data atau hasil pemrosesan data selama Arduino menerima posokan catu daya. *Flash memory* untuk menaruh program yang Anda buat. EEPROM digunakan untuk menaruh program bawaan dari Arduino Uno dan sebagian lagi dapat dimanfaatkan untuk menaruh data milik Anda secara permanen (Kadir, 2018).

2.3.6 Sensor Api

Sensor api adalah sensor yang ditujukan untuk mendeteksi api dan radiasi. Sensor ini juga dapat digunakan untuk mendeteksi sumber cahaya dengan panjang gelombang dengan jangkauan gelombang 760 nm hingga 1100 nm. Contoh sensor ini salah satunya adalah sensor api KY-026. Sensor tersebut mampu mendeteksi dari 20 cm hingga pada jarak 100 cm (Kadir, 2018).

2.3.7 Sensor Gas/Asap

Sensor asap/gas adalah sensor yang ditujukan untuk mendeteksi asap maupun gas. Salah satu kelompok sensor gas asap yang terkenal berserikan MQ, sensor ini berguna untuk mendeteksi keberadaan gas didalam ruangan tertutup (Kadir, 2018).

2.3.8 *Buzzer Alarm*

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan *speaker*, jadi *buzzer* juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara (Sulistiyowati dan Febriantorodi, 2012).

2.3.9 *Kabel Jumper*

Kabel *jumper* atau kawat lompat adalah kawat listrik, atau kelompok mereka dalam kabel, dengan konektor atau pin di setiap ujungnya, yang biasanya digunakan untuk menghubungkan komponen papan *breadboard* atau prototipe lain atau rangkaian uji, secara internal atau dengan peralatan atau komponen lain, tanpa solder (Wikipedia).

2.3.10 *Modul GSM*

Modul GSM adalah peralatan yang didesain supaya dapat digunakan untuk aplikasi komunikasi dari mesin ke mesin atau dari manusia ke mesin. Modul GSM merupakan peralatan yang digunakan sebagai mesin dalam suatu aplikasi. Dalam aplikasi yang dibuat harus terdapat mikrokontroler yang akan mengirimkan perintah kepada modul GSM berupa AT command melalui RS232 sebagai komponen penghubung (*communication links*).

Modul GSM merupakan bagian dari pusat kendali yang berfungsi sebagai *transceiver*. Modul GSM mempunyai fungsi yang sama dengan sebuah telepon seluler yaitu mampu melakukan fungsi pengiriman dan penerimaan SMS. Dengan adanya sebuah modul GSM maka aplikasi yang dirancang dapat dikendalikan dari jarak jauh dengan menggunakan jaringan GSM sebagai media akses (Kadir, 2018).

2.3.11 SMS

SMS (*Short Message Service*) merupakan layanan yang banyak diaplikasikan pada sistem komunikasi tanpa kabel (nirkabel), memungkinkan dilakukannya pengiriman pesan dalam bentuk *alphanumeric* antar terminal pelanggan atau antar terminal pelanggan dengan sistem eksternal. SMS berupa pesan teks, jumlah karakter pada setiap pengiriman bergantung pada operatornya. Operator selular di Indonesia umumnya membatasi 160 karakter untuk satu pengiriman dan penerimaan SMS. Selain itu SMS merupakan metode *store* dan *forward* sehingga keuntungan yang didapat adalah pada saat telepon selular penerima tidak dapat dijangkau, dalam arti tidak aktif atau diluar *service area*, penerima tetap dapat menerima SMS-nya apabila telepon selular tersebut sudah aktif kembali.

SMS menyediakan mekanisme untuk mengirimkan pesan singkat dari dan menuju media-media *wireless* dengan menggunakan sebuah *Short Message Service Centre* (SMSC), yang bertindak sebagai sistem yang berfungsi menyimpan dan mengirimkan kembali pesan-pesan singkat. jaringan *wireless* menyediakan mekanisme untuk menemukan *station* yang dituju dan mengirimkan pesan singkat antara SMSC dengan *wireless station*. SMS mendukung banyak mekanisme *input* sehingga memungkinkan adanya interkoneksi dengan sumber dan tujuan pengiriman pesan yang berbeda (Riadi, 2012).

