

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu ini menjadi suatu acuan penulis dalam membuat laporan akhir sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Berikut merupakan penelitian terdahulu berupa beberapa jurnal yang terkait dengan judul laporan akhir penulis.

Pada penelitian sebelumnya yang dilakukan oleh (Fendy Setiawan dan Endang Sri Rahayu, 2020), dengan judul **“Sistem Security Door Lock Berbasis Gerakan dengan Pengiriman Gambar Menggunakan Internet of Things”**. Penelitian ini bertujuan untuk menyediakan perangkat yang mampu memberikan informasi tentang kondisi pintu rumah. Metode yang dipergunakan adalah dengan mengaplikasikan sistem operasi Android menggunakan teknologi Internet of Things (IoT) agar memudahkan pengguna memantau serta mengontrol penguncian pintu (door lock) rumah dari jarak jauh. Penggunaan sensor PIR sebagai pendeteksi keadaan pintu serta kamera ESP32-Cam untuk mengambil gambar (foto) orang yang membuka atau menutup pintu. Foto akan dikirimkan ke pemilik rumah dan tersimpan ke *Google Drive*.

Pada penelitian sebelumnya yang dilakukan oleh (Andi Setiawan, 2019) dengan judul **“Pengembangan Passive Infrared Sensor (PIR) HC-SR501 dengan Microcontrollers ESP32-CAM Berbasis Internet of Things (IoT) dan Smart Home sebagai Deteksi Gerak untuk Keamanan Perumahan”**. Secara teknis kedua perangkat keras akan bekerja secara optimal ketika HC-SR501, sensor PIR terhubung melalui ESP32-CAM mikrokontroler yang bertugas mengirimkan gambar atau video, ketika radius dari HC-SR501 dan sensor PIR yang mengandung pancaran infra merah melalui lensa Fresnel dan mengandung energi panas mengenai sensor pyroelektrik terpenuhi. Kemudian gambar atau video yang terkirim melalui ESP32-CAM yang sebelumnya telah diprogram melalui arduino IDE, dihubungkan melalui Wi-Fi dan diterima melalui perangkat *smartphone* milik penghuni rumah untuk mendapatkan respon terhadap hasil dari deteksi HC-

SR501, sensor PIR dan ESP32-CAM. Hasil yang diperoleh dari penelitian ini adalah pada jarak efektif dari HC-SR501 dan sensor PIR saat mengenai human detektor adalah 0-5 meter, ESP32-CAM akan mengirimkan gambar yang artinya ada indikasi pencuri atau orang yang tidak dikenal memasuki rumah. Sedangkan pada jarak 5 meter lebih, HC-SR501 dan sensor PIR tidak mengirimkan gambar yang artinya aman.

Pada penelitian sebelumnya yang dilakukan oleh (Kaleb Yefune Sun, dkk. 2021), dengan judul “**Perancangan Sistem IoT Menggunakan Aplikasi Blynk**”. Alat ini dirancang ketika tamu datang, tamu melihat peringatan sensor sebelum pintu masuk dan kemudian bergerak menuju sensor pemberitahuan yang telah direncanakan dengan aplikasi *blynk* sebelumnya, Aplikasi *blynk* mengirimkan pemberitahuan melalui ponsel, setelah pemilik properti menyadari bahwa ada seseorang di depannya dan baru saja membuka kunci *Savvy Entryway Lock* dengan memanfaatkan ponsel jika pemilik properti mengenal tamu tersebut.

Pada penelitian sebelumnya yang dilakukan oleh (Jacqueline Waworundeng, dkk, 2017) dengan judul “**Implementasi Sensor PIR sebagai Pendeteksi Gerakan untuk Sistem Keamanan Rumah menggunakan Platform IoT**”. Alat ini dirancang untuk membantu pemilik rumah untuk mendeteksi gerakan yang terjadi di rumah, ketika pemilik rumah tidak berada di rumah. Pendeteksi gerakan atau detektor yang dirancang, bertujuan untuk membantu sistem keamanan rumah. Sensor PIR digunakan sebagai pendeteksi gerakan yang mengirimkan notifikasi kepada user melalui aplikasi *Blynk* yang sudah di instal pada *smartphone*. User dapat melihat dan mengakses data *logging* berupa grafik melalui platform IoT yaitu *thingspeak.com*. Metode yang digunakan dalam penelitian ini adalah metode 5 Rekayasa Perangkat Lunak (RPL) serta menggunakan proses model *prototype*. Alat dibuat dengan menggunakan empat sensor PIR dan satu *WEMOS board* mikrokontroler dengan modul Wi-Fi ESP8266 terintegrasi, yang berfungsi untuk mengirimkan hasil *input* data sensor ke *Internet of Things (IoT)* platform yaitu *Blynk* dan *Thingspeak*.

Penelitian yang dilakukan oleh (Anita, 2017) dengan judul **“Sistem Keamanan Rumah Menggunakan IoT”**. Sistem keamanan yang digunakan adalah mikrokontroler Arduino Uno sebagai penghubung antar komponen yaitu sensor reed magnetik, *buzzer*, modul *wifi*, dan Esp8266 untuk terhubung dan berkomunikasi dengan internet. Sistem akan memberi tahu pemilik rumah jika ada penyusup dan mengirimkan notifikasi kepada pengguna. Setelah mendapat notifikasi dia dapat mengambil tindakan yang diperlukan.

