

BAB II

TINJAUAN PUSTAKA

2.1 Internet OF Things (IOT)

IoT (*Internet of things*) merupakan suatu konsep yang bertujuan untuk memperluas manfaat dari konektivitas *internet* yang tersambung secara terus menerus. Pada dasarnya IoT (*Internet of things*) mengacu pada benda yang dapat diidentifikasi secara unik sebagai *representative* virtual dalam struktur berbasis internet. Cara kerja IoT (*Internet of things*) adalah interaksi antara sesama mesin yang terhubung secara otomatis tanpa campur tangan *user* dan dalam jarak berapapun (Nandika, 2020).

Internet of things merupakan teknologi baru dalam internet akses yang dapat mengenali objek perilaku intelijen terkait dengan pengambilan suatu keputusan dan dapat berkomunikasi dengan dirinya sendiri. IoT dapat menghubungkan berbagai objek tidak hidup melalui koneksi internet dan dapat menghubungkan mereka untuk berbagi informasi dan dapat melakukan proses otomatis (Ahdan & Susanto, 2021). *Internet of things* merupakan sebuah konsep di mana suatu benda atau objek ditanamkan teknologi-teknologi seperti sensor dan *software* dengan tujuan untuk berkomunikasi, mengendalikan, menghubungkan, dan bertukar data melalui perangkat lain selama masih terhubung ke internet.

2.2 Sistem

Sistem adalah seperangkat komponen yang saling berhubungan dan saling bekerjasama untuk mencapai beberapa tujuan. Selain itu pengertian yang lain sistem terdiri dari unsur-unsur dan masukan (*input*), pengolahan (*processing*), serta keluaran (*output*). Dengan demikian, secara sederhana sistem dapat diartikan sebagai kumpulan atau himpunan dari unsur atau variabel- variabel yang terorganisasi, saling berinteraksi dan saling bergantung satu sama lain. Sistem didesain untuk memperbaiki atau meningkatkan pemrosesan informasi. Setelah

dirancang, sistem diperkenalkan dan diterapkan ke dalam organisasi penggunanya. Jika sistem yang diterapkan itu digunakan maka implementasi sistem dapat dikatakan berhasil. Sedangkan jika para penggunanya menolak sistem yang diterapkan, maka sistem itu dapat digolongkan gagal (Agustin, 2018).

2.3 Pemberi Makan Kucing

Untuk memelihara kucing, pemberian pakan adalah yang menjadi salah satu faktor utama dan sebaiknya pemberian sesuai jadwal. Jika pemberian pakan langsung dengan porsi banyak dapat berakibat tersebut tidak habis dimakan oleh kucing karena sudah tidak renyak untuk di makan dan terbuang sia-sia. Akibat dari permasalahan ini maka timbulah rasa cemas dari segi kesehatan dan perkembangan kucing dan menimbulkan pengeluaran untuk pembelian pakan kucing yang berlebih. Hasil dari penelitian ini yaitu sebuah alat yang dapat membantu memberi pakan kucing secara real time dengan 2 mode pemberian pakan yaitu untuk penggunaan kucing dewasa/adult dan untuk kucing anakan/kitten dengan jadwal pemberian pakan pukul 8.00 WIB, 12.00 WIB dan 18.00 WIB. Pengontrolan alat ini dapat diakses secara nirkabel dengan smartphone android sehingga dapat memudahkan dalam memonitor status kerja alat ini. Manfaat yang diperoleh dari hasil pembuatan alat ini jika di lihat dari kucing yaitu dapat menjaga pola makan, mencegah mal nutrisi, mencegah obesitas serta mencegah kucing stres akibat dititipkan. Jika dilihat dari sisi manfaat terhadap pemilik kucing atau pengguna yaitu dapat meringankan kinerja dan meminimalisir lupa dalam pemberian pakan serta dapat menghemat dalam pembelian pakan kucing dikarenakan porsi sudah diatur sedemikian mungkin untuk menghemat pengeluaran.

2. 4 Analisis Kebutuhan Sistem

2.4.1 Mikrokontroler

Mikrokontroler merupakan sebuah *chip* yang bertugas sebagai pengatur rangkaian elektronika dan umumnya dapat ditanamkan program didalamnya. Berdasarkan definisi tersebut bisa dinyatakan bahwa mikrokontroller merupakan suatu integrated circuit yang dirancang dengan kepadatan tinggi, dimana bagian yang dibutuhkan suatu mikrokontroler sudah dibuat menjadi kepingan, biasanya

mencakup CPU (*Central Processing Unit*), EEPROM/EPROM/PROM/ROM, RAM (*Random Access Memory*), Parallel & Serial, *Timer* dan *Interrupt Controller* yang berfungsi sebagai pengatur rangkaian elektronik serta secara umum dapat ditanamkan program di dalamnya (Muhaemin, 2018:96-97).

