

BAB II

LANDASAN TEORI

2.1 Pengertian Internet of Things

IoT adalah sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus menerus, berikut kemampuan kontrol, berbagi data dan sebagainya. Bahan pangan, elektronik, koleksi, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor tertanam dan selalu ON (Sugiono, 2017). Jadi Internet of Things adalah sebuah konsep dimana suatu objek yang memiliki kemampuan untuk mengirim data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer.

2.2 Mikrokontroller

Mikrokontroller disebut juga MCU (Microcontroller Unit) adalah sebuah sistem komputer yang seluruh atau sebagian besar elemennya dikemas dalam satu chip IC (Integrated Circuit) yang memiliki beberapa sifat seperti komputer yaitu CPU (Central Processing Unit) atau unit pemrosesan terpusat, secara teknis sistem komputer yang mempunyai satu atau beberapa tugas yang sangat spesifik. Seperti pengambilan kode dari memori, operasi aritmatika, dan penyimpanan data kedalam memori, I/O (port untuk input dan output) yang saling bekerja sama (Anna Nur, 2010).

1. Karakteristik Mikrokontroler

- a. Memiliki program khusus yang disimpan didalam memori untuk aplikasi tertentu, tidak seperti PC yang multifungsi karena mudahnya masukan program. Program mikrokontroler relatif lebih kecil dari pada program-program pada PC.
- b. Unit I/O yang sederhana, misalnya LCD, LED, Latch.

- c. Rangkaiannya lebih sederhana dan kompak.
- d. Konsumsi daya lebih kecil.

2. Klasifikasi Mikrokontroler

Mikrokontroler memiliki beberapa klasifikasi sebagai berikut:

- a. ROM (Flash Memory) dengan kapasitas 1024 byte (1KB).
- b. RAM berkapasitas 68 byte.
- c. EEPROM (memori data) berkapasitas 64 byte.
- d. Total 13 jalur I/O (Port B 8 bit).
- e. Timer/Counter bit dengan prescaler.
- f. Fasilitas pemrograman di dalam sistem (ISCP = In Circuit Serial Programming).

3. Fitur-Fitur Mikrokontroler

Ada beberapa fitur yang pada umumnya ada dalam mikrokontroler adalah sebagai berikut:

a. RAM

RAM digunakan oleh mikrokontroler untuk tempat penyimpanan variable. Memori ini bersifat volatile yang artinya akan kehilangan semua data nya jika tidak mendapatkan catu daya.

b. ROM

ROM disebut sebagai kode memori karena berfungsi untuk tempat penyimpanan program yang akan diberikan oleh user.

c. Pin Input dan Output Pin

Pin input adalah bagian yang berfungsi sebagai penerima signal dari luar dan pin ini dihubungkan ke berbagai media inputan seperti keypad, sensor, keyboard, dan sebagainya.

2.2.1 Arduino

Dalam situs resmi Arduino, Arduino didefinisikan sebagai sebuah platform elektronik yang open source, berbasis pada software dan hardware yang fleksibel dan mudah digunakan, yang ditujukan pada setiap orang yang tertarik dalam membuat suatu objek atau lingkungan yang interaktif (Arduino, 2015).

2.2.2 Mikrokontroller Arduino Uno

Arduino UNO merupakan *board* yang menggunakan IC Atmega328. Arduino ini memiliki pin 14 digital input/output (6 pin dapat digunakan sebagai output PWM), 6 analog *input*, frekuensi clock 16 MHz, koneksi USB, *jack power*, ICSP *header*, dan tombol *reset*. Untuk menggunakannya tinggal melakukan koneksi USB dengan komputer atau menggunakan adaptor AC ke DC. Tabel 2.1 merupakan karakteristik dari Arduino UNO.

