

BAB II

LANDASAN TEORI

2.1 Kondisi Umum Lingkungan yang Menimbulkan Gagasan

Pada sistem absensi kehadiran yang masih menggunakan absensi secara manual yaitu dengan cara menuliskan nama, lalu mengisikan jam kedatangan dan membuat paraf dinilai masih kurang efektif. Banyak sekali faktor-faktor yang mempengaruhi penyebab dari permasalahan tersebut, salah satunya adalah tanda tangan yang berubah-ubah pada setiap harinya sehingga muncul dugaan adanya pemalsuan tanda tangan. Faktor lainnya disebabkan oleh hilangnya buku absensi akibat keteledoran yang membuat sulit rekapitulasi serta memberikan penilaian pada parameter kedisiplinan. Sehingga sistem absensi secara manual ini kurang efektif dan efisien karena terdapat banyak kekurangan mulai dari proses menuliskan absensi, menyimpan data absensi, serta memantau data kehadiran.

Berawal dari permasalahan tersebut, timbul sebuah gagasan untuk membuat sebuah sistem absensi kehadiran dengan menggunakan mikrokontroler sebagai *interface* dengan memanfaatkan teknologi biometrik sidik jari. Output dari absensi kehadiran akan ditampilkan pada sebuah layar LCD. Pembuatan alat ini tidak memerlukan biaya yang cukup mahal, sehingga untuk implementasinya masih dapat dijangkau.

Sistem absensi kehadiran menggunakan sensor *optical fingerprint* mampu melakukan koneksi secara *real time* dengan adanya modul RTC DS1307. Sensor *optical fingerprint* mampu membaca data sidik jari pada setiap individu untuk mempermudah proses identifikasi absensi kehadiran sehingga proses absensi dapat dilakukan lebih akurat, aman, dan nyaman untuk digunakan.

2.2 Teori Sidik Jari

Menurut Komarinski (2005:3), *Fingerprint* atau sidik jari adalah sebuah biometric yang telah digunakan secara sistematis untuk identifikasi selama 100 tahun yang telah diukur, diduplikasi dan diperiksa secara ekstensif, sebuah biometric yang tidak berubah dan relative mudah untuk diambil. Sidik Jari

merupakan identitas pribadi yang tidak mungkin ada yang menyamainya. Sifat-sifat atau karakteristik yang dimiliki oleh sidik jari adalah *perennial nature* yaitu guratan-guratan pada sidik jari yang melekat pada manusia seumur hidup, *immutability* yang berarti bahwa sidik jari seseorang tak akan pernah berubah kecuali sebuah kondisi yaitu terjadi kecelakaan yang serius sehingga mengubah pola sidik jari yang ada dan *individuality* yang berarti keunikan sidik jari merupakan originalitas pemiliknya yang tak mungkin sama dengan siapapun di muka bumi ini sekali pun pada seorang yang kembar identik.

Sidik jari adalah gurat-gurat yang terdapat di kulit ujung jari. Sidik jari berfungsi untuk memberi gaya gesek lebih besar agar jari dapat memegang benda-benda lebih erat. Sistem pengamanan dengan menggunakan sidik jari sudah mulai dipergunakan di Amerika oleh seorang bernama E. Henry pada tahun 1902. Henry menggunakan metode sidik jari untuk melakukan identifikasi pekerja dalam rangka mengatasi pemberian upah ganda. Sistem Henry menggunakan pola *ridge*, yang terpusat pada pola jari tangan, khususnya telunjuk. Untuk memperoleh gambar pola *ridge*, dilakukan dengan cara menggulung jari yang diberi tinta pada suatu kartu cetakan hingga dihasilkan suatu pola *ridge* yang unik bagi masing-masing individu. Para pakar membuktikan bahwa tidak ada dua individu yang mempunyai pola *ridge* yang serupa. Pola *ridge* dibentuk waktu embrio, dan tidak pernah berubah seumur hidup. Perubahan *ridge* hanya dapat terjadi akibat trauma, misalnya akibat luka-luka, terbakar, penyakit, atau penyebab lainnya. Sistem biometrika sidik jari merupakan sistem yang paling banyak digunakan saat ini karena memiliki tingkat akurasi yang tinggi dan mudah untuk diterapkan.

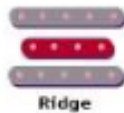


Dari hasil penelitian ditemukan 9 macam pola utama *papillary ridge*, yaitu :

1. *Loop* : Terdiri dari satu atau lebih kurva bebas dari *ridge* dan sebuah delta.
2. *Arch* : Membentuk pola dengan *ridge* berada diatas *ridge* yang lain dalam bentuk lengkungan umum.
3. *Whorl* : Pola ini terdiri dari satu atau lebih kurva bebas *ridge* dan dua buah delta.
4. *Tented Arch* : Pola ini terdiri dari paling tidak sebuah *ridge* yang melengkung keatas yang kemudian bercabang menjadi dua *ridge*.


5. *Double Loop*: Pola ini membentuk dua formasi lengkungan yang lalu berpisah, dengan dua titik delta.
6. *Central Pocket Loop* : Terdiri dari satu atau lebih kurva *ridge* dan dua titik delta.
7. *Accidental* : Pola ini mempunyai dua titik delta. Satu delta akan berhubungan dengan lengkungan keatas, dan delta yang lain terhubung dengan lengkungan yang lain.
8. *Composite* : Terdiri dari gabungan dua atau lebih pola yang berbeda.
9. *Lateral Pocket Loop* : Pola ini terdiri dari dua lengkungan yang terpisah.

Sekitar 60% orang memiliki pola sidik jari *loop*. Sekitar 30% orang memiliki pola *whorl*, sekitar 5% berbentuk *arch*, dan 5% sisanya adalah bentuk-bentuk lainnya. Semua pola tersebut dapat dibedakan oleh mata biasa. Komputer dapat menganalisa garis-garis perubahan arah bentuk *ridge*, dengan kemampuan seperti mata manusia yang terlatih. Gambaran ukuran-ukuran karakteristik anatomi pola tersebut dapat digambarkan sebagai berikut :

Tabel 2.1 Variasi Pola Ridge

NO	GAMBAR	NAMA POLA	KETERANGAN
1.		<i>Ridge</i>	Mempunyai ketegasan jarak ganda dari permulaan ke-akhir, sebagai lebar ridgessatu dengan lainnya
2.		<i>Evading Ends</i>	Dua <i>ridge</i> dengan arah berbeda berjalan sejajar satu sama lain kurang dari 3mm.
3.		<i>Bifurcation</i>	Dua <i>ridge</i> dengan arah berbeda berjalan sejajar satu sama lain kurang dari 3mm.