Pada penelitian sebelumnya yang dilakukan oleh (Dedi Setiawan, 2019) dengan judul **“Perancangan Sistem Pengontrol Keamanan Rumah Dengan Smart CCTV Menggunakan Arduino Berbasis Telegram”**. Dalam prototype alat ini cara kerjanya adalah ketika perangkat diaktifkan NodeMCU ESP8266 akan langsung terhubung ke jaringan Wi-Fi yang ditentukan, kamera yang terhubung ke monitor pengawasan juga dapat secara otomatis mengambil gambar pada saat yang sama dan mengirim gambar ke aplikasi Telegram ketika sensor mendeteksi gerakan, bel juga akan secara otomatis berdering sehingga dapat memberikan informasi kepada orang-orang di sekitar bahwa di rumah ada upaya kejahatan.

Penelitian yang dilakukan oleh (Kodali, 2016) dengan judul **“Keamanan berbasis IoT dengan Sistem Otomatisasi Rumah”**. Pembuatan ini dibuat dengan sistem *wireless* yang mengirimkan peringatan kepada pemiliknya dengan menggunakan internet jika ada penyusup dan membunyikan alarm secara opsional. Keuntungan sistem keamanan ini peringatan dan status yang dikirim oleh sistem yang dikelola mikrokontroler terhubung wifi dapat diterima oleh pengguna di ponselnya dari jarak berapa pun terlepas dari apakah ponselnya terhubung ke internet. Mikrokontroler yang digunakan dalam *prototype* ini adalah papan Launchpad TI-CC3200 yang dilengkapi dengan mikrokontroler tertanam dan pelindung Wi-Fi.

Penelitian yang dilakukan oleh (Abu, 2018) dengan judul **“Desain dan Pengembangan Sistem Keamanan Rumah berdasarkan Internet of Things Via Favoriot Platform”**. Sistem ini menggunakan IoT dengan *database server online* FAVORIOT yang dilengkapi dengan sensor PIR dan sensor inframerah. Sensor ini akan memantau keberadaan penyusup dengan menggunakan aplikasi *Blynk*. Data

yang diterima dari sensor akan dikirim ke mikrokontroler yang telah dilengkapi dengan modul internet. Platform FAVORIOT akan menerima data yang dapat membantu pengguna untuk memantau rumah dan mengirimkan peringatan real time kepada pengguna.

Penelitian yang dilakukan oleh (Chandra, dkk. 2017) dengan judul **“Rumah Berbasis IoT dengan Keamanan Cerdas”**. Sistem yang direncanakan adalah menangkap gambar penyusup dan mengirimkannya ke surat resmi melalui internet melalui Simple Mail Transfer Protocol (SMTP). Alat ini dirancang untuk memberikan keamanan dan mengontrol peralatan rumah tangga di dalam rumah oleh pemiliknya melalui server IoT. Mikrokontroler yang digunakan adalah Raspberry Pi 3 untuk operasi pemrosesan dan pengontrolan. Perangkat lain yang digunakan yaitu sensor suhu LM35, *Light Dependent Resistor* (LDR), sensor PIR dengan *magnetic door switch*, *smoke detector* dihubungkan dengan kamera dan koneksi LAN.

Penelitian yang dilakukan oleh (Suresh, dkk. 2016) dengan judul **“Sistem Pemantauan dan Keamanan Rumah”**. Sistem keamanan rumah yang ekonomis dan terjangkau yang telah mengintegrasikan komponen keamanan dengan memanfaatkan sensor PIR, sensor suhu, sensor kelembaban untuk merasakan gerakan, perubahan suhu dan kelembaban di ruangan dari normal. Pemilik rumah akan diberitahu jika ada penyusup dengan cara mengirim pesan teks dengan menggunakan modul GSM. Aktivitas ini dikendalikan oleh mikrokontroler Atmega Arduino.

Tabel 2.1 Perbandingan Penelitian Terdahulu dengan Penelitian Sekarang

No	Penelitian	Persamaan	Perbedaan
1	Fendy Setiawan, Endang Sri Rahayu, "Sistem <i>Security Door Lock</i> Berbasis Gerakan dengan Pengiriman	1. Menggunakan Esp32-Cam sebagai mikrokontroler 2. Menggunakan sensor PIR	1. Hasil gambar dikirim ke aplikasi Telegram.

	Gambar menggunakan <i>Internet of Things</i> .		
2	Andi Setiawan, Ade Irma Purnamasari Pengembangan <i>Passive Infrared Sensor (PIR)</i> HC-SR501 dengan <i>Microcontrollers ESP32-CAM</i> Berbasis <i>Internet of Things (IoT)</i> dan <i>Smart Home</i> sebagai Deteksi Gerak untuk Keamanan Perumahan	1. Menggunakan Esp32-Cam sebagai mikrokontroler. 2. Menggunakan sensor PIR.	1. Hasil gambar dikirim ke aplikasi Telegram.
3	Kaleb Yefune Sun, Yonky Pernando, M.Ibnu Safari. Perancangan Sistem IoT Menggunakan Aplikasi <i>Blynk</i>	1. Menggunakan Esp32-Cam sebagai mikrokontroler. 2. Menggunakan <i>buzzer</i> sebagai alarm.	1. Menggunakan sensor PIR untuk mendeteksi gerak. 2. Hasil gambar dikirim secara otomatis melalui aplikasi telegram.

Berdasarkan jurnal diatas terdapat beberapa persamaan yaitu menggunakan Esp32-Cam sebagai mikrokontroler yang sekaligus dapat menangkap gambar. Adapun perbedaan dari ketiga jurnal penelitian tersebut yaitu pada jurnal 1 dan jurnal 2 menggunakan aplikasi telegram untuk menampilkan hasil gambar yang mana pada jurnal 1 menggunakan *google drive*, dan jurnal 2 menggunakan *website* untuk menampilkan hasil gambarnya sehingga tidak ada notifikasi jika gerakan terdeteksi. Pada jurnal penelitian 3 menggunakan sensor PIR untuk mendeteksi gerakan dan menampilkan gambar secara otomatis melalui aplikasi telegram. Sedangkan pada alat yang akan penulis buat menggunakan Esp32-Cam sebagai

mikrokontroler, yang dibangun menggunakan sensor PIR, *buzzer* dan aplikasi telegram.

