

BAB II

LANDASAN TEORI

Untuk mendukung keberhasilan penelitian, penulis melakukan pendekatan teoritis melalui beberapa literatur yang berhubungan dengan penelitian yang dilakukan. Beberapa tinjauan pustaka pada penelitian ini antara lain.

II.1. Penelitian Terkait

Penelitian terdahulu sangat penting guna menemukan titik perbedaan maupun persamaan dengan penelitian yang akan dilakukan. Selain itu, penelitian terdahulu juga berguna sebagai perbandingan sekaligus landasan dalam penelitian tersebut. Penelitian robot bawah air pernah dilakukan oleh Yuh (1990), robot bawah yang diteliti adalah robot bawah air jenis pengendali dari jarak jauh (*Remotely Operated Vehicles*). Robot bawah air diperlukan guna menjadi alat yang penting untuk mengeksplorasi rahasia kehidupan bawah laut. Yuh meneliti tentang model dinamis yang dapat diterapkan untuk robot bawah air, hal ini mutlak diperlukan karena berbagai hambatan yang dialami oleh robot yang bawah air meliputi cara mengatasi kepadatan bawah air yang tinggi, air laut yang terstruktur dan memiliki kondisi lingkungan yang bersifat *non-uniform*. Berbagai masalah tersebut menyebabkan respon robot bawah air terlampau sulit dan membuat tingkat otonomi robot juga sulit untuk dicapai. Penelitian tersebut telah menyajikan model dinamis dan system control adaptif untuk robot bawah air jenis *Remotely Operated Vehicles (ROVs)*. Model dinamis pada ROVs dijelaskan menggunakan enam rangkaian *non-linear*

yang dipasang dalam kendaraan berbentuk perangkat hidrodinamik. Melalui komponen tersebut, robot bawah air dalam uji cobanya tidak mengalami efek pergerakan fluida oleh gelombang air.

Penelitian juga dilakukan oleh Wettergreen, *et al* (1999) tentang robot bawah air. Robot bawah air memerlukan bimbingan dan *control* yang memadai untuk melakukan tugas-tugas yang berguna selama berada di bawah air. Metode yang dapat digunakan untuk membimbing robot bawah air adalah control servo secara visual. Untuk mengkoordinasikan dan mengontrol *thruster*, model yang kompleks serta mengontrol skema robot dapat digantikan menggunakan pendekatan *connectionist learning*, yaitu penguatan pembelajaran menggunakan sinyal *reward* dan memperbanyak interaksi dengan lingkungan untuk membentuk suatu perilaku yang diharapkan. Hal tersebut dilakukan dengan menggabungkan bimbingan berbasis visi menggunakan *neuro-controller* yang dilatih oleh penguat pembelajaran. Tujuan dilakukannya penelitian tersebut adalah untuk mengaktifkan robot bawah air guna dapat melakukan pemberhentian pada batu- batu karang yang ada didalam laut. Hasil penelitian tersebut adalah fitur metode pelacakan dan skema untuk melakukan bimbingan berbasis *control servo* secara visual. Selain itu, *neuro-controller* yang menjadi tambahan pada robot membantu mengatasi permasalahan yang selama ini terjadi pada robot bawah air berjenis *Autonomous Underwater Vehicles* (AUVs).

Penelitian Bokser (2003) menjabarkan mengenai robot kecil bawah air yang dirancang khusus untuk percobaan menggunakan jaringan sensor aktuator. Robot tersebut didasarkan pada platform *mote* yang digunakan secara ekstensif dalam

sensor jaringan komunitas sebagai testbed eksperimental. Tes awal untuk pengaturan dalaman robot dalam air dan pengaturan suhu bawah air akan dianalisis. Pada tahap pengujian, serangkaian tes dilakukan untuk pembuktian platform. Pengujian dengan mengukur tekanan dan kedalaman menghasilkan bukti linear, yaitu tekanan dan kedalaman air ternyata berbanding lurus dalam mempengaruhi kinerja robot. Perbaikan lain yang diperlukan oleh robot bawah air adalah cara untuk memperpanjang daya tahan baterai robot. Baterai sangat mempengaruhi kualitas komunikasi antara *system* robot dengan *system* minimum dalam pengontrol. Selain itu, teknik sederhana dalam manajemen daya akan dapat mengubah kekuatan penggunaan motor robot. Penelitian lanjutan adalah mengembangkan fungsi ALITY pada pergerakan robot bawah air, dan mewujudkan jaringan tesbed pada robot.