Tabel 2. 1 Karakteristik Arduino UNO

<i>Microcontroller</i>	Atmega328P
<i>Operating Voltage</i>	5V
<i>Input Voltage (recommended)</i>	7-12V
<i>Input Voltage (limit)</i>	6-20V
<i>Digital I/O Pins</i>	14 (<i>of which 6 provide PWM output</i>)
<i>PWM Digital I/O Pins</i>	6
<i>Analog Input Pins</i>	6
<i>DC Current per I/O Pin</i>	20 mA
<i>DC Current for 3.3V Pin</i>	50 mA
<i>Flash Memory</i>	32 KB (Atmega328P) <i>of which 0.5 KB used by bootloader</i>
<i>EEPROM</i>	1 KB (Atmega328P)
<i>Clock Speed</i>	16 MHz

Pemilihan *board* mikrokontroler dalam penelitian ini memiliki beberapa alasan yaitu:

1. kemudahan dalam mengaksesnya karena sudah dilengkapi IDE dari Arduino.
2. murah dan fleksibel.
3. bersifat *open source* sehingga banyak *developer* yang membuat *library* pemrogramannya..



Gambar 2.1 Arduino Board

2.2.3 Arduino IDE (Integrated Development Environment)

Untuk memprogram Arduino menggunakan pemrograman Arduino IDE. IDE adalah sebuah *software* untuk menulis program, meng-compile menjadi kode biner dan meng-*upload* ke dalam memori mikrokontroler. Arduino berevolusi menjadi sebuah *platform* karena ia menjadi pilihan dan acuan bagi banyak praktisi. Salah satu yang membuat Arduino memikat hati banyak orang adalah karena sifatnya yang *open source*, baik untuk *hardware* maupun *software* nya. Desain *hardware*, bahasa pemrograman dan IDE Arduino yang berkualitas tinggi dan sangat berkelas mestinya menjadi hal yang dapat diandalkan dalam dunia IT. Saat ini *software* Arduino yang digunakan adalah *driver* dan IDE, walaupun masih ada beberapa *software* lain yang sangat berguna selama pengembangan Arduino



Gambar 2.2 *Software* Arduino IDE

2.2.4 Bahasa Pemrograman C/C++

C adalah Bahasa pemrograman yang dikembangkan oleh Dennis Ritchie pada tahun 1970 untuk pemakaian pada sistem operasi UNIX. Hingga sekarang, bahasa ini telah dipergunakan secara praktis pada hampir semua sistem operasi. Bahasa C merupakan Bahasa yang paling populer untuk menulis sistem software dan aplikasi serta dalam pelajaran komputer sains (Hendra, 2011). Fasilitas dari Bahasa C yang perlu menjadi perhatian programmer adalah kemampuannya dalam mengatur isi memori komputer. Bahasa C standar tidak menyediakan fasilitas array bounds checking yang dengan mudah akan menyebabkan bug dalam kaitannya dengan operasi memori, seperti buffer overflows, serta computer insecurity.

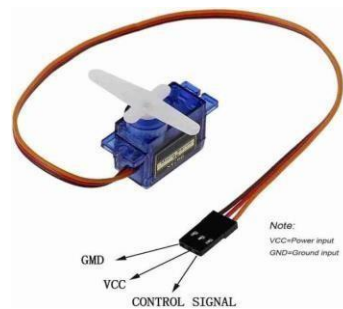
Beberapa fitur bahasa pemrograman C antara lain:

1. Suatu bahasa dengan inti yang sederhana, dimana fungsi- fungsi yang kurang penting tersedia sebagai kumpulan *library* yang distandarisasi.
2. Terfokus pada paradigma pemrograman prosedural, dengan fasilitas pemrograman yang terstruktur.
3. Memiliki bahasa *preprocessor*.
4. Akses secara low-level pada memori komputer melalui pointer.
5. Parameter selalu dilewatkan ke *function* secara *by value*, bukan *by reference*.

2.2.5 Servo Motor

Motor servo adalah sebuah perangkat sebagai aktuator putar yang disebut motor DC, motor ini terdiri dari sebuah, serangkaian gear, potensiometer, Serangkaian gear yang terpasang pada poros motor DC akan memperlambat putaran poros dan meningkatkan torsi motor servo, sedangkan potensiometer dengan perubahan resistansinya saat motor berputar berfungsi sebagai penentu batas posisi putaran poros motor servo.

rangkaian kontrol yang dirancang dengan sistem kontrol umpan balik loop tertutup (servo), sehingga dapat di set-up atau di atur untuk menentukan dan memastikan posisi sudut dari poros output motor. Motor servo biasa digunakan dalam aplikasi-aplikasi di industri, dan lain sebagainya.