4.	 Hook	Hook	<i>Ridges</i> merobek; satu <i>ridges</i> tidaklah lebih panjang dibanding 3mm
5.	 Fork	Fork	Dua <i>ridges</i> dihubungkan oleh seper tiga <i>ridges</i> tidak lebih panjang dibanding 3mm
6.	 Dot	Dot	Bagian <i>ridges</i> tidak lagi dibanding <i>ridges</i> yang berdekatan
7.	 Eye/Island	Eye	<i>Ridges</i> merobek dan menggabungkan lagi di dalam 3mm
8.	 Eye/Island	Island	<i>Ridges</i> merobek dan tidak bergabung lagi, kurang dari 3mm dan tidak lebih dari 6mm. Area yang terlampir adalah <i>Ridge</i> .
9.	 Enclosed Ridge	Enclosed Ridge	<i>Ridges</i> tidak lebih panjang dibanding 6mm antara dua <i>ridges</i> .
10.	 Enclosed Loop	Enclosed Loop	<i>Ridges</i> yang tidak mempola dan menentukan pengulangan antar dua atau lebih <i>ridges</i> paralel

11.		<i>Specialties</i>	Rare <i>ridge</i> membentuk seperti tanda tanya dan sangkutan pemotong
-----	---	--------------------	--

Area *papillary ridge* kadang-kadang dikenal sebagai *pattern area*. Masing-masing pola *papillary ridge* menghasilkan suatu bentuk pola area yang berbeda. Pusat gambar jari mencerminkan pola area, dikenal sebagai inti *core point*. Bagian *ridges* yang berwujud dua paralel yang berbeda mengelilingi pola area inti disebut *type lines*. Titik awal percabangan dua *ridge* disebut *delta*. Proses perpecahan sebuah garis menjadi dua garis *ridge* disebut *bifurcation*. Banyaknya persimpangan *ridge* di dalam pola area disebut suatu *ridge count*. Komputer *Tormography* dapat digunakan untuk mendeteksi titik-titik tersebut berdasarkan sumbu koordinat x-y. (Eko Nugroho, 2009).

2.3 Sensor Sidik Jari

Sistem biometrik sidik jari merupakan sistem yang paling banyak digunakan saat ini, karena memiliki kecenderungan tingkat akurasi yang tinggi dan mudah diterapkan. Sifat yang dimiliki sidik jari antara lain :

1. *Perennial nature*, yaitu guratan-guratan pada sidik jari yang melekat pada kulit manusia seumur hidup.
2. *Immutability*, yaitu sidik jari seseorang tidak pernah berubah, kecuali mendapatkan kecelakaan yang serius.
3. *Individuality*, pola sidik jari adalah unik dan berbeda untuk setiap.

Ciri khas sidik jari yang digunakan adalah sidik jari yang diidentifikasi dengan cara menganalisis detail dari guratan-guratan sidik jari yang dinamakan "*minutiae*" (Naslim Lathif, 2001). *Minutiae* berasal dari bahasa Inggris yang artinya barang tidak berarti atau rincian tidak penting dan terkadang diartikan sebagai detail. *Minutiae* sebenarnya merupakan rincian sidik jari yang tidak

penting bagi kita, tetapi bagi sebuah mesin sidik jari itu adalah detil yang diperhatikan.



Gambar 2.1 Definisi Sidik Jari
(A. Prawira. 2017)

Pemindai sidik jari saat ini sudah banyak digunakan, mulai dari absensi, sebagai *access control*, hingga sebagai identitas pribadi seperti pada SIM atau *passport*. Seperti halnya bagian tubuh yang lain, sidik jari terbentuk karena faktor genetik dan lingkungan. Kode genetik pada DNA memberi perintah untuk terbentuknya janin yang secara spesifik membentuk hasil secara acak. Demikian juga halnya dengan sidik jari. Sidik jari memiliki bentuk unik bagi setiap orang.

Artinya setiap orang memiliki bentuk sidik jari yang berbeda-beda meskipun terlahir kembar. Jadi, walaupun sidik jari terlihat seperti sama bila dilihat sekilas, buat penyidik terlatih atau dengan menggunakan software khusus akan terlihat perbedaannya.

2.3.1 Sensor *Optical Fingerprint AS608*

Sensor *optical* adalah adanya CCD (*Charge Couple Device*) yang cara kerjanya sama seperti sistem sensor yang terdapat pada kamera digital atau *camcorder*. CCD merupakan chip silikon yang terbentuk dari ribuan bahkan jutaan dioda fotosensitif yang disebut *photosites*, *photodelements*, atau disebut juga piksel. Tiap *photosite* menangkap satu titik objek, kemudian dirangkai dengan hasil tangkapan *photosite* lain menjadi satu gambar. Bila mengambil contoh pada kamera, saat menekan tombol '*capture*' pada kamera digital, sel pengukur intensitas cahaya akan menerima dan merekam setiap cahaya yang masuk menurut intensitasnya. Dalam waktu yang sangat singkat, tiap titik *photosite* akan merekam cahaya yang diterima dan diakumulasikan dalam sinyal

elektronis. Gambar yang sudah dikalkulasikan dalam gambar yang sudah direkam dalam bentuk sinyal elektronik akan dikalkulasi untuk kemudian disimpan dalam bentuk angka-angka digital. Angka tersebut akan digunakan untuk menyusun ulang gambar untuk ditampilkan kembali. Perekaman gambar yang dilakukan oleh CCD sebenarnya dalam format *grayscale* atau *monochrome* dengan 256 macam intensitas warna dari putih sampai hitam.

Sensor tersebut terhubung dengan sebuah integrator yang dilengkapi dengan inverter penguat yang dapat menerjemahkan, sehingga pada akhirnya akan membentuk sidik jari yang sedang dipindai. Setelah mesin pemindai menyimpan image atau gambar yang diambil, mesin lalu melakukan '*searching minutiae*' atau mencari titik-titik *minutiae*. Lalu mesin pemindai akan mencari kecocokan pola pada *minutiae-minutiae* yang telah terkumpul tersebut.

Jika mesin pemindai sidik jari mendapatkan pola sidik jari yang sama, maka proses identifikasi sudah berhasil. Tidak semua *minutiae* harus digunakan, dan pola yang ditemukan tidak harus sama. Maka dapat disimpulkan bahwa posisi jari kita pada saat identifikasi tidak harus sama persis dengan pada saat pertama kali menyimpan data sidik jari pada mesin tersebut. Pemindai sidik jari *optical* menghasilkan tingkat keamanan yang tinggi, karena tidak bisa dipalsukan dengan fotocopy sidik jari, sidik jari tiruan, atau bahkan dengan cetak lilin yang mendetil dengan guratan-guratan kontur sidik jari sekalipun. (Alpha, 2012).

AS608 adalah salah satu tipe sensor sidik jari dengan verifikasi sangat sederhana. Modul sensor *optical fingerprint* AS608 memiliki chip DSP bertenaga tinggi untuk melakukan rendering gambar, pencarian fitur dan pencarian sidik jari yang tersimpan. Modul ini menggunakan komunikasi serial TTL (*transistor transistor logic*) untuk menerima dan mengirim data untuk mengambil foto, mendeteksi cetakan dan pencarian. Modul sensor ini dapat menyimpan 127 sidik jari yang disimpan dalam memori *flash onboard*. Dalam modul ini juga terdapat LED (*light emitting diode*) biru di lensa yang akan menyala selama sensor tersebut bekerja.