2.3.12 SMS Gateway

SMS *Gateway* adalah teknologi mengirim, menerima dan bahkan mengolah SMS melalui komputer dan sistem komputerisasi biasanya digunakan pada aplikasi bisnis baik kepentingan promosi, penyebaran informasi pada pengguna. Seperti kita ketahui, pada jaman sekarang, hampir semua individu telah memiliki telepon selular (*handphone*), bahkan ada individu yang memiliki lebih dari satu *handphone*. SMS merupakan salah satu fitur pada *handphone* yang pasti digunakan oleh pengguna (*user*), baik untuk mengirim, maupun untuk menerima SMS.

Bagi perusahaan, hal ini dimanfaatkan dengan baik dalam hal pemasaran dan pengumuman terhadap pelanggan (*customer*) perusahaan mereka. Data nomor

handphone disimpan dalam database perusahaan dan ketika terdapat informasi atau layanan terbaru dapat memanfaatkan *SMS gateway* dalam proses informasinya (harga murah, cepat, dan mudah) dengan menggunakan sistem komputerisasi (Faesal, 2012).

2.3.13 Diagram Blok

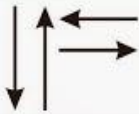
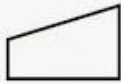






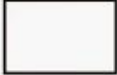







Diagram blok adalah diagram sistem di mana bagian atau fungsi utama diwakili oleh blok yang dihubungkan oleh garis yang menunjukkan hubungan blok. Mereka banyak digunakan dalam rekayasa dalam desain perangkat keras, desain elektronik, desain perangkat lunak, dan diagram alir proses.

Block diagram biasanya digunakan untuk level yang lebih tinggi, deskripsi yang kurang rinci yang dimaksudkan untuk memperjelas konsep keseluruhan tanpa memperhatikan detail implementasi. Bandingkan ini dengan diagram skematik dan diagram tata letak yang digunakan dalam teknik kelistrikan, yang menunjukkan rincian implementasi komponen listrik dan konstruksi fisik (Wikipedia).

2.3.14 Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program". *Flowchart* menolong analis dan *programmer* untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengopersian (Sulindawati dan Fathoni, 2010).

Menurut Indrajani (2015), *Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Simbol-simbol dalam *Flowchart* ditunjukkan pada Tabel 2.1.

	Flow Direction symbol Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga connecting line.		Simbol Manual Input Simbol untuk pemasukan data secara manual on-line keyboard
	Terminator Symbol Yaitu simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan		Simbol Preparation Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam storage.
	Connector Symbol Yaitu simbol untuk keluar - masuk atau penyambungan proses dalam lembar / halaman yang sama.		Simbol Predefine Proses Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure
	Connector Symbol Yaitu simbol untuk keluar - masuk atau penyambungan proses pada lembar / halaman yang berbeda.		Simbol Display Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.
	Processing Symbol Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer		Simbol disk and On-line Storage Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.
	Simbol Manual Operation Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh computer		Simbol magnetik tape Unit Simbol yang menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
	Simbol Decision Simbol pemilihan proses berdasarkan kondisi yang ada.		Simbol Punch Card Simbol yang menyatakan bahwa input berasal dari kartu atau output ditulis ke kartu
	Simbol Input-Output Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya		Simbol Dokumen Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

Tabel 2.1. Simbol-simbol *Flowchart* (Indrajani, 2015).

2.3.15 Blackbox Testing

Blackbox testing adalah cara pengujian yang dilakukan dengan hanya menjalankan atau mengeksekusi unit atau model kemudian diamati apakah hasil dari unit itu sesuai dengan proses yang diinginkan (Arie, 2014).