2.2 Sistem

Sistem adalah kumpulan dari sub-sub *system* abstrak maupun fisik yang saling terintegrasi dan berkolaborasi untuk mencapai suatu tujuan tertentu (Fadli, 2017). Jika dilihat dari sudut pandang sistem informasi yang berorientasi objek, sistem merupakan sekumpulan komponen yang mengimplementasikan model dan fungsionalitas yang dibutuhkan, komponen tersebut saling berinteraksi di dalam sistem guna mentransformasikan *input* yang diberikan kepada sistem tersebut menjadi *output* yang berguna dan bernilai bagi aktor-nya (Handarly & Lianda, 2018).

2.3 Keamanan

Secara umum, keamanan adalah keadaan bebas dari bahaya. Istilah ini bisa digunakan dengan hubungan kepada kejahatan, segala bentuk kejahatan, dan lain-lain. Keamanan merupakan topik yang luas termasuk keamanan nasional terhadap serangan teroris, keamanan komputer terhadap *hacker* atau *cracker*, keamanan rumah terhadap maling dan penyusup lainnya, keamanan finansial terhadap kehancuran ekonomi dan banyak situasi berhubungan lainnya. (Permana Cresta, 2013)

2.4 Monitoring

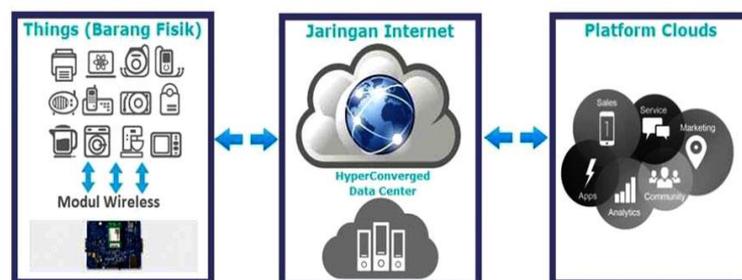
Monitoring berhubungan dengan siklus kegiatan yang mencakup pengumpulan, peninjauan ulang, pelaporan, dan tindakan atas informasi suatu proses yang sedang diimplementasikan, sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan kegiatan itu selanjutnya (Wijaya & Rivai, 2018).

Dalam pengertian lain, *monitoring* merupakan pemantauan yang dapat dijelaskan sebagai kesadaran (*awareness*) tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan ke arah tujuan atau menjauh dari itu.

Monitoring akan memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu ke waktu, pemantauan umumnya dilakukan untuk tujuan tertentu (Zaida & Sunardi, 2019).

2.5 Internet Of Things (IoT)

Internet of Things (IoT) merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus menerus. *Internet of Things* (IoT) mengacu pada benda yang diidentifikasi secara unik sebagai representasi virtual dalam struktur berbasis internet. *Internet of Things* (IoT) sebagai sebuah infrastruktur jaringan global, yang menghubungkan benda-benda fisik dan virtual melalui eksploitasi data *capture* dan kemampuan komunikasi dengan sensor dan koneksi sebagai pengembangan layanan. Dalam hal tersebut dapat disimpulkan bahwa IoT mengacu dan memanfaatkan pada suatu benda yang nantinya benda tersebut akan dapat berkomunikasi antara satu dengan yang lain melalui sebuah jaringan internet. (Kusumaningrum, dkk. 2017)



Gambar 2.1 Konsep IoT

2.6 Sensor Gerak PIR (*Passive Infra Red*)

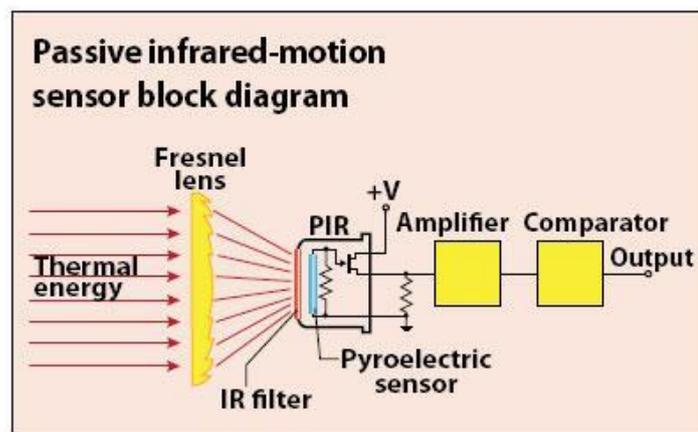
Sensor PIR adalah sensor yang digunakan untuk mendeteksi adanya pancaran sinar infra merah. Sensor PIR ini bersifat pasif, artinya sensor ini tidak memancarkan sinar infra merah tetapi hanya menerima radiasi sinar infra merah dari luar. Sesuai dengan namanya *Passive*, sensor ini hanya merespon energi dari pancaran sinar inframerah pasif yang dimiliki oleh setiap benda yang terdeteksi olehnya. Benda yang bisa dideteksi oleh sensor ini biasanya adalah tubuh manusia. (Alfazri, 2015)



Gambar 2.2 Sensor Gerak PIR

2.6.1 Bagian-bagian Sensor PIR

Menurut Saputra (2014:3), di dalam sensor PIR terdapat bagian-bagian yang mempunyai perannya masing-masing, yaitu *Fresnel Lens*, *IR Filter*, *Pyroelectric sensor*, *amplifier*, dan *comparator*.