Mikrokontroler adalah perkembangan dari mikroprosesor. Mikrokontroler adalah *Single Chip* Mikrokomputer (SCM), yaitu sebuah komputer yang di paket dengan sebuah *chip* (IC). Didalam mikrokontroler itu sendiri sudah terdapat RAM, ROM atau EPROM, *timer*, asilator, ADC, *buffer I/O port*, saluran alamat, serta saluran data sehingga dapat bekerja dengan tepat dan mampu melakukan pekerjaan yang rumit walaupun hanya dengan rangkaian sangat sederhana.

Mikrokontroler berfungsi sebagai chip untuuk pengendali rangkaian elektronika yang mempunyai kemampuan untuk menyimpan program. Perangkat ini tersusun dari CPU , memori, *input/output* (I/O) tertentu serta unit pendukung seperti ADC (*Analog to Digital Converter*) yang saling terintegrasi. Adapun jenis-jenis mikrokontroler yaitu:

1. MCS51

Mikrokontroler ini termasuk dalam keluarga mikrokontroler CISC. Sebagian besar instruksinya dieksekusi dalam 12 siklus *clock*. Yang termasuk dalam keluarga MCS-51 adalah mikrokontroler 8031 (versi 8051 tanpa EPROM), 8751, 8052, 8031, 8751H, 80C51, 80C31, 8052, dan 8032. Keluarga MCS-51 memiliki tipe CPU, RAM, *counter/timer*, *port paralel*, dan *port serial* yang sama.

2. AVR

Mikrokontroler *Alv and Vegard's Risc Processor* atau sering disingkat AVR merupakan mikrokontroler RISC 8 bit. Karena RISC inilah sebagian besar kode instruksinya dikemas dalam satu siklus *clock*. Secara umum, AVR dapat dikelompokkan dalam 4 kelas yaitu keluarga ATTiny, keluarga AT90Sxx, keluarga ATmega dan AT86FTxx.

3. PIC

PIC ialah keluarga mikrokontroler tipe RISC buatan *Mikrochip Technology*. Saat ini PIC telah dilengkapi dengan EPROM dan komunikasi serial, UAT, kernel kontrol motor serta memori program dari 512 word hingga 32 word. 8051 Salah satu contoh seri PIC yaitu PIC 16F88.

Contoh berikut memberikan gambaran opcode (*operation code*) yang digunakan dalam bahasa *assembly* untuk beberapa keluarga mikrokontroler:

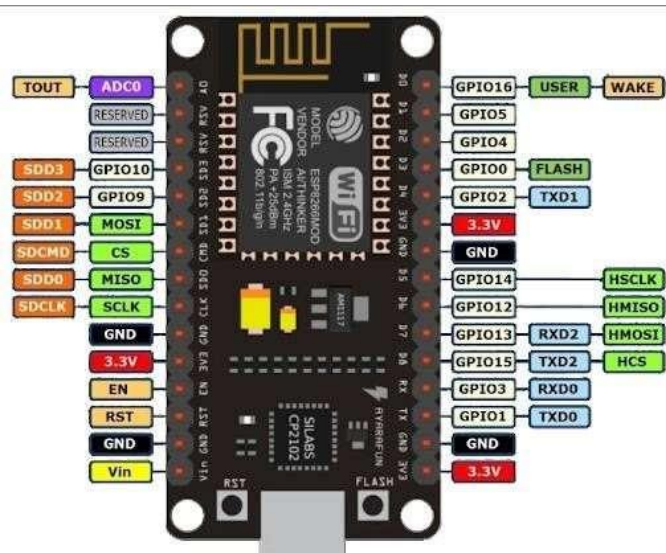
1. Keluarga mikrokontroler 8051
 - a. Intruksi transfer data : MOV, MOVX, MOVC
 - b. Instruksi Aritmatika : ADD, ADDC, SUBB, INC, DEC, MUL, DIV
 - c. Instruksi Logika : ANL, ORL, XRL, SETB, CLR
2. Keluarga mikrokontroler PIC16Cxx
 - a. Intruksi transfer data : MOVLW, MOVXF, MOVF
 - b. Instruksi Aritmatika : ADDLW, ADDWF, SUBLW, SUBWF, INCF, DECF
 - c. Instruksi logika : ANDLW, ANDWF, IORLW, IORWF, BCF, BSF
3. Keluarga mikrokontroler AVR
 - a. Intruksi transfer data : MOV, MOVX, MOVC
 - b. Instruksi Aritmatika : ADD, ADDC, SUB, SBC, INC, DEC, MUL, MULS
 - c. Instruksi Logika : AND, OR, EOR, SBR, CBR

Pada perancangan alat ini menggunakan Mikrokontroler ESP32.