Blackbox testing juga disebut pengujian tingkah laku, memusat pada kebutuhan fungsional perangkat lunak. Teknik pengujian *blackbox* memungkinkan memperoleh serangkaian kondisi masukan yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Beberapa jenis kesalahan yang dapat diidentifikasi adalah fungsi tidak benar atau hilang,

kesalahan antar muka, kesalahan pada struktur data (pengaksesan basis data), kesalahan performasi, kesalahan inisialisasi dan akhir program (Pressman, 2012).

2.3.16 Pengamanan Otomatis

Menurut Welman (2013), Pengaman otomatis dapat didefinisikan sebagai suatu alat yang mampu bekerja dengan sendirinya tanpa adanya campur tangan dari manusia (operator) dalam menanggulangi suatu permasalahan (problem). Maksud dari pengertian di atas adalah sebuah perangkat/alat yang bekerja secara sendiri sesuai dengan fungsinya, tanpa menunggu perintah dari luar atau adanya orang yang menjalankannya. Sedangkan sistem merupakan susunan/gabungan dari beberapa perangkat yang secara teratur dan saling berkaitan untuk membentuk satu kesatuan. Jadi sistem pengaman otomatis dapat disimpulkan sebagai susunan dari beberapa perangkat yang masing-masing memiliki fungsi yang berbeda namun saling berkaitan membentuk satu kesatuan dengan secara terus menerus (*continue*) memeriksa kondisi masukan yang mempengaruhi untuk kemudian melaksanakan pekerjaan sesuai dengan fungsinya dengan sendirinya. Beberapa sistem kecil dapat digabungkan menjadi sebuah sistem yang lebih besar dan kompleks.

Terdapat tiga elemen dasar yang menjadi syarat mutlak bagi sistem otomatis, yaitu power, *program of instruction*, dan sistem kontrol.

a. Power

Power atau bisa dikatakan sumber energi dari sistem otomatis berfungsi untuk menggerakkan semua komponen dari sistem otomatis. Sumber energi bisa menggunakan energi listrik, baterai, ataupun Accu, semuanya tergantung dari tipe sistem otomatis itu sendiri.

b. *Program of instruction*

Proses kerja dari sistem otomatis mutlak memerlukan sistem kontrol baik menggunakan proses mekanik, elektronik ataupun komputer. Program instruksi/perintah pada sistem kontrol mekanik maupun rangkaian elektronik tidak menggunakan bahasa pemrograman dalam arti sesungguhnya, karena sifatnya yang analog. Untuk sistem kontrol yang menggunakan komputer dan keluarganya (PLC maupun mikrokontroler) bahasa pemrograman merupakan hal yang wajib

ada. Hal ini dikarenakan perintah dalam PLC dan mikrokontroler dapat dimasukkan ke dalam perangkat melalui bahasa program yang sudah diatur sedemikian rupa dalam komputer sesuai keinginan programmer.

c. Sistem kontrol

Sistem kontrol merupakan bagian penting dalam sistem otomatis. Sistem kontrol juga dapat tersusun dari komputer, rangkaian elektronik sederhana, peralatan mekanik. Saat ini, lebih banyak penggunaan sistem kontrol dengan komputer dan keluarganya (PLC, mikrokontroler). Apabila suatu sistem otomatis dikatakan layaknya semua organ tubuh manusia seutuhnya maka sistem kontrol merupakan bagian otak ataupun pikiran, yang mengatur dari keseluruhan gerak tubuh.

2.3.17 Arduino IDE (*Integrated Development Environment*)

Sesungguhnya, Arduino tidak hanya berupa perangkat keras, melainkan juga menyatakan perangkat lunak. *Arduino Integrated Development Environment* (Arduino IDE) adalah nama perangkat lunak yang bersifat *open source* yang digunakan untuk membuat sketsa (istilah program di Arduino) hingga mengompilasi dan menggunakannya ke papan Arduino (Kadir, 2018).

Software arduino yang digunakan adalah driver dan IDE, walaupun masih ada beberapa software lain yang sangat berguna selama pengembangan arduino. *Integrated Development Environment* (IDE), suatu program khusus untuk suatu komputer agar dapat membuat suatu rancangan atau sketsa program untuk papan Arduino. IDE Arduino merupakan *software* yang sangat canggih ditulis dengan menggunakan java IDE Arduino terdiri dari :

1. *Editor Program*

Sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.