Gambar 2.3 Bagian Sensor PIR

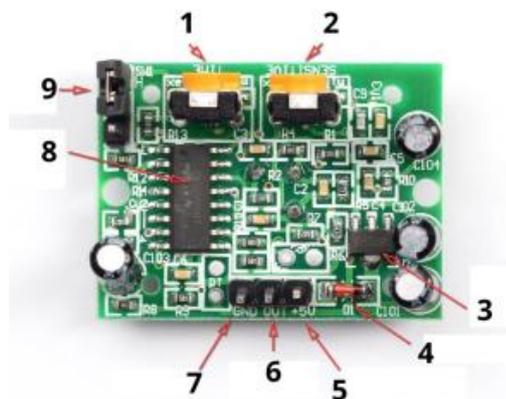
1. *Fresnel Lens* : untuk memfokuskan sinar terang, tetapi juga karena intensitas cahaya yang relatif konstan di seluruh lebar berkas cahaya
2. *IR Filter* : *IR Filter* di modul sensor PIR ini mampu menyaring panjang gelombang sinar infrared pasif antara 8 sampai 14 mikrometer, sehingga panjang gelombang yang dihasilkan dari tubuh manusia yang berkisar antara 9 sampai 10 mikrometer ini saja yang dapat dideteksi oleh sensor. Sehingga Sensor PIR hanya bereaksi pada tubuh manusia saja.
3. *Pyroelectric sensor* : Seperti tubuh manusia yang memiliki suhu tubuh kira-kira 320 C, yang merupakan suhu panas yang khas yang terdapat pada lingkungan. Pancaran sinar inframerah inilah yang kemudian ditangkap

oleh *Pyroelectric sensor* yang merupakan inti dari sensor PIR.

4. *Amplifier* : Sebuah sirkuit amplifier yang ada menguatkan arus yang masuk pada material pyroelectric.
5. Komparator: Setelah dikuatkan oleh amplifier kemudian arus dibandingkan oleh komparator sehingga menghasilkan *output*.

2.6.2 Komponen Sensor PIR

Gambar berikut menunjukkan komponen dari sensor PIR yang perlu untuk diketahui (Prima, 2015).



Gambar 2.4 Komponen Sensor PIR

1. Pengatur Waktu Jeda : Digunakan untuk mengatur lama pulsa *high* setelah terdeteksi terjadi gerakan dan gerakan telah berakhir.
2. Pengatur Sensitivitas : Pengatur tingkat sensitivitas sensor PIR.
3. Regulator 3VDC : Penstabil tegangan menjadi 3V DC.
4. Dioda Pengaman : Mengamankan sensor jika terjadi salah pengkabelan VCC dengan GND.
5. DC Power : *Input* tegangan dengan *range* (3 – 12) VDC (direkomendasikan menggunakan *input* 5VDC).
6. Output Digital : *Output* digital sensor.
7. Ground : Hubungkan dengan *ground* (GND).
8. BISS0001 : IC Sensor PIR.
9. Pengatur Jumper : Untuk mengatur *output* dari pin digital.

2.7 Telegram

Telegram adalah aplikasi layanan pengiriman pesan dengan fokus pada kecepatan dan keamanan. Kita dapat menggunakan telegram di semua perangkat kerja pada saat bersamaan, pesan kita dapat tersinkronisasi di sejumlah ponsel, tablet ataupun komputer (*Windows, Mac, Linux*). (Efendi & Chandra, 2019)

Dengan menggunakan aplikasi telegram, pengguna dapat mengirim pesan dalam bentuk apapun seperti foto, *video*, dokumen dalam jenis apapun (zip, jpd, mp3, dan lain-lain). Pengguna telegram juga dapat membuat grup dengan jumlah anggota mencapai 10.000 orang atau saluran untuk disiarkan ke member tak terbatas. Pengguna dapat menulis kontak telepon dan menemukan orang dengan nama pengguna mereka. Sebagai hasilnya, telegram seperti gabungan sms dan *email*, dan dapat mengurus semua kebutuhan pribadi atau bisnis. (Efendi & Chandra, 2019)



Gambar 2.5 Logo Telegram

2.7.1 Bot Telegram

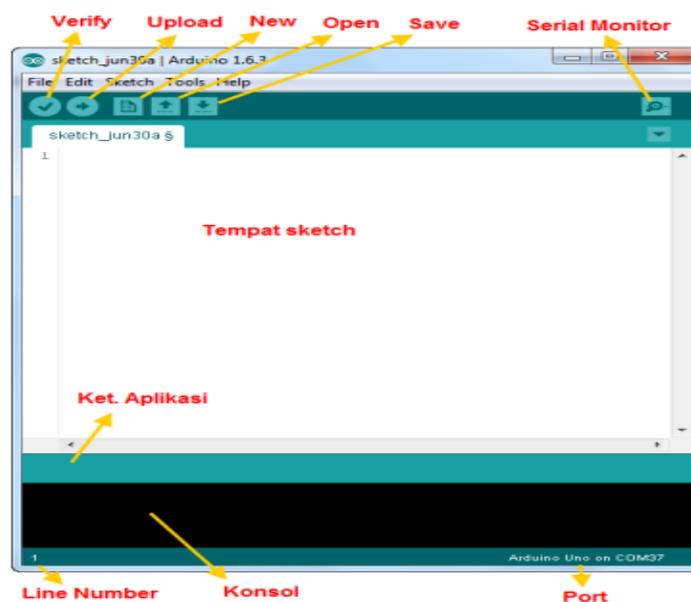
Bot merupakan kependekan daripada robot. Salah satu fungsi utama adanya bot adalah untuk memudahkan tugas manusia. Telegram merupakan salah satu aplikasi yang mendukung adanya bot ini. Dengan adanya bot ini memudahkan kita membuat semacam aplikasi chattingan khusus dan juga menggantikan tugas moderasi di dalam sebuah grup.