Gambar 2.2 Sensor *Optical Fingerprint* AS608
(Dhgate. 2020)

Tabel 2.2 Spesifikasi Sensor *Optical Fingerprint* AS608

Nama	Spesifikasi
Tegangan suplai	3,3 – 6.0 VDC
Arus operasi	100 mA maks
Arus puncak	150 mA maks
Waktu pencitraan sidik jari	<0,5 detik
Area Jendela	14mm x 18mm
File sidik jari	128 <i>bytes</i>
File template	512 <i>bytes</i>
Kapasitas Penyimpanan	127 <i>Template</i>
Peringkat keamanan	1-5 level rendah hingga tinggi
Tingkat penolakan salah	<1.0%
Antarmuka	TTL Serial

Fungsi sensor *fingerprint* digunakan untuk media verifikasi, sama halnya seperti *username* dan *password*. Sensor *optical fingerprint* membutuhkan modul mikrokontroler yang bertanggung jawab untuk melakukan komunikasi (mengirim gambar, menerima perintah, dan sebagainya) dengan dibantu oleh output berupa layar (menampilkan teks). Sensor *optical fingerprint* AS608 membutuhkan *library* agar dapat terkoneksi dengan mikrokontroler, apabila *library* pada program tidak ditambahkan maka sensor *optical fingerprint* AS608 tidak dapat diprogram.

2.4 Mikrokontroler

Mikrokontroler adalah sebuah sistem komputer fungsional dalam sebuah chip yang di dalamnya terdapat sebuah inti prosesor, memori (sejumlah kecil RAM, memori program, atau keduanya), dan perlengkapan input output. Dengan kata lain, mikrokontroler merupakan komputer didalam chip yang digunakan untuk mengontrol peralatan elektronik, yang menekankan efisiensi dan efektifitas biaya. Secara harfiahnya bisa disebut “pengendali kecil” dimana sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti *IC TTL* dan *CMOS* dapat direduksi/ diperkecil dan akhirnya terpusat serta dikendalikan oleh mikrokontroler ini.

Mikrokontroler digunakan dalam produk dan alat yang dikendalikan secara otomatis, seperti sistem kontrol mesin, *remote control*, mesin kantor, peralatan rumah tangga, alat berat, dan mainan. Dengan mengurangi ukuran, biaya, dan konsumsi tenaga dibandingkan dengan mendesain menggunakan mikroprosesor memori, dan alat input output yang terpisah, kehadiran mikrokontroler membuat kontrol elektrik untuk berbagai proses menjadi lebih ekonomis. Dengan penggunaan mikrokontroler ini maka :

1. Sistem elektronik akan menjadi lebih ringkas
2. Rancang bangun sistem elektronik akan lebih cepat karena sebagian besar dari sistem adalah perangkat lunak yang mudah dimodifikasi
3. Pencarian gangguan lebih mudah ditelusuri karena sistemnya yang kompak.

Agar sebuah mikrokontroler dapat berfungsi, maka mikrokontroler tersebut memerlukan komponen eksternal yang kemudian disebut dengan sistem minimum. Untuk membuat sistem minimal paling tidak dibutuhkan sistem *clock* dan reset, walaupun pada beberapa mikrokontroler sudah menyediakan sistem *clock* internal, sehingga tanpa rangkaian eksternal pun mikrokontroler sudah beroperasi.

Mikrokontroler tersusun dalam satu chip dimana prosesor, memori, dan I/O terintegrasi menjadi satu kesatuan kontrol sistem sehingga mikrokontroler dapat dikatakan sebagai komputer mini yang dapat bekerja secara inovatif sesuai dengan kebutuhan sistem. Sistem *running* bersifat berdiri sendiri tanpa tergantung dengan

komputer sedangkan parameter komputer hanya digunakan untuk *download* perintah instruksi atau program. Langkah-langkah untuk *download* komputer dengan mikrokontroler sangat mudah digunakan karena tidak menggunakan banyak perintah. Pada mikrokontroler tersedia fasilitas tambahan untuk pengembangan memori dan I/O yang disesuaikan dengan kebutuhan sistem. (elektronika dasar, 2010)



Gambar 2.3 Chip Mikrokontroler ESP32
(Nodemcu. 2020)

2.4.1 NodeMCU ESP32

NODEMCU adalah sebuah *platform IoT* yang bersifat *opensource*. Terdiri dari perangkat keras berupa *system on chip* ESP32, juga *firmware* yang menggunakan bahasa pemrograman *scripting* Lua. Istilah NODEMCU secara *default* sebenarnya mengacu pada *firmware* yang digunakan pada perangkat keras *development kit*. Tampilan secara fisik dari NodeMCU ESP32 dapat dilihat pada gambar 2.3 berikut ini :



Gambar 2.4 Mikrokontroler NodeMCU ESP32
(Nodemcu. 2020)

NodeMCU ESP32 merupakan penerus dari ESP8266 yang memiliki banyak fitur tambahan dan keunggulan dibandingkan generasi sebelumnya. Pada ESP32

terdapat inti CPU serta *Wi-Fi* yang lebih cepat, GPIO yang lebih banyak, dan dukungan terhadap *Bluetooth* 4.2, serta konsumsi daya yang rendah. Berikut spesifikasi dari ESP32 dapat dilihat pada table 2.3

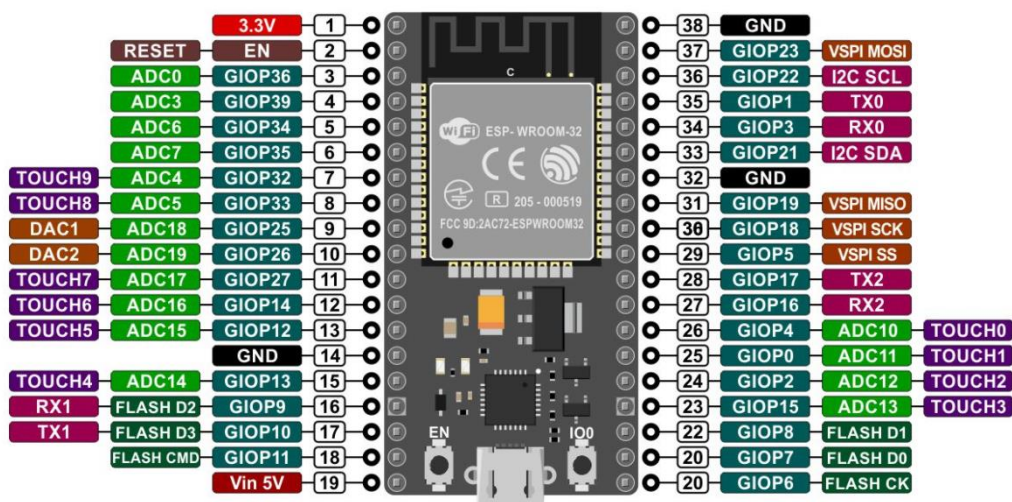
Tabel 2.3 Spesifikasi NodeMCU ESP32

Spesifikasi	Fitur
<i>MCU</i>	Xtensa Dual-Core 32bit LX6 600DMIPS
802.11 b/g/n <i>Wi-Fi</i>	HT40
<i>Bluetooth</i>	<i>Bluetooth</i> 4.2
<i>Typical Frequency</i>	160Mhz
<i>SRAM</i>	512 <i>kBytes</i>
<i>Flash</i>	SPI
<i>GPIO</i>	36
<i>Hardware/ Software PWM</i>	1/ 16 <i>Channels</i>
SPI/ I2C/ I2S/ UART	4/2/2/2
<i>ADC</i>	12 Bit
<i>CAN</i>	1
<i>Ethernet MAC Interface</i>	1
<i>Touch Sensor</i>	<i>Yes</i>
<i>Temperature Sensor</i>	<i>Yes</i>
<i>Working Temperature</i>	-40°C – 125°C
<i>Current GPIO</i>	12mA

Pada ESP32, tidak semua pin dapat diakses oleh *development board*-nya, dan terdapat 38 pin GPIO yang bisa difungsikan sebagai berikut :

1. *Analog to Digital Converter* (ADC) : 18 kanal SAR ADC 12 bit. Rentang ADC bisa diatur di dalam program, apakah 0-1 V, 0-1.4 V, 0-2V atau 0-4V.
2. *Digital to Analog Converter* (DAC) : terdapat DAC 8 bit yang bisa menghasilkan tegangan analog.