2.4.2 NodeMCU ESP8266

NodeMCU adalah sebuah board elektronik yang berbasis chip ESP8266 dengan kemampuan menjalankan fungsi mikrokontroler dan juga koneksi internet (*WiFi*). Terdapat beberapa pin I/O sehingga dapat dikembangkan menjadi sebuah aplikasi monitoring maupun controlling pada proyek IOT. NodeMCU ESP8266 dapat diprogram dengan compiler-nya Arduino, menggunakan Arduino IDE. Bentuk fisik dari NodeMCU ESP 8266, terdapat port USB (mini USB) sehingga akan memudahkan dalam pemrogramannya. NodeMCU *ESP8266* merupakan

modul turunan pengembangan dari modul platform IoT (Internet of Things) keluarga ESP8266 tipe ESP-12. Secara fungsi modul ini hampir menyerupai dengan platform modul arduino, tetapi yang membedakan yaitu dikhususkan untuk “Connected to Internet“. Untuk saat ini modul NodeMCU sudah terdapat 3 tipe versi, namun yang digunakan dalam penelitian ini adalah NodeMCU 1.0 (*unofficial board*) Dikatakan *unofficial board* dikarenakan produk modul ini diproduksi secara tidak resmi terkait persetujuan dari *Developer Official NodeMCU*. Perbedaannya tidak begitu mencolok dengan versi 1.0 (*official board*) yaitu hanya penambahan V usb power output (Sulistyorini, Sofi, & Sova (2022).



Gambar 2. 1 Pin Out NodeMCU ESP8266

2.4.3 ESP32 CAM

Papan pengembangan WiFi dan Bluetooth dengan mikrokontroler Esp32 dan kamera. Modul ini menyediakan fitur yang dapat digunakan siapa saja, atau bisa dikatakan *open source*, salah satu fiturnya yaitu digunakan untuk mengambil gambar, Pengenalan wajah dan deteksi wajah. Modul periferan tersebut dapat digunakan menggunakan editor Arduino IDE untuk memanfaatkan *library* atau fitur yang sudah disediakan Esp32-Cam ini merupakan modul yang dapat digunakan pada banyak proyek juga merupakan modul lengkap dengan mikrokontroler terintegrasi, yang dapat membuatnya bekerja secara mandiri.

Selain konektivitas WiFi dan Bluetooth, modul ini juga memiliki kamera video terintegrasi, dan slot microSD untuk penyimpanan (Arrahman (2023-vol1)).



Gambar 2. 2 *ESP32-Cam*

2.4.4 LCD I2C

I2C LCD adalah modul LCD yang dikendalikan secara serial sinkron dengan protokol I2C/IIC (*Inter Integrated Circuit*) atau TWI (*Two Wire Interface*). Modul LCD pada normalnya dikendalikan secara paralel baik untuk jalur data maupun kontrolnya. Namun jalur paralel akan memakan banyak pin di sisi controller (misal Arduino, komputer ,dll). Setidaknya akan membutuhkan 6 atau 7 pin untuk mengendalikan sebuah modul LCD. Dengan demikian untuk sebuah controller yang harus mengendalikan banyak I/O, menggunakan jalur paralel adalah solusi yang kurang tepat. Modul I2C converter diperlihatkan pada Gambar 2.3 ini menggunakan chip ICPCF8574 produk dari NXP sebagai kontrolernya. IC ini adalah sebuah 8 bit I/O expander for I2c bus yang pada dasarnya adalah sebuah *shift register* (Suryantoro,Budiyanto (2019)).



Gambar 2. 3 LCD 12C

2.4.5 Motor Servo

Motor servo adalah sebuah motor dengan sistem closed feedback di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor, serangkaian gear, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Tampak pada gambar dengan pulsa 1.5 mS pada periode selebar 2 mS maka sudut dari sumbu motor akan berada pada posisi tengah. Semakin lebar pulsa OFF maka akan semakin besar gerakan sumbu ke arah jarum jam dan semakin kecil pulsa OFF maka akan semakin besar gerakan sumbu ke arah yang berlawanan dengan jarum jam.