Adapun cara untuk membuat *account* Bot pada telegram cukup mudah, yaitu *search BotFather* pada telegram dan ditemukan *account BotFather*. Kirimkan pesan pada *BotFather : /start, /newbot, (Nama Bot), (Nama Bot)_bot*, maka *account* bot pada telegram akan tersedia dengan nama *account* yang sudah diatur pada awal memulai bot.

Untuk dapat mengintegrasikan fitur Bot pada mikrokontroler maupun mikroprosesor, diharuskan terlebih dahulu menguasai bahasa pemrograman seperti Python, Java, PHP dan lainnya. Hal itu disebabkan karena bot dengan kata lain robot bisa dijalankan atas dasar perintah, perintah itu dengan kata lain dibuat melalui bahasa pemrograman, apabila bot diberi perintah akan berjalan sesuai bahasa pemrograman yang sudah dibuat sesuai perintah yang ingin kita jalankan. (Yuliza, 2018)

2.8 Software Arduino IDE

Arduino IDE (*Integrated Development Environment*) merupakan sebuah *software* yang digunakan untuk menulis program, meng-*compile* menjadi kode biner dan mengunggah ke dalam memori mikrokontroler pada Arduino. Arduino IDE menggunakan bahasa pemrograman C++ dengan versi yang 20 telah disederhanakan, sehingga menjadi lebih mudah dalam penggunaan. Sebuah kode program Arduino pada umumnya biasa disebut dengan *sketch*. Arduino IDE dibuat dari bahasa pemrograman JAVA. Arduino IDE dilengkapi dengan library C/C++ yang biasanya disebut *wiring*, sehingga operasi *input* dan *output* menjadi lebih mudah. Arduino IDE dikembangkan dari *software processing* yang diubah menjadi Arduino IDE khusus untuk pemrograman Arduino. (Hermawan, 2016)



Gambar 2.6 Software Arduino IDE

Berikut adalah penjabaran dari bagian-bagian yang ada pada menu aplikasi Arduino IDE:

- ***Verify***

Pada versi sebelumnya dikenal dengan istilah *Compile*. Sebelum aplikasi di-*upload* ke *board* Arduino, biasakan untuk memverifikasi terlebih dahulu *sketch* yang dibuat. Jika ada kesalahan pada *sketch*, nanti akan muncul *error*. Proses *Verify / Compile* mengubah *sketch* ke *binary code* untuk di-*upload* ke mikrokontroler.

- ***Upload***

Tombol ini berfungsi untuk mengupload *sketch* ke *board* Arduino. Walaupun kita tidak mengklik tombol *verify*, maka *sketch* akan di-*compile*, kemudian langsung diupload ke *board*. Berbeda dengan tombol *verify* yang hanya berfungsi untuk memverifikasi source code saja.

- ***New Sketch***

Membuka window dan membuat *sketch* baru.

- ***Open Sketch***

Membuka *sketch* yang sudah pernah dibuat. *Sketch* yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi *file .ino*

- ***Save Sketch***

Menyimpan *sketch*, tapi tidak disertai dengan mengkompile.

- ***Serial Monitor***

Membuka interface untuk komunikasi serial, nanti akan kita diskusikan lebih lanjut pada bagian selanjutnya.

- **Keterangan Aplikasi**

Pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal “Compiling” dan “Done Uploading” ketika kita meng*compile* dan mengupload *sketch* ke *board* Arduino.

- **Konsol log**

Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* akan muncul pada bagian ini. Misal, ketika aplikasi meng*compile* atau ketika ada

kesalahan pada *sketch* yang kita buat, maka informasi error dan baris akan diinformasikan di bagian ini.

- **Baris *Sketch***

Bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.

- **Informasi *Board* dan *Port***

Bagian ini menginformasikan *port* yang dipakai oleh *board* Arduino.

2.9 Modul ESP-32 Cam

Esp32-Cam adalah papan pengembangan mode ganda Wifi + *Bluetooth* yang menggunakan antena dan inti papan PCB berbasis chip ESP32. Modul ini dapat bekerja secara independen sebagai sistem minimum. Modul ini merupakan sebuah modul Wifi yang sudah dilengkapi dengan kamera ov2640. Dari modul ini kita bisa digunakan untuk berbagai keperluan, contoh untuk CCTV, mengambil gambar dan sebagainya. Fitur lain yaitu kita bisa mendeteksi wajah (*face detection*) dan pengenalan wajah (*face recognition*). (Adjie, 2020). Maka dengan demikian, modul Esp-32 Cam ini dapat digunakan untuk mengambil gambar, dan juga dapat digunakan sebagai modul wifi untuk mengirim data. (Wicaksono, 2020)