3. *Pulse Width Modulation* (PWM) : 16 kanal PWM yang bisa digunakan untuk mengendalikan LED atau motor.
4. *Touch Sensor* : 10 GPIO memiliki kemampuan pengindera kapasitif yang dapat digunakan sebagai 10 tombol buttonpad.
5. *UART* : 3 kanal antarmuka UART. Satu diantaranya digunakan untuk mendownload program secara serial.
6. I2C, SPI, I2S : Terdapat dua antarmuka I2C dan 3 antarmuka SPI untuk mengakses sensor dan perangkat ditambah lagi 2 antarmuka I2S.
7. RST : berfungsi mereset modul.
8. EN: *Chip Enable*, Aktif tinggi.
9. VCC: Catu daya 3.3V (VDD)
10. CS0 : *Chip selection*
11. MISO : *Main input Slave output*
12. GND: Ground



Pin yang diberi *highlight* hijau, bisa digunakan di dalam project. Sedangkan pin dengan *highlight* kuning bisa digunakan namun dengan catatan yang perlu diperhatikan, karena terdapat perilaku yang tak terduga terutama saat proses boot. Pin dengan *highlight* merah tidak direkomendasikan sebagai input ataupun output. Pin input & output ESP32 dapat dilihat pada tabel 2.4 dibawah ini :

Tabel 2.4 Pin Input & Output ESP32

GPIO	Input	Output	Catatan
0	pulled up	OK	output sinyal PWM saat boot
1	TX pin	OK	output debug saat boot
2	OK	OK	Terhubung ke LED on board
3	OK	TX pin	HIGH saat boot
4	OK	OK	
5	OK	OK	output sinyal PWM saat boot
6	x	x	terhubung dengan SPI Flash terintegrasi
7	x	x	terhubung dengan SPI Flash terintegrasi
8	x	x	terhubung dengan SPI Flash terintegrasi
9	x	x	terhubung dengan SPI Flash terintegrasi
10	x	x	terhubung dengan SPI Flash terintegrasi
11	x	x	terhubung dengan SPI Flash terintegrasi
12	OK	OK	boot gagal ketika mendapatkan input high
13	OK	OK	
14	OK	OK	output sinyal PWM saat boot
15	OK	OK	output sinyal PWM saat boot
16	OK	OK	
17	OK	OK	
18	OK	OK	
19	OK	OK	
20	OK	OK	

21	OK	OK	
22	OK	OK	
23	OK	OK	
24	OK	OK	
25	OK	OK	
26	OK	OK	
27	OK	OK	
28	OK	OK	
29	OK	OK	
30	OK	OK	
31	OK	OK	
32	OK	OK	
33	OK	OK	
34	OK	OK	
35	OK	OK	
36	OK		Hanya input
37	OK		Hanya input
38	OK		Hanya input
39	OK		Hanya input

2.4.2 Memori

Terdapat tiga jenis memori yang terdapat pada NodeMCU ESP32 yaitu :

1. *Flash Memory*, memori yang digunakan untuk menyimpan sketch/ program NodeMCU ESP32. *Flash Memory* adalah media penyimpanan yang berjenis “*non-volatile*” yang berarti tidak memerlukan power untuk menjaga keberadaan data. *Flash Memory* hampir sama dengan EEPROM (*Electrically Erasable Programmable ROM*). Kapasitas memorinya pun beragam, mempunyai kemampuan transfer data untuk penulisan mencapai 88 Mbps sedangkan untuk pembacaan mencapai 5 Mbps. Para ilmuwan membuatnya menjadi sistem penyimpanan data portabel, mirip disket, maka sering disebut *Flash Disk*.

2. SRAM (*Static Random Access Memory*), memori yang digunakan untuk menyimpan data variabel sementara. Memori SRAM (*Static Random Access Memory*) adalah tipe memori yang digunakan untuk menyimpan data. Berbeda dengan ROM yang menyimpan program, memori bertipe RAM ini digunakan untuk menyimpan data. Data dalam memori ini akan hilang ketika daya ke mikrokontroler ditiadakan (*volatile*). Data disini misalnya saat kita mendeklarasikan variabel tertentu atau array, atau data hasil penjumlahan dan pengurangan, dan sebagainya. Oleh karena itu, efektivitas dalam pemrograman sebuah mikrokontroler menjadi tantangan tersendiri. Kita tidak boleh mendeklarasikan variabel sebanyak-banyaknya tanpa memperhatikan pertimbangan memori yang tersedia. Sebagai contoh, jika kita ingin membuat program *logger* yang melibatkan banyak data, kita disarankan menggunakan memori eksternal, misalnya SD card.
3. EEPROM (*Electrically Erasable Programmable Read Only Memory*), memori yang menyimpan data variabel dalam jangka waktu yang lama. EEPROM merupakan salah satu jenis memori yang memiliki alamat (*address*) yang didalam terdapat data (*value*). Pada mikrokontroler NodeMCU ESP32 yang digunakan memiliki memori EEPROM sebesar 512 *kBytes*. EEPROM memiliki alamat sebanyak 1024 atau mulai dari 0 – 1023 dimana setiap alamat memiliki data sebesar 8 bit atau bernilai 0 – 255. Memori EEPROM tidak terhapus walaupun tanpa dialiri listrik. Analoginya mirip seperti *harddisk drive* atau *flash disk*. Data yang disimpan tidak akan terhapus walau tanpa dialiri listrik. Berbeda dengan RAM, tanpa dialiri listrik, Mikrokontroler akan padam dan memori kembali menjadi kosong. EEPROM ini memiliki fungsi yang sangat banyak terutama pada sistem absensi kehadiran, karena salah satu fungsinya adalah untuk menyimpan data kehadiran dari variabel id yang telah diinputkan pada sensor *optical fingerprint*. Dengan sifat EEPROM yaitu *read*, *write*, dan *idle* maka kegunaan EEPROM sendiri begitu sangat berguna dalam memanfaatkan sistem memori.