Gambar 2.3 Servo Motor

Karakteristik Motor Servo:

- a. Tidak bergetar dan tidak ber-resonansi saat beroperasi.
- b. Daya yang dihasilkan sebanding dengan ukuran dan berat motor.
- c. Penggunaan arus listrik sebanding dengan beban yang diberikan.
- d. Resolusi dan akurasi dapat diubah dengan mengganti *encoder*.
- e. Tidak berisik saat beroperasi dengan kecepatan tinggi.

Motor servo dapat dimanfaatkan salah satunya sebagai penggerak wiper. Motor servo dipilih sebagai penggerak pada wiper karena motor servo memiliki tenaga atau torsi yang besar, sehingga dapat menggerakkan wiper dengan beban yang cukup berat.

2.2.6 Module Joystick

Modul *joystick* seperti Gambar 2.4 ini sama seperti *joystick* analog pada *gamepad*.

Ini dibuat dengan memasang 2 buah potensiometer dengan sudut 90°. Potensio terhubung dengan *stick* pendek yang berpusat pada pegas. Modul ini menghasilkan output sekitar 2.5 *volt* dari X dan Y bila dalam posisi ditengah. Menggerakan joystick akan menyebabkan output yang bervariasi dari 0 *volt* hingga 5 *volt* tergantung pada arah yang ditentukan. Jika modul dihubungkan dengan mikrokontroler maka dapat dilihat nilainya sebesar 512 bila berada pada posisi di tengah. Bila *joystick* digerakkan maka nilainya akan berubah antara 0-1023 tergantung pada posisinya.



Gambar 2.4 Module Joystick

2.3 Konsep Pengembangan Sistem

Pengembangan system (development system) dapat berarti menyusun suatu system baru untuk menggantikan system yang lama secara keseluruhan atau memperbaiki system yang lama. Sistem yang lama perlu diperbaiki atau diganti disebabkan karena beberapa hal (Jogiyanto, 2006). Pada umumnya proses pengembangan system sederhana di organisasi mengikuti pendekatan pemecahan masalah. Pendekatan tersebut biasanya terdiri dari beberapa langkah problem-solving yang umum, yaitu (Whitten, 2005):

1. Mengidentifikasi masalah,
2. Menganalisis dan memahami masalah,
3. Mengidentifikasi solusi yang diharapkan,
4. Mengidentifikasi solusi alternative dan memilih tindakan yang terbaik,
5. Mendesain solusi yang dipilih,
6. Mengimplementasikan solusi yang dipilih,
7. Mengevaluasi hasil (jika masalah tidak terpecahkan, kembali ke langkah satu atau dua seperlunya).

2.4 Perencanaan Mikrokontroler untuk *Internet of Things*

2.4.1 Alur Kerja Mikrokontroler

Penulis membagi menjadi dua alur kerja, yaitu pengembangan Hardware dengan melibatkan servo dan pengembangan Hardware tanpa melibatkan mikrokontroler.

a. Pengembangan Hardware dengan melibatkan motor servo

Penelitian ini menggunakan metodologi penelitian *Rapid Application Development* dengan 4 tahapan alur, *Requirement Planning Phase*, *Design Phase*, *Construction Phase* dan *Cutover*. Penulis mengidentifikasi pengembangan elemen Hardware dilakukan pada saat *Construction Phase*, dengan tahapan sebagai berikut:

1. Pemrograman

Pada tahapan ini, rancangan diterjemahkan menjadi suatu rangkaian modul Arduino dan kode program. Bahasa pemrograman C digunakan untuk memprogram mikrokontroler Arduino Uno dan software editor yang digunakan adalah Arduino IDE.

2. Pengujian (*Testing*)

Pada tahap ini dilakukan pengujian masing-masing modul atau unit program guna mengetahui apakah modul-modul tersebut bekerja sesuai dengan tugasnya.