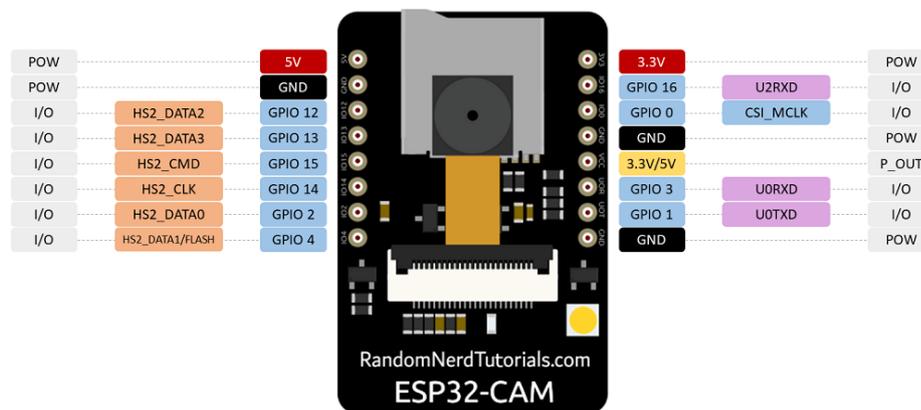
Tabel 2.2 Spesifikasi Esp32-Cam

Module Model	ESP32-CAM
Package	DIP-16
Size	27*40.5*4.5(±0.2)mm
SPI <i>Flash</i>	Default 32Mbit
RAM	520KB SRAM +4M PSRAM
Bluetooth	Bluetooth 4.2 BR/EDR and BLE standards
Wi-Fi	802.11 b/g/n/
<i>Support interface</i>	UART, SPI, I2C, PWM
<i>Support TF card</i>	Maximum support 4G
IO <i>port</i>	9
UART <i>Baudrate</i>	Default 115200 bps
Image <i>Output Format</i>	JPEG(OV2640 support only),BMP,GRAYSCALE
Spectrum Range	2412 ~2484MHz

Antenna	Onboard PCB antenna, gain 2dBi
Transmit Power	802.11b: 17±2 dBm (@11Mbps) 802.11g: 14±2 dBm (@54Mbps) 802.11n: 13±2 dBm (@MCS7)
Security	WPA/WPA2/WPA2-Enterprise/WPS
Power Supply Range	5V

2.9.1 Komunikasi Pin GPIO

Module ESP32-CAM di lengkapi dengan GPIO yang dapat di program sebagai *input* dan *output*, untuk lebih detail fitur atau fungsi pin khusus sebagai berikut :



Gambar 2.7 Modul ESP32-Cam

- **Power Pins**

Module ESP32-CAM memiliki 3 pin GND, bisa di lihat pada gambar yang warna hitam. Sedangkan untuk power ada 2 opsi bisa menggunakan 3.3V atau 5V yang terdapat pada pin warna merah. Jadi, development board ESP32-CAM bisa menggunakan 3.3V atau 5V, tetapi lebih baik menggunakan 5V supaya tegangan lebih stabil.

- **Power Output Pin**

Memiliki 1 *output* pin, bisa 3.3V atau 5V bisa diatur dengan cara di jumper diboardnya.

- **Serial Pins**

Module Development Board ESP32-CAM memiliki serial *hardware* di pin GPIO1 untuk U0TXD dan pin GPIO3 untuk U0RXD. Pin ini dapat digunakan untuk komunikasi dengan sensor atau module-module yang menggunakan komunikasi serial.

Akan tetapi, ESP32-CAM tidak memiliki *built-in programmer*. Jadi membutuhkan pin serial ini untuk *upload* atau *flash* program ke ESP32-CAM menggunakan serial programmer seperti *FTDI Module*.

- **GPIO 0**

Pin GPIO 0 digunakan untuk *Flash Mode Selection*, jadi Pin GPIO 0 digunakan untuk menyetting ESP32-CAM dalam mode *upload* atau *flash* program ketika terhubung ke GND. Saat selesai di *upload*, GPIO 0 di lepas dari pin GND supaya modusnya kembali ke system normal. Jadi system bisa jalan normal keluar dari mode *flash* atau *upload*, maka pin ini dihubungkan dengan ground ketika akan *upload* atau *flash* saja.

- **Module Micro Sd Card Reader**

ESP32-CAM sudah terintegrasi dengan module microSD, dapat difungsikan untuk menyimpan gambar atau data logger lainnya. Untuk pin yang di pakai detailnya sebagai berikut :

1. Module Micro SD ESP32
2. CLK GPIO 14
3. CMD GPIO 15
4. DATA0 GPIO 2
5. DATA1
6. GPIO 4
7. DATA2
8. GPIO 12
9. DATA3 GPIO 13

- **Module Kamera**

Module Kamera ESP32-CAM sudah terintegrasi dengan FPC *Camera Connector* 24 pin dan di paket penjualan sudah mendapat Modul OV2640 untuk cameranya, jadi tinggal di pasang di *connector FPC cameranya*. Selanjutnya Module *Camera* ESP32-CAM sudah siap di pakai dan sudah terintegrasi dengan module *Camera* OV2640.

Berikut koneksi camera dengan ESP32-CAM AI-Thinker :

1. OV2640 ESP32
2. D0 GPIO 5
3. D1 GPIO 18
4. D2 GPIO 19
5. D3 GPIO 21
6. D4 GPIO 36
7. D5 GPIO 39
8. D6 GPIO 34
9. D7 GPIO 35
10. XCLK GPIO 0
11. PCLK GPIO 22
12. SYNC GPIO 25
13. HREF GPIO 23
14. SDA GPIO 26
15. SCL GPIO 27
16. POWER PIN GPIO 32
17. GPIO 4

Pin ini merupakan *built-in flash* dan bisa digunakan untuk *flash light camera* atau fungsi lainnya.