Gambar 2. 4 Motor Servo

Motor servo adalah motor yang berputar lambat, dimana biasanya ditunjukkan oleh rate putarannya yang lambat, namun demikian memiliki torsi yang kuat karena internal gearnya. Lebih dalam dapat digambarkan bahwa sebuah motor servo memiliki :

- jalur kabel : power, ground, dan control
- Sinyal control mengendalikan posisi
- Operasional dari servo motor dikendalikan oleh sebuah pulsa selebar ± 20 ms, dimana lebar pulsa antara 0.5 ms dan 2 ms menyatakan akhir dari range sudut maksimum..
- Konstruksi didalamnya meliputi internal gear, potensiometer, dan feedback control.

2.4.6 Buzzer

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan loudspeaker, jadi *buzzer* juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektro magnet. Buzzer biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (Ramadhan, 2020). Berikut pada gambar 2.4 merupakan gambar *Buzzer*.



Gambar 2. 5 Buzzer

2.4.7 USB UART FTDI

USB UART FTDI generasi sebelumnya memerlukan resonator kristal atau keramik eksternal. Sirkuit jam kini telah terintegrasi ke perangkat sehingga tidak diperlukan resonator kristal atau keramik. Namun, jika diinginkan, kristal 12MHz eksternal dapat digunakan sebagai jam sumber. Perangkat USB UART FTDI generasi sebelumnya memerlukan EEPROM eksternal jika perangkat tersebut menggunakan ID Vendor USB (VID), ID Produk (PID), nomor seri, dan string deskripsi produk selain nilai default di perangkat itu sendiri. EEPROM eksternal ini kini telah diintegrasikan ke dalam chip FT232R yang berarti bahwa semua desain memiliki opsi untuk mengubah string deskripsi produk. Area pengguna EEPROM internal tersedia untuk menyimpan data tambahan. EEPROM internal dapat diprogram dalam sirkuit, melalui USB tanpa persyaratan tegangan tambahan.

USB UART FTDI generasi sebelumnya memerlukan dua resistor seri eksternal pada jalur USBDP dan USBDM, dan resistor pull up 1,5 k Ω pada USBDP. Ketiga resistor ini kini telah terintegrasi ke dalam perangkat. Perangkat USB UART FTDI generasi sebelumnya memiliki pin AVCC terpisah – Nomor Izin: FTDI#132 pasokan ke PLL internal. Pin ini memerlukan filter RC eksternal. Pin AVCC yang terpisah sekarang dihubungkan secara internal ke VCC, dan filter kini telah diintegrasikan ke dalam chip (Fadillah H.R. dkk, 2023)



Gambar 2. 6 USB UART FTDI

2.4.8 Arduino IDE

Arduino Software (IDE), IDE itu merupakan kependekan dari *Integrated Development Environment*, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. disebut sebagai lingkungan karena melalui *software* inilah *Arduino* dilakukan pemrograman untuk melakukan fungsi-fungsi yang dibenamkan melalui sintaks pemrograman. *Arduino* menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman *Arduino (Sketch)* sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler *Arduino* telah ditanamkan suatu program bernama *Bootlader* yang berfungsi sebagai penengah antara *compiler* *Arduino* dengan mikrokontroler. *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*. Pada *Software Arduino IDE*, terdapat semacam *message box* berwarna hitam yang berfungsi menampilkan status, seperti pesan *error*, *compile*, dan *upload* program.

Di bagian bawah paling kanan *Software Arduino IDE*, menunjukkan board yang terkonfigurasi beserta COM Ports yang digunakan (Safitri, 2019:31). Berikut pada gambar 2.7 merupakan gambar *Arduino IDE*.



Gambar 2. 7 *Arduino IDE*

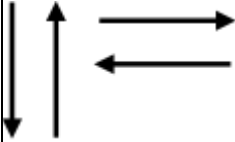
- a. *Icon* menu *verify* yang bergambar ceklis berfungsi untuk mengecek program yang ditulis apakah ada yang salah atau *error*.
- b. *Icon* menu *upload* yang bergambar panah ke arah kanan berfungsi untuk memuat/transfer program yang dibuat di *software* arduino ke hardware *arduino*.
- c. *Icon* menu *New* yang bergambar sehelai kertas berfungsi untuk membuat halaman baru dalam pemrograman.
- d. *Icon* menu *Open* yang bergambar panah ke arah atas berfungsi untuk membuka program yang disimpan atau membuka program yang sudah dibuat dari pabrikan *software* arduino.
- e. *Icon* menu *Save* yang bergambar panah ke arah bawah berfungsi untuk menyimpan program yang telah dibuat atau dimodifikasi.
- f. *Icon* menu *Serial Monitor* yang bergambar kaca pembesar berfungsi untuk mengirim atau menampilkan serial komunikasi data saat dikirim dari hardware arduino