2.4.3 Komunikasi

NodeMCU ESP32 telah melengkapi komunikasi serial dengan *port library* yang memudahkan untuk memprogram, yaitu :

1. *Serial Available*, digunakan untuk menyatakan angka, *bytes* atau karakter yang sudah siap dibaca dari *serial port*. Data ini adalah data yang telah diterima dan disimpan dalam serial *receive buffer*. *Serial receive buffer* dapat menampung 64 *bytes* data.
2. *Serial Begin*, digunakan untuk mengatur *baudrate/* kecepatan transmisi data. Beberapa pilihan kecepatan komunikasi data yang dapat digunakan pada board NodeMCU ESP32 adalah 300, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600 atau 115200. Pengaturan *baudrate* dilakukan pada bagian *setup*.
3. *Serial End*, digunakan untuk menutup komunikasi *serial port*.
4. *Serial Find*, digunakan untuk membaca data dari *serial port buffer* hingga target yang ditentukan dalam perintah.
5. *Serial Print*, digunakan untuk menampilkan data ke serial monitor. Data yang ditampilkan dapat berupa karakter, *bytes*, atau angka.
6. *Serial Read*, digunakan untuk membaca data dari *serial port*.
7. *Serial Write*, digunakan untuk membaca data biner dari *serial port*. Data ini dikirim dalam bentuk *byte* atau deretan data *byte*.

2.5 Teori Dasar I2C

InterIntegrated Circuit atau sering disebut I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang didesain khusus untuk pengontrolan IC. System I2C terdiri dari saluran SCL (*Serial Clock*) dan SDA (*Serial Data*) yang membawa informasi data antara I2C dengan pengontrol. (Purnomo, 2011)

I2C memiliki sistem pengantar yang dilengkapi dengan komponen pengendali untuk melayani pertukaran data antara komponen *hardware* satu dengan komponen *hardware* lainnya. Pada sistem mikrokontroler terdapat bus data, bus alamat, dan beberapa pengantar pengendali. Semakin tinggi frekuensi klok prosesor, maka semakin lebih cermat pengembang untuk memperhatikan Timing dari seluruh komponen yang terlibat, agar tidak terjadi kesalahan dalam transaksi

data. Bus yang cukup sering digunakan adalah bus bersifat paralel. Transaksi data dilakukan secara paralel sehingga transaksi data lebih cepat. Namun disisi lain harganya cukup mahal. Jika sistem relatif tidak membutuhkan transaksi yang cepat, maka penggunaan serial bus menjadi pilihan. Salah satu pilihan sistem data bus yang sering digunakan adalah I2C (*Inter Integrated Circuit*). Sistem Bus I2C pertamakali diperkenalkan oleh Firma Philips pada tahun 1979.

Karakteristik dari komunikasi I2C adalah :

1. *Serial Bus*, data dikirim serial secara per-bit.
2. Menggunakan dua penghantar koneksi dengan ground bersama I2C terdiri dari dua penghantar koneksi yaitu : SDA (*Serial data*) yang berfungsi untuk mentransmisikan data dan SCL (*Serial Clock Line*) untuk menghantarkan sinyal *clock*.
3. Jumlah data bus maksimal 127.
4. Setiap transaksi data terjadi antara pengirim (*transmitter*) dan penerima (*receiver*).
5. Device yang mengendalikan operasi transfer disebut *master*, sementara device yang dikendalikan oleh *master* disebut *slave*.

I2C juga memiliki syarat dan aturan khusus dalam proses kendalinya yaitu :

1. I2C adalah protokol transfer data serial. Device atau komponen yang mengirim data disebut *transmitter*, sedangkan perangkat yang menerimanya disebut *receiver*.
2. *Device* yang mengendalikan operasi transfer data disebut *master*, sedangkan device lainnya yang dikendalikan oleh master disebut *slave*.
3. *Master* harus menghasilkan serial clock melalui pin SCL, mengendalikan akses ke BUS serial dan menghasilkan sinyal kendali *start* dan *stop*.

2.5.1 I2C LCD

Menurut Abdul Kadir (2017:111), *Inter-Integrated Circuit* (I2C) LCD adalah jenis LCD yang menggunakan komunikasi I2C untuk berhubungan dengan arduino dan mikrokontroler lainnya. LCD yang digunakan juga tidak berbeda pada dengan LCD jenis paralel.

Pada pin modul I2C LCD, terdapat 4 pin yang biasa digunakan yaitu :

1. VCC (5V)
2. GND
3. SDA (*Serial Data*)
4. SCL (*Serial Clock*)

Pada gambar 2.5 adalah gambar dari modul I2C yang sering digunakan, salah satunya adalah untuk komunikasi I2C pada LCD.



Gambar 2.6 Modul I2C LCD
(Galilelo. 2020)

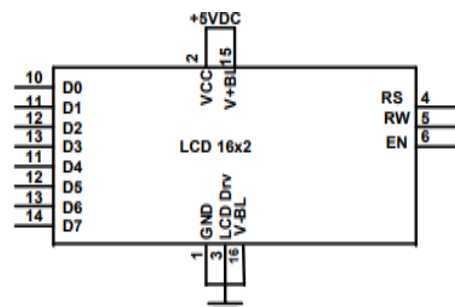
2.6 LCD (*Liquid Crystal Display*)

Menurut Abdul Kadir (2017:110), *Liquid Crystal Display* adalah komponen yang digunakan untuk menampilkan informasi dalam bentuk layar sederhana.

LCD adalah salah satu jenis tampilan elektronik yang dibuat dengan teknologi *CMOS logic* yang bekerja memantulkan cahaya yang terdapat di sekelilingnya terhadap *front-lit* dan *back-lit*. LCD banyak sekali digunakan dalam merancang suatu sistem dengan menggunakan mikrokontroler. *Liquid Crystal Display* ini juga berfungsi untuk menampilkan suatu teks, atau menampilkan menu pada aplikasi mikrokontroler. *Liquid Crystal Display* yang digunakan adalah *Liquid Crystal Display* 16x4, artinya LCD terdiri dari 4 baris dan 16 karakter dengan 16 pin konektor.



(a)



(b)

Gambar 2.7 a. Tampilan LCD 16x4 b. Konfigurasi Pin LCD 16x4
(N. Baity Sitorus. 2017)

Pada LCD terdapat pin konfigurasi, dan deskripsi dari pin-pin LCD yaitu :

1. VSS (Pin 1) : merupakan *power supply* (GND).
2. VCC (Pin2) : merupakan *power supply* (+5V).
3. VEE (Pin 3) : merupakan input tegangan kontras LCD.
4. RS *Register Select* (Pin 4) : merupakan register pilihan 0 = Register Perintah, 1 = Register Data.
5. R/W (Pin 5) : merupakan *read select*, 1 = *Read*, 0 = *Write*.
6. *Enable Clock* LCD (Pin 6) : merupakan masukan logika 1 setiap kali pengiriman atau pembacaan data.
7. D0 sampai D7 (Pin 7 sampai Pin 14) : merupakan data bus 1 sampai 7.

Dalam modul LCD (*Liquid Crystal Display*) terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter. Mikrokontroler tersebut dilengkapi dengan register dan memori. Memori yang digunakan pada mikrokontroler LCD adalah :

1. *Display Data Random Acces Memory* atau DDRAM merupakan memori tempat karakter untuk ditampilkan.
2. *Character Generator Random Acces Memory* atau CGRAM merupakan memori untuk menggambarkan sebuah pola karakter yang dimana dari karakter dapat diubah sesuai dengan keinginan.
3. *Character Generator Read Only Memory* atau CGROM merupakan memori untuk menggambarkan sebuah pola karakter yang dimana pola tersebut adalah karakter dasar yang sudah ditentukan oleh pabrikan pembuat LCD, sehingga pengguna hanya mengambilnya sesuai dengan alamat memorinya dan tidak dapat merubah karakter dasar didalam CGROM.