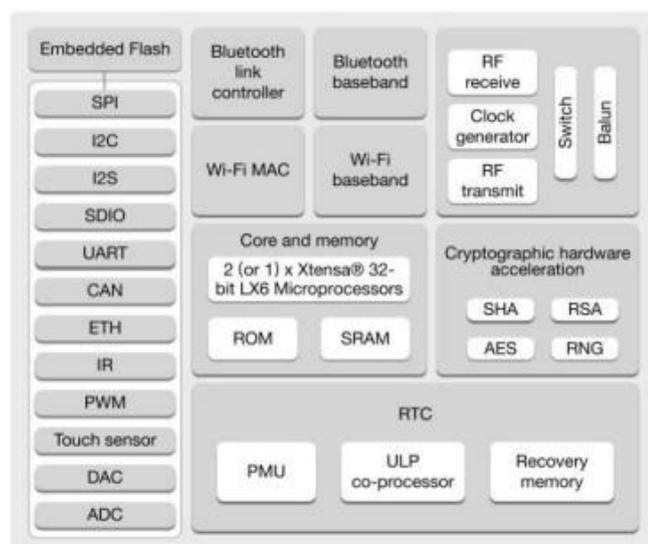
- **GPIO 33**

ESP32-CAM memiliki *built-in* yang sudah terintegrasi di modulnya, jadi bisa digunakan untuk test program Blink atau untuk test status program.

Module ESP32-CAM dapat diaplikasikan untuk project IoT (Internet of Things), misalnya *video monitoring, email telegram, whatsapp monitoring, face recognition, camera Wifi robot* dan project-project lainnya.

Fitur dari modul EPS32-CAM adalah :

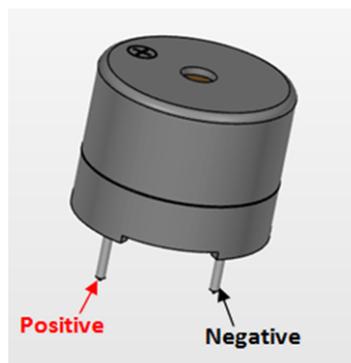
- Modul Ultra-small 802.11b / g / n Wifi + BT / BLE SoC
- Daya rendah dual-core 32-bit CPU untuk prosesor aplikasi
- Hingga 240MHz, hingga 600 DMIPS
- Built-in 520 KB SRAM, eksternal 4M PSRAM
- Mendukung antarmuka seperti UART / SPI / I2C/PWM / ADC / DAC
- Mendukung kamera OV2640 dan OV7670 dengan *flash* built-in
- Dukungan untuk upload gambar WiFi
- Dukungan kartu TF
- Mendukung beberapa mode tidur
- Tertanam Lwip dan FreeRTOS
- Mendukung mode kerja STA / AP / STA + AP
- Dukungan Smart Config / AirKiss Jaringan distribusi sekali klik
- Dukungan untuk peningkatan lokal serial dan peningkatan firmware jarak jauh (FOTA)
- Mendukung pengembangan sekunder



Gambar 2.8 Blok Diagram Pada Esp32-Cam

2.10 Buzzer

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan *loud 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. *Buzzer* biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat. (Rafiuddin Syam, 2013)



Gambar 2.9 Buzzer

Tabel 2.3 Konfigurasi PIN Buzzer

PIN	Deskripsi
Positif	Diidentifikasi dengan simbol (+) atau ujung terminal yang lebih panjang. Dapat didukung oleh 6V
Negatif	Diidentifikasi oleh kabel terminal pendek. Biasanya terhubung ke <i>ground</i>

2.11 Saklar

Saklar adalah suatu alat dengan dua sambungan dan bisa memiliki dua keadaan, yaitu keadaan *on* dan keadaan *off*. Keadaan *off* (tutup) merupakan suatu

keadaan dimana tidak ada arus yang mengalir. Keadaan *on* (buka) merupakan satu keadaan yang mana arus bisa mengalir dengan bebas atau dengan kata lain (secara ideal) tidak ada resistivitas dan besar voltase pada saklar sama dengan nol. (Richard Blocher, 2004)

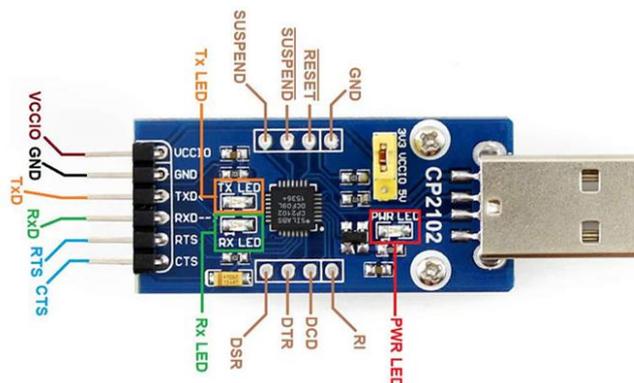


Gambar 2.10 Saklar

2.12 USB to TTL CP2102

USB to TTL adalah konverter USB ke TTL menggunakan *chip* konverter yang dapat digunakan pada *board* arduino seperti *lylypad*, *boarduino* atau *board arduino clone* sejenisnya. Selain untuk arduino, konverter ini juga bisa digunakan untuk segala macam aplikasi elektronik lainnya yang membutuhkan komunikasi serial melalui USB.