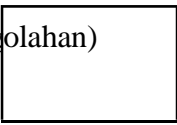
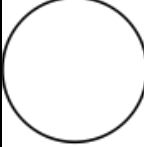
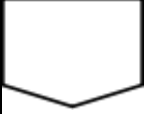


2.4.9 Flowchart







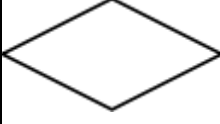
Flowchart (Diagram Alir) atau di sebut *Flowchart* merupakan bagan (*Chart*) yang mengarahkan alir (*flow*) di dalam prosedur atau program sistem secara logika. *Flowchart* adalah cara untuk menjelaskan tahap-tahap pemecahan masalah dengan merepresentasikan simbol-simbol tertentu yang mudah dipahami, mudah digunakan dan standar. Tujuan penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai dan rapi dengan menggunakan simbol-simbol (Syamsiah, 2019:87).

Flowchart atau sering disebut dengan diagram alir merupakan suatu jenis diagram yang merepresentasikan algoritma atau langkah-langkah instruksi yang berurutan dalam sistem. seorang analis sistem menggunakan *flowchart* sebagai bukti dokumentasi untuk menjelaskan gambaran logis sebuah sistem yang akan dibangun kepada programmer. Dengan begitu, *flowchart* dapat membantu untuk memberikan solusi terhadap masalah yang bisa saja terjadi dalam membangun sistem. Pada dasarnya, *flowchart* digambarkan dengan menggunakan simbol-simbol. Setiap simbol mewakili suatu proses tertentu. Sedangkan untuk menghubungkan satu proses ke proses selanjutnya digambarkan dengan menggunakan garis penghubung (Rosaly & Prasetyo, 2019). Berikut pada Tabel 2.2 merupakan tabel *Flowchart*.

Tabel 2. 1 Flowchart

No.	Simbol	Keterangan
1.	 <p><i>Symbol</i></p>	Untuk menghubungkan antara simbol yang satu dengan simbol yang lain atau menyata kan jalannya arus dalam suatu proses.

No	Simbol	Keterangan
2.	<i>Terminal</i> (mulai atau berhenti) 	Simbol ini digunakan untuk menunjukkan awal kegiatan (<i>start</i>) atau akhir dari suatu kegiatan (<i>stop</i>).
3.	<i>Input dan Output</i> 	Untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
4.	Proses (Pengolahan) 	Untuk menunjukkan pengolahan yang dilakukan oleh komputer atau PC.
5.	<i>Connector</i> 	Simbol suatu keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang sama.
6.	<i>Offline Connector</i> 	Simbol untuk keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang berbeda.
7.	<i>Document</i> 	Untuk menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas.
8.	<i>Manual Input</i> 	Berfungsi untuk pemasukan data secara manual <i>on-line keyboard</i> .

No.	Simbol	Keterangan
9.	<p><i>Preparation</i></p> 	Berfungsi untuk mempersiapkan penyimpanan yang sedang/ akan digunakan sebagai tempat pengolahan didalam <i>storage</i> .
10.	<p><i>Manual Operation</i></p> 	Berfungsi untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer atau PC.
11.	<p><i>Multiple Document</i></p> 	Sama seperti symbol document ,hanya saja dokumen yang digunakan lebih dari satu dalam simbol ini.
12.	<p><i>Disk Storage</i></p> 	Untuk menyatakan input yang berasal dari <i>disk</i> atau disimpan ke <i>disk</i> .
13.	<p><i>Magnetic Disk</i></p> 	Untuk <i>input</i> atau <i>output</i> yang menggunakan disk magnetic.
14.	<p><i>Predefined</i></p> 	Untuk pelaksanaan suatu bagian (subprogram) / prosedur.
15.	<p><i>Decision</i> (Keputusan)</p> 	Menunjukkan suatu perbandingan yang harus dibuat bila hasilnya “ya”, maka alir data menunjukkan ke suatu tempat, bila “tidak” maka akan menuju ke tempat lain.