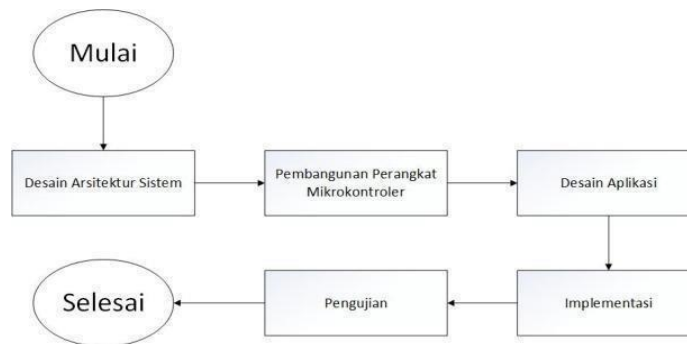
3. Menjalankan Aplikasi

Pada langkah ini, user membuka source code Arduino di dalam Arduino IDE. Menyelesaikan rangkaian Arduino dan mengkoneksikan Arduino ke motor servo untuk menjalankan aplikasi.

4. Tahap implementasi

Penganalisis merancang aspek-aspek teknis dan non-teknis dari workshop. Setelah aspek-aspek ini disepakati, sistem telah dibangun dan diperbaiki, maka sistem atau bagian dari sistem yang baru akan diuji coba lalu diperkenalkan ke organisasi. Pada penelitian Sadewa, Sujaini, Nyoto & Dwi (2015),

tahapan yang berkaitan dengan pembangunan Hardware dilakukan pada tahapan Pembangunan Perangkat Mikrokontroler, Implementasi dan Pengujian. Secara keseluruhan, metodologi penelitian yang dilakukan adalah sebagai berikut:



Gambar 2. 5 Metodologi Penelitian

b. Pengembangan Hardware yang tidak melibatkan mikrokontroler Langkah ini dimulai dari pembuatan project pada *stand* papan tulis sebagai berikut:

1. Material yang digunakan

Material komponen merupakan faktor pendukung penting dalam keseluruhan pembuatan benda. Material tersebut merupakan sumber kekuatan, kemudahan produksi, tingkat estetika, dan faktor lainnya yang berpengaruh. Oleh karena itu dengan penggunaan material benda yang tepat, suatu komponen tersebut dapat memiliki beberapa keunggulan yang dapat mempermudah maupun mempersulit proses ini. Oleh karena itu harus dipertimbangkan juga dari material apa benda akan dibuat.

2. *Machining Process*

Machining yang digunakan banyak jenisnya mulai dari pemotongan, pembubutan, pengeboran, dan proses lainnya yang memiliki beragam efek pada hasil komponen. Keseluruhan proses tersebut dapat menghasilkan komponen yang sudah jadi maupun komponen yang butuh beberapa pengerjaan lagi yang selanjutnya akan diolah kembali. *Machining* yang dilakukan juga harus mempertimbangkan aspek assembly agar tidak malah membebani proses tersebut.

3. Proses penyatuan material Penyatuan komponen-komponen banyak macamnya, tentunya juga sangat berpengaruh pada tingkat kekuatan hingga estetika dari benda yang akan diassembly. Penyatuan

4. Toleransi

Seluruh proses yang dilakukan dalam proses pembuatan benda tentu saja memiliki toleransi yang harus tercapai. Toleransi sendiri adalah batasan ketidak sesuaian yang dapat diterima untuk mencapai benda yang diinginkan. Proses yang sebelumnya telah dilakukan harus memenuhi toleransi yng ditentukan. Karena jika tidak sesuai toleransi maka pemasangan material tidak dapat dilakukan.

5. Proses *finishing*

Finishing merupakan proses terakhir yang bertujuan untuk menyempurnakan komponen- komponen yang telah dibuat. Proses-proses sebelumnya sebaik apapun itu pasti akan meninggalkan sisa kecacatan pada komponen yang telah jadi, sehingga proses inilah yang kemudian akan memperbaikinya sehingga komponen sudah siap *assembly*.

6. *Quality Control*

Kualitas dari assembly juga harus diamati dengan baik. Dengan diadakannya kontrol terhadap kualitas, maka benda yang telah dibuat akan dicek kesesuaiannya dengan kualitas yang diinginkan. Selain pada bagian assembly, quality control juga berfungsi untuk mengetahui kecacatan pada proses sebelumnya sehingga untuk kedepannya dapat dihindari kesalahan tersebut.