Perangkat ini cocok digunakan untuk pengguna *interfacing* yang menggunakan komputer dengan menggunakan *port* USB, karena alat ini mengkonvert dari USB menjadi serial, yang mana keluarannya berupa RX dan TX. (Gunardi & Munandar, 2017)



Gambar 2.11 USB to TTL CP2102

Tabel 2.4 Deskripsi PIN USB TTL CP2102

PIN	Deskripsi
VCCIO	Pin Out VCC (3.3v atau 5v) yang Dapat Dikonfigurasi Jumper
GND	Ground
PWR LED	LED Indikator Daya
TxD	<i>Output</i> data asinkron (Transmisi UART)
Tx LED	UART Transmit (data keluar) LED Indikator
RxD	<i>Input</i> data asinkron (UART Receive)
Rx LED	UART Receive (data masuk) LED Indikator
RTS	<i>Output</i> kontrol Siap Kirim (<i>active low</i>)
CTS	Hapus Untuk Mengirim <i>input</i> kontrol (<i>active low</i>)
GND	Ground
RESET	Atur Ulang Perangkat. <i>Output</i> saluran terbuka dari monitor POR atau VDD internal. Sumber eksternal dapat memulai pengaturan ulang sistem dengan mengarahkan pin ini ke posisi rendah setidaknya selama 15 μ s.
SUSPEND	Pin ini didorong tinggi ketika CP2102 memasuki status penangguhan USB
RI	Ring Indicator control <i>input</i> (<i>active low</i>)
DCD	<i>Input</i> kontrol Deteksi Pembawa Data (<i>active low</i>)
DTR	<i>Output</i> kontrol Siap Terminal Data (<i>active low</i>)
DSR	Data Set Ready control <i>input</i> (<i>active low</i>)

2.13 Flowchart

Flowchart menggunakan simbol untuk menggambarkan urutan suatu proses, termasuk proses pengolahan data. *Flowchart* sering digunakan untuk menggambarkan algoritma suatu aplikasi, urutan proses, prosedur, maupun aliran kerja (*workflow*). Untuk menggambar *flowchart* dibutuhkan alat bantu seperti *template* atau aplikasi seperti Microsoft Visio. Secara umum *flowchart* dikelompokkan menjadi empat, yaitu keluaran dan masukan, pengolahan, penyimpanan, dan simbol lainnya. *Flowchart* sudah lama digunakan, bahkan semenjak masa awal komputasi, meskipun sudah cukup lama digunakan *Flowchart* masih digunakan secara intensif. (Sarosa, 2017).

Dari beberapa definisi diatas dapat disimpulkan bahwa *flowchart* adalah diagram atau alur kerja suatu sistem yang menggambarkan urutan suatu proses pengolahan data suatu program.

2.13.1 Simbol Flowchart

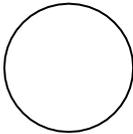
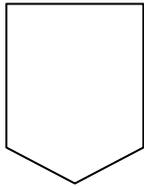
Simbol *Flowchart* terbagi menjadi 3 kategori yaitu (Sarosa, 2017) :

a. Simbol Arus (*Flow Direction Symbols*)

Biasanya simbol yang termasuk kedalam kategori ini digunakan sebagai simbol penghubung. Beberapa simbol yang termasuk ke dalam kategori ini, yaitu:

Tabel 2.5 Simbol Arus

No	Simbol	Nama	Fungsi
1		<i>Flow Direction Symbol/Connecting Line.</i>	Berfungsi untuk menghubungkan simbol yang satu dengan yang lainnya, menyatakan arus suatu proses.
2		<i>Communication Link</i>	Berfungsi untuk transmisi data dari satu lokasi ke lokasi lain.

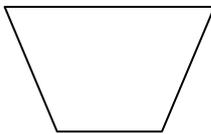
3		<i>Connector</i>	Berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
4		<i>Offline connector</i>	Berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.

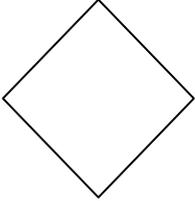
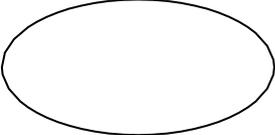
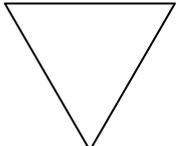
(Sumber: Sarosa, 2017).

b. Simbol Proses (*Processing Symbols*)

Simbol proses digunakan untuk menyatakan simbol yang berkaitan dengan serangkaian proses yang dilakukan. Berikut beberapa simbol yang termasuk ke dalam bagian proses, yaitu:

Tabel 2.6 Simbol Proses

No	Simbol	Nama	Fungsi
1		<i>Processing</i>	Berfungsi untuk menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
2		<i>Manual Operation</i>	Berfungsi untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.

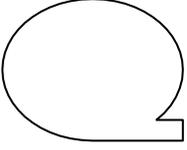
3		<i>Decision</i>	Berfungsi untuk memilih proses yang akan dilakukan berdasarkan kondisi tertentu.
4		<i>Terminal</i>	Berfungsi untuk menyatakan permulaan atau akhir suatu program.
5		<i>Predefined process</i>	Berfungsi untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
6		<i>Offline storage</i>	Berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
7		<i>Manual input</i>	Berfungsi untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i> .

(Sumber: Sarosa, 2017).

c. Simbol I/O (*Input-Output*)

Simbol yang termasuk kedalam bagian *input-output* berkaitan dengan masukan dan keluaran. Berikut beberapa simbol yang termasuk, yaitu:

Tabel 2.7 Simbol I/O

No	Simbol	Nama	Fungsi
1		<i>Input/output</i>	Berfungsi untuk menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya.
2		<i>Magnetic tape</i>	Berfungsi untuk menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis.
3		<i>Disk storage</i>	Berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
4		<i>Document</i>	Berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui printer).
5		<i>Punched card</i>	Berfungsi untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu.
6		<i>Display</i>	Berfungsi untuk menyatakan keluaran melalui layar monitor.

(Sumber: Sarosa, 2017).