2. *Compiler*

Berfungsi untuk kompilasi *sketch* tanpa unggah ke *board* bisa dipakai untuk pengecekan kesalahan kode *sintaks sketch*. Sebuah modul yang mengubah kode program menjadi kode *biner* bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *processing*.

3. *Uploader*

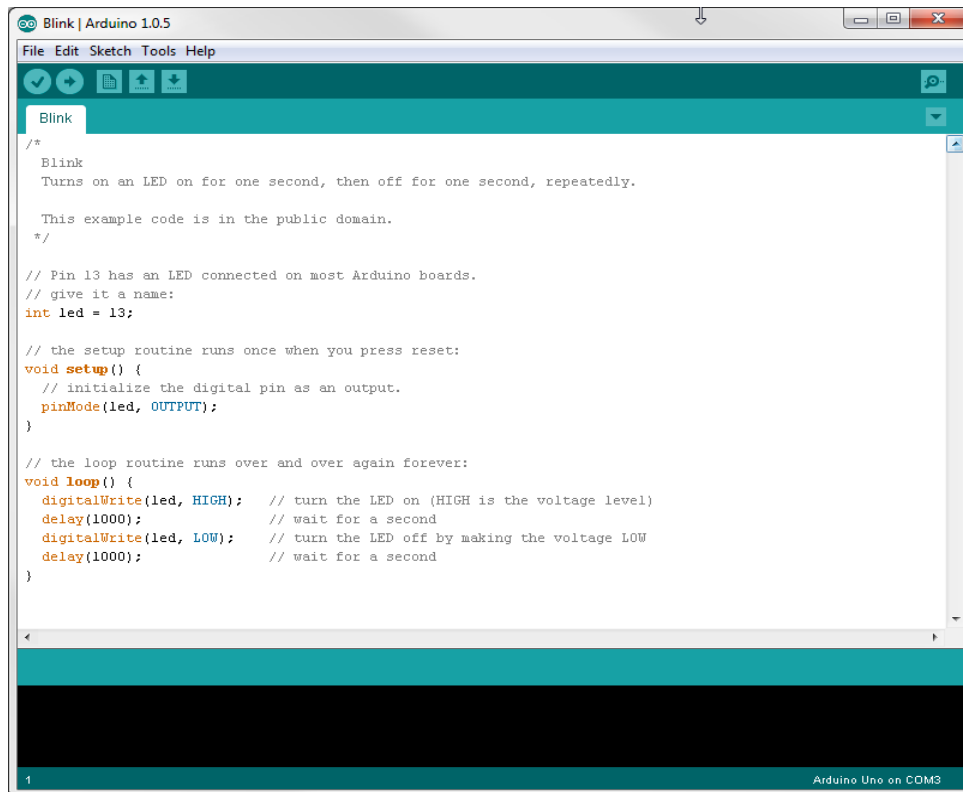
Berfungsi untuk mengunggah hasil kompilasi *sketch* ke *board* target. Pesan error akan terlihat jika *board* belum terpasang atau alamat *port* COM belum terkonfigurasi dengan benar. Sebuah modul yang memuat kode *biner* dari komputer ke dalam *memory* didalam papan Arduino (Gustomo, 2015).

2.3.18 Serial Monitor Arduino

Komunikasi serial Arduino adalah Komunikasi antara Arduino Uno dan komputer dapat dilakukan melalui port USB. Dalam hal ini, Arduino Uno tidak hanya bisa mengolah data dari pin I/O secara independ, tetapi dapat juga dikomunikasikan dengan komputer untuk ditampilkan hasil dari pengolahan datanya sehingga komunikasi yang dilakukan bersifat dua arah. Pada Arduino IDE terdapat fasilitas untuk berkomunikasi dua arah melalui serial monitor yang dapat dimanfaatkan dalam berbagai keperluan. Dengan menggunakan fasilitas ini, dapat dikirimkan data ke Arduino Uno dan sebaliknya dapat membaca kiriman dari arduino uno. Tentu saja, hal ini memungkinkan dapat mengontrol Arduino Uno melalui komputer dan memantau sesuatu yang sedang terjadi di Arduino Uno (Nyebarilmu.com)