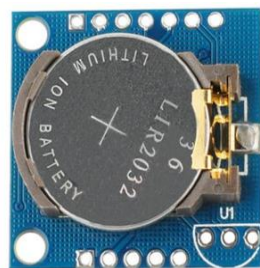
Register yang ada pada LCD diantaranya adalah :

1. Register perintah adalah register yang berisikan perintah-perintah dari mikrokontroler ke LCD pada saat proses penulisan data atau tempat status dari LCD dapat dibaca pada pembacaan data.
2. Register data yaitu register yang digunakan untuk menuliskan atau membaca data dari LCD ke DDRAM. Register data akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur.

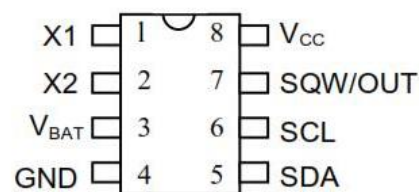
2.7 RTC DS1307

Menurut Inggit (2019:18), RTC adalah salah satu jenis modul yang berfungsi sebagai *Real Time Clock* atau pewaktuan digital serta penambahan fitur pengukur suhu yang dikemas kedalam 1 IC.

Selain itu pada modul terdapat IC EEPROM tipe AT24C32 yang dapat dimanfaatkan. *Interface* atau antarmuka untuk mengakses modul ini yaitu menggunakan I2C atau *two wire* (SDA dan SCL). Sehingga apabila diakses menggunakan mikrontroler misal NodeMCU ESP32 pin yang dibutuhkan 2 pin saja dan 2 pin *power*. Modul DS1307 RTC ini pada umumnya sudah tersedia dengan battery CR2032 3V yang berfungsi sebagai *backup* RTC apabila catudaya utama mati. Dibandingkan dengan RTC DS1302, DS1307 RTC ini memiliki banyak kelebihan. Sebagai contoh untuk *range* VCC input dapat di *supply* menggunakan tegangan antara 2.3V sampai 5.5V dan memiliki cadangan baterai. Berbeda dengan DS3231, pada DS1307 juga memiliki kristal terintegrasi sehingga tidak diperlukan kristal eksternal, dan juga terdapat sensor suhu, 2 alarm waktu terprogram, pin output 32.768 kHz untuk memastikan akurasi yang lebih tinggi. Selain itu, terdapat juga EEPROM AT24C32 yang bisa digunakan untuk menyimpan data, RTC ini mempunyai fitur data *logging*, dengan presisi waktu yang lebih tinggi.



(a)



DS1307 8-Pin DIP (300-mil)

(b)

Gambar 2.8 a. Tampilan RTC DS1307 b. Konfigurasi Pin IC DS1307 (Inggit. 2019)

RTC DS1307 memiliki spesifikasi dan fitur :

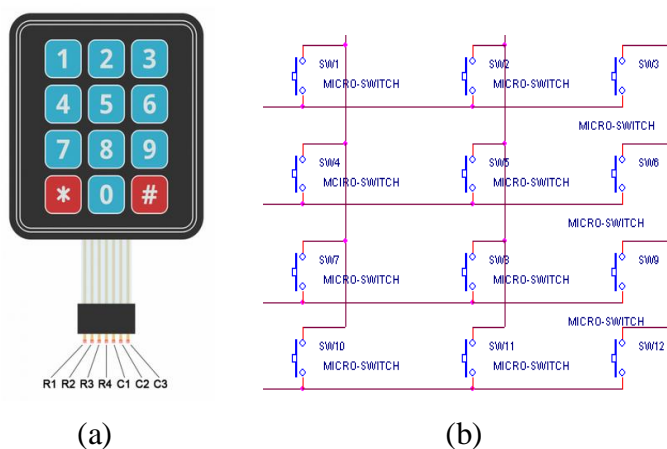
1. RTC yang sangat akurat mengelola semua fungsi pengatur waktu.
2. Jam *Real Time* menghitung detik, menit, jam, tanggal, bulan, dan hari dalam seminggu, dan tahun, dengan kompensasi tahun lawan berlaku hingga 2100

3. Akurasi $\pm 2\text{ppm}$ dari 0°C sampai $+40^\circ\text{C}$
4. Akurasi $\pm 3.5\text{ppm}$ dari -40°C sampai $+85^\circ\text{C}$
5. Digital temperatur sensor output: $\pm 3^\circ\text{C}$
6. Mendaftar untuk aging trim g. *Active-Low RST Output/ Pushbutton*
7. *Two Time-of-Day Alarms*
8. *Output Programmable Square-Wave Output*
9. Antarmuka serial sederhana menghubungkan ke kebanyakan mikrokontroler
10. Kecepatan data transfer *I2C Interface (400kHz)*

2.8 Keypad 4x3

Menurut Mardiah (2019:3), Keypad adalah saklar-saklar *push button* yang disusun secara matriks yang berfungsi untuk menginput data.

Konstruksi matrix keypad 4x3 yaitu terdiri dari 4 baris dan 3 kolom dengan keypad berupa saklar *push button* yang diletakan disetiap persilangan kolom dan barisnya. Rangkaian matrix keypad terdiri dari 16 saklar *push button* dengan konfigurasi 4 baris dan 3 kolom. 7 line yang terdiri dari 4 baris dan 3 kolom tersebut dihubungkan dengan *port* mikrokontroler. Sisi baris dari matrix keypad ditandai dengan nama Row1, Row2, Row3 dan Row4 kemudian sisi kolom ditandai dengan nama Col1, Col2, dan Col3. Sisi input atau output dari matrix keypad 4x3 ini tidak mengikat, dapat dikonfigurasi kolom sebagai input dan baris sebagai output atau sebaliknya tergantung program yang digunakan.



Gambar 2.9 a. Tampilan Keypad 4x3 b. Konfigurasi Keypad 4x3 (Mikroavr. 2020)

Proses *scanning* untuk membaca penekanan tombol pada matrix keypad 4×3 untuk mikrokontroler dilakukan secara bertahap kolom demi kolom dari kolom pertama sampai kolom ke 3 dan baris pertama hingga baris ke 4. Program untuk *scanning* matrix keypad 4×3 dapat bermacam-macam, tapi pada intinya sama. Misal kita asumsikan keypad aktif rendah (semua line kolom dan baris dipasang resistor *pull-up*) dan dihubungkan ke port mikrokontroler dengan jalur kolom adalah jalur input dan jalur output maka proses *scanning* matrix keypad 4×3 diatas dapat dituliskan sebagai berikut:

1. Apabila Kolom 1 diberi logika '0', kolom kedua dan kolom ketiga diberi logika '1' maka program akan mengecek tombol 1, 4, 7, dan *, sehingga apabila salah satu baris berlogika '0' maka ada tombol yang ditekan.
2. Apabila Kolom 2 diberi logika '0', kolom pertama dan kolom ketiga diberi logika '1' maka program akan mengecek tombol 2, 5, 8, dan 0, sehingga apabila salah satu baris berlogika '0' maka ada tombol yang ditekan.
3. Apabila Kolom 3 diberi logika '0', kolom pertama dan kolom kedua diberi logika '1' maka program akan mengecek tombol 3, 6, 9, dan #, sehingga apabila salah satu baris berlogika '0' maka ada tombol yang ditekan.
4. Kemudian kembali ke semula, artinya program looping terus mendeteksi data kolom dan data baris, cara ini disebut *scanning* atau penyapuan keypad untuk mendapatkan saklar mana yang ditekan.