2.3.19 Program Arduino IDE

Kode Program Arduino biasa disebut *sketch* dan dibuat menggunakan bahasa pemrograman C. Program atau *sketch* yang sudah selesai ditulis di Arduino IDE bisa langsung *dicompile* dan *diupload* ke *Arduino Board*. Secara sederhana, *sketch* dalam Arduino dikelompokkan menjadi 3 blok. Tampilan program arduino IDE dapat dilihat pada Gambar 2.4.



Gambar 2.4. Tampilan Program Arduino IDE (Ajie, 2016).

Penjelasan program Arduino IDE yang dikelompokkan menjadi 3 blok seperti yang ditunjukkan pada Gambar 2.4 diatas adalah:

a. *Header*

Pada bagian ini biasanya ditulis definisi-definisi penting yang akan digunakan selanjutnya dalam program, misalnya penggunaan *library* dan pendefinisian variable. *Code* dalam blok ini dijalankan hanya sekali pada waktu *compile*. Di bawah ini contoh *code* untuk mendeklarasikan *variable led* (integer) dan sekaligus di isi dengan angka 13, seperti yang ditunjukkan pada Gambar 2.5.

```

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;
  
```

Gambar 2.5. Tampilan Program *Header*

b. *Setup*

Di sinilah awal program Arduino berjalan, yaitu di saat awal, atau ketika *power on Arduino board*. Biasanya di blok ini diisi penentuan apakah suatu pin

digunakan sebagai *input* atau *output*, menggunakan perintah *pinMode*. Inisialisasi *variable* juga bisa dilakukan di blok ini, seperti yang ditunjukkan pada Gambar 2.6.

```
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}
```

Gambar 2.6. Tampilan Program *Setup*

OUTPUT adalah suatu makro yang sudah didefinisikan Arduino yang berarti = 1. Jadi perintah di atas sama dengan `pinMode(led, 1);`

Suatu pin bisa difungsikan sebagai *OUTPUT* atau *INPUT*. Jika difungsikan sebagai *output*, dia siap mengirimkan arus listrik (maksimum 100 mA) kepada beban yang disambungkannya. Jika difungsikan sebagai *INPUT*, pin tersebut memiliki *impedance* yang tinggi dan siap menerima arus yang dikirimkan kepadanya.

c. *Loop*

Blok ini akan dieksekusi secara terus menerus. Apabila program sudah sampai akhir blok, maka akan dilanjutkan dengan mengulang eksekusi dari awal blok. Program akan berhenti apabila tombol power Arduino di matikan. Di sinilah fungsi utama program Arduino berada, seperti yang ditunjukkan pada Gambar 2.7.

```
// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

Gambar 2.7. Tampilan Program *Loop*

Perintah `digitalWrite(pinNumber,nilai)` akan memerintahkan arduino untuk menyalakan atau mematikan tegangan di *pinNumber* tergantung nilainya. Jadi perintah di atas `digitalWrite(led,HIGH)` akan membuat pin nomor 13 (karena di *header* dideklarasikan `led = 13`) memiliki tegangan = 5V (*HIGH*). Hanya ada dua kemungkinan nilai `digitalWrite` yaitu *HIGH* atau *LOW* yang sebetulnya adalah nilai integer 1 atau 0. Kalau sudah dibuat program diatas, selanjutnya kita ambil

kabel USB yang diikutsertakan pada saat membeli Arduino, pasang ke komputer dan *board* Arduino, dan *upload* programnya. Lampu LED yg ada di Arduino *board* kita akan kelap-kelip. Sekedar informasi, sebuah LED telah disediakan di *board* Arduino Uno dan disambungkan ke pin 13. Selain blok *setup()* dan *loop()* di atas kita bisa mendefinisikan sendiri blok fungsi sesuai kebutuhan (Ajie, 2016).