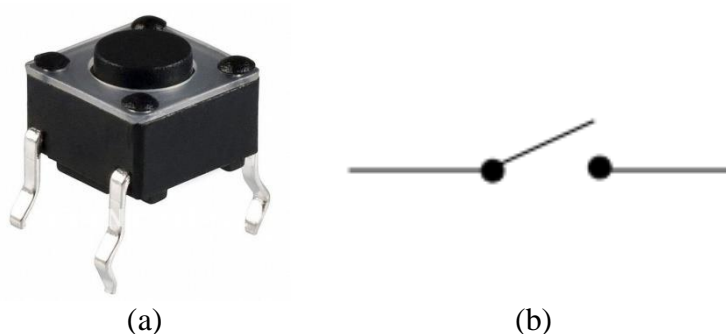
2.9 Push Button

Menurut Abdul Kadir (2017:224), *Push button* atau tombol tekan sering juga dinamakan sebagai *tactile switch* atau *momentary push button* yang memiliki 2 atau 4 terminal dibagian bawah yang dipisahkan oleh 2 kelompok terminal kiri dan kanan.

Push button adalah komponen elektikal yang berfungsi untuk memberikan sinyal atau untuk memutuskan dan menyambungkan suatu sistem rangkaian listrik. *Push button* berupa komponen kontaktor mekanik yang digerakan karena suatu kondisi tertentu. *Push button* merupakan komponen yang mendasar dalam sebuah rangkaian elektronika maupun rangkaian kontrol sistem. Komponen ini

sederhana namun memiliki fungsi yang paling vital di antara komponen listrik yang lain. Jadi *push button* pada dasarnya adalah suatu alat yang dapat berfungsi menghubungkan atau memutuskan aliran listrik (arus listrik) baik itu pada jaringan arus listrik kuat maupun pada jaringan arus listrik lemah. Yang membedakan *push button* arus listrik kuat dan saklar arus listrik lemah adalah bentuknya kecil jika dipakai untuk peralatan elektronika arus lemah, demikian pula sebaliknya semakin besar *push button* yang digunakan jika aliran arus listrik semakin besar.

Pada umumnya *push button* adalah tipe saklar yang hanya kontak sesaat saja saat ditekan dan setelah dilepas maka akan kembali lagi menjadi *normally open* (NO), biasanya saklar tipe NO ini memiliki rangkaian penguncinya dan tipe *normally open* (NO) digunakan untuk memberikan nilai digital “1” yang digunakan untuk mengaktifkan *logic*. *Push button* ada juga yang bertipe *normally closed* (NC), biasanya digunakan untuk tombol *off*. *Push button* juga berfungsi sebagai pemberi sinyal masukan pada rangkaian listrik, ketika/ selama bagian knopnya ditekan maka alat ini akan bekerja sehingga kontak-kontaknya akan terhubung untuk jenis *normally open* dan akan terlepas untuk jenis *normally close*, dan sebaliknya ketika knopnya dilepas kembali maka kebalikan dari sebelumnya, untuk membuktikannya pada terminalnya bisa digunakan alat ukur *tester* / ohm meter.



Gambar 2.10 a. Tampilan *push button* b. Simbok Saklar *push button* (Evelta. 2020)

2.10 Buzzer

Menurut Abdul Kadir (2017:236), *Buzzer* adalah komponen berukuran kecil yang digunakan untuk mengeluarkan suara.

Buzzer merupakan salah satu komponen elektronika yang dapat menghasilkan bunyi atau suara. *Buzzer* biasanya digunakan sebagai penanda pada sistem keamanan, absensi, bel rumah, jam alarm, dan pada rangkaian elektronika yang memerlukan indikator suara. Cara kerja *buzzer* yaitu ketika sinyal keluar dari mikrokontroler berlogika *high*, maka mikrokontroler akan mengirimkan sinyal ke *buzzer* sehingga memicu *buzzer* untuk bekerja. Ketika *buzzer* telah bekerja maka akan menciptakan suara yang telah diatur sesuai dengan instruksi program pada mikrokontroler. *Buzzer* dapat bekerja di frekuensi 1 - 100 kHz.

Terdapat 2 kabel yang dapat dihubungkan pada mikrokontroler yaitu :

1. Kabel berwarna merah, dipasangkan pada pin VCC.
2. Kabel berwarna hitam, dipasangkan pada pin GND.



Gambar 2.11 Buzzer
(Triyana, 2017)

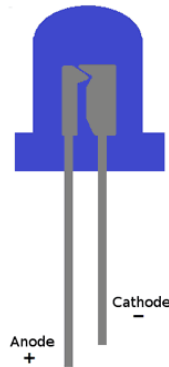
2.11 LED

Menurut Abdul Kadir (2017:22), *light emitting diode* atau LED adalah komponen yang dapat memancarkan apabila kaki pin *anode* terhubung ke tegangan positif dan kaki yang disebut *katode* terhubung ke *ground*.

LED atau *light emitting diode* dapat memancarkan cahaya monokromatik ketika diberikan tegangan *forward bias*. LED merupakan keluarga dioda yang terbuat dari bahan semikonduktor. Warna-warna Cahaya yang dipancarkan oleh LED tergantung pada jenis bahan semikonduktor yang dipergunakannya. Cara kerja LED hanya akan memancarkan cahaya apabila dialiri tegangan maju (*forward bias*) dari Anoda menuju ke Katoda. LED terdiri dari sebuah chip semikonduktor yang di *doping* sehingga menciptakan *junction* P dan N. Proses *doping* dalam semikonduktor adalah proses untuk menambahkan ketidakmurnian (*impurity*) pada semikonduktor yang murni sehingga menghasilkan karakteristik

kelistrikan yang diinginkan. Ketika LED dialiri *forward bias* yaitu dari anoda (A) menuju ke katoda (K), kelebihan elektron pada N-Type material akan berpindah ke wilayah yang kelebihan *hole* (lubang) yaitu wilayah yang bermuatan positif (P-Type material). Saat Elektron berjumpa dengan *hole* akan melepaskan photon dan memancarkan cahaya monokromatik (satu warna).

Untuk dapat membedakan pin pada LED, pada komponen LED terdapat dua katub yang jika diamati akan nampak perbedaannya. Pada pin *anode* biasanya katub akan lebih kecil dan kaki pinnya akan lebih panjang. Lalu pada pin *katode* biasanya katub akan lebih besar dan kaki pinnya akan lebih pendek daripada pin *anode*. Untuk memperjelas dapat dilihat pada gambar 2.9 sebagai berikut :



Gambar 2.12 LED
(Triyana, 2017)

2.12 Adaptor

Adaptor adalah sebuah perangkat berupa rangkaian elektronika untuk mengubah tegangan listrik yang besar menjadi tegangan listrik lebih kecil, atau rangkaian untuk mengubah arus bolak-balik (arus AC) menjadi arus searah (arus DC). Adaptor/ *power supply* merupakan komponen inti dari peralatan elektronik. Adaptor digunakan untuk menurunkan tegangan AC 220 Volt menjadi kecil antara 3 volt sampai 12 volt sesuai kebutuhan alat elektronika.

Terdapat 2 jenis adaptor berdasarkan sistem kerjanya, adaptor sistem trafo *step down* dan adaptor sistem *switching*. Dalam prinsip kerjanya kedua sistem adaptor tersebut berbeda, adaptor *stepdown* menggunakan teknik induksi medan magnet, komponen utamanya adalah kawat email yang di lilit pada teras besi, terdapat 2 lilitan yaitu lilitan primer dan lilitan skunder. Ketika listrik masuk

kelilitan primer maka akan terjadi induksi pada kawat email sehingga akan terjadi gaya medan magnet pada teras besi kemudian akan menginduksi lilitan skunder. Sedangkan sistem *switching* menggunakan teknik transistor maupun IC *switching*, adaptor ini lebih baik dari pada adaptor teknik induksi, tegangan yang di keluarkan lebih stabil dan komponennya suhunya tidak terlalu panas sehingga mengurangi tingkat resiko kerusakan karena suhu berlebih, biasanya *regulator* ini di gunakan pada peralatan elektronik digital. Adaptor dapat dibagi menjadi empat macam, diantaranya adalah sebagai berikut :

1. Adaptor *DC Converter* adalah sebuah adaptor yang dapat mengubah tegangan DC yang besar menjadi tegangan DC yang kecil. Misalnya : Dari tegangan 12v menjadi tegangan 6v;
2. Adaptor *Step Up* dan *Step Down*. Adaptor *Step Up* adalah sebuah adaptor yang dapat mengubah tegangan AC yang kecil menjadi tegangan AC yang besar. Misalnya : Dari Tegangan 110v menjadi tegangan 220v. Sedangkan Adaptor *Step Down* adalah adaptor yang dapat mengubah tegangan AC yang besar menjadi tegangan AC yang kecil. Misalnya : Dari tegangan 220v menjadi tegangan 110v.
3. Adaptor *Inverter* adalah adaptor yang dapat mengubah tegangan DC yang kecil menjadi tegangan AC yang besar. Misalnya : Dari tegangan 12v DC menjadi 220v AC.
4. Adaptor *Power Supply* adalah adaptor yang dapat mengubah tegangan listrik AC yang besar menjadi tegangan DC yang kecil. Misalnya : Dari tegangan 220v AC menjadi tegangan 6v, 9v, atau 12v DC.

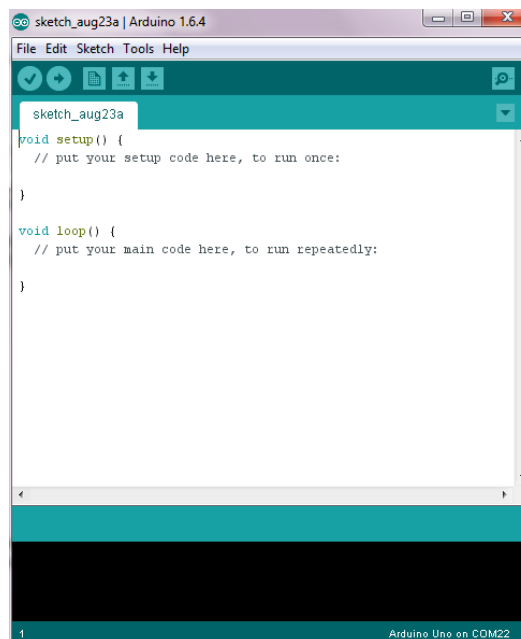


Gambar 2.13 Adaptor *Power Supply*
(Damayanti. 2017)

2.13 Software Arduino IDE

Menurut Abdul Kadir (2017:2), Arduino IDE (*Integrated Development Environment*) adalah sebuah perangkat lunak yang berfungsi dalam pengendalian dan perancangan program di dalam *development board*.

Secara harfiah berarti arduino IDE mempunyai bahasanya sendiri yang menyerupai bahasa C. Arduino IDE dibuat dari bahasa pemrograman JAVA. Arduino IDE juga dilengkapi dengan library C/ C++ yang biasa disebut *Wiring* yang membuat operasi input dan output menjadi lebih mudah. Arduino IDE ini dikembangkan dari *software Processing* yang dirombak menjadi Arduino IDE khusus untuk pemrograman dengan Arduino dan *development kit* lainnya.



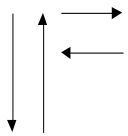

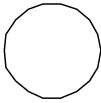
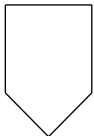

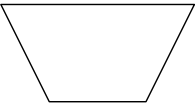
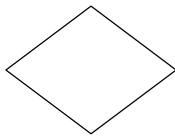
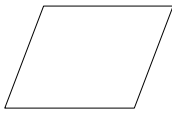
Gambar 2.14 Tampilan *software* Arduino IDE
(Kelas robot. 2020)

Program yang ditulis dengan menggunakan Arduino Software (IDE) disebut sebagai sketch. Sketch ditulis dalam suatu editor teks dan disimpan dalam file dengan ekstensi *.ino*. Teks editor pada Arduino Software memiliki banyak fitur seperti *cutting/paste* dan *searching/ replacing* sehingga memudahkan kamu dalam menulis kode program.

2.14 Teori Dasar *Flowchart*

Flowchart atau diagram alir merupakan sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran algoritma atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing-masing langkah tersebut menggunakan tanda panah.

Tabel 2.5 Simbol – simbol *flowchart*.

No	Simbol	Keterangan
1.		Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga connecting line.
2.		Yaitu simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan
3.		Yaitu simbol untuk keluar – masuk atau penyambungan proses dalam lembar / halaman yang sama.
4.		Yaitu simbol untuk keluar – masuk atau penyambungan proses pada lembar / halaman yang berbeda.
5.		Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer
6.		Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh komputer
7.		Simbol pemilihan proses berdasarkan kondisi yang ada.
8.		Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya

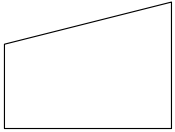
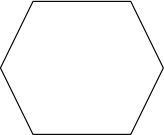
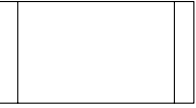
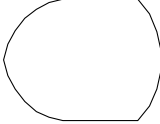

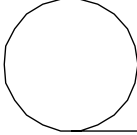
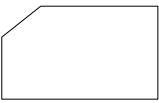

9.		Simbol untuk pemasukan data secara manual on-line keyboard
10.		Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam storage.
11.		Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure
12.		Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.
13.		Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.
14.		Simbol yang menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
15.		Simbol yang menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
16.		Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

Diagram ini memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut. Flowchart disusun dengan symbol - simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses di dalam program. (Mardiah. 2019:13)