Adrianyah (2008) meneliti bahwa perancangan robot harus mengatasi beberapa masalah. Kompleksitas dan ketidaktahuan yang sempurna tentang lingkungan dan situasi yang akan dihadapi, kemampuan untuk mempresepsikan keadaan berdasarkan informasi yang didapati dari sensor yang terkadang tidak akurat, dan kemampuan mengambil keputusan tentang pergerakannya dalam waktu yang terbatas adalah contohnya. Di sisi lain, kebutuhan terhadap penelitian bawah air cukup tinggi. Oleh karena itu, penelitian tersebut berupaya untuk membahas perancangan robot bawah air yang lebih baik demi mengatasi masalah-masalah yang disebutkan. Perancangan tersebut difokuskan kepada kemampuan robot untuk bermanuver dan bergerak dengan efektif di bawah air. Agar manuver dan pergerakan robot efektif, maka diajukan suatu arsitektur pengendali tertentu, yaitu

arsitektur pengendali robot berbasis perilaku. Beberapa eksperimen akan dilakukan untuk menguji efektifitas manuver dan pergerakan robot bawah air tersebut. Metode pengendali berbasis perilaku adalah pendekatan yang diinspirasi oleh struktur umum sistem biologi makhluk hidup, yaitu sistem yang berasaskan pada falsafah dari bawah ke atas (*bottom-up philosophy*), dimana tugas-tugas robot dibagi ke dalam beberapa kerja-kerja kecil yang harus diselesaikan oleh robot pada satu waktu tertentu. Kerja-kerja kecil itu disebut dengan perilaku (*behavior*). Prinsip-prinsip penting arsitektur pengendali berbasis perilaku yang membuatnya menjadi alternatif dibanding arsitektur pengendali lainnya adalah: *parallelism*, *modularity*, *situatedness*, *embodiment*, dan *emergence*. Beberapa behavior yang biasanya terdapat pada sebuah robot antara lain adalah: menuju target (*goal seeking*), mengikut dinding (*wall following*), menghindari rintangan (*obstacle avoiding*) dan kembali ke tempat semula (*homing*). Hasil akhir penelitian tersebut adalah hasil simulasi robot bawah air yang menunjukkan bahwa pergerakan robot, baik pergerakan dasar maupun kombinasinya memiliki karakteristik yang andal.

McGrath, *et al* (2008) dalam penelitiannya tentang robot bawah air LEGO mengemukakan bahwa salah satu cara mudah untuk membangun robot bawah air untuk siswa dan mahasiswa adalah menggunakan *LEGO Mindstorms kit*, yaitu seperangkat komponen robot yang dapat dengan mudah untuk dibongkar dan dipasang kembali. Perancangan yang sederhana dan pemrograman NXT yang mudah dimengerti akan membuat siswa yang merancang robot dapat melakukan eksplorasi lebih jauh. Robot dapat diimplementasikan dalam bidang matematika, konsep fisika apung, hukum newton, momentum, kepadatan dan lain sebagainya.

Putri (2012) melakukan penelitian tentang robot bawah air untuk pemetaan dengan nama STROMS. STROMS adalah robot yang dirancang untuk bisa bekerja di bawah air guna pemetaan bawah laut. Cara kerjanya hampir sama seperti robot darat pada umumnya. Sedangkan dari sisi desainnya, robot ini memiliki tingkat kerumitan yang lebih tinggi daripada robot darat. Hal ini disebabkan karena STROMS didesain untuk mampu menyelam sehingga harus memperhitungkan keseimbangan robot bila dioperasikan di dalam air. STROMS memiliki 7 sensor antara lain: 4 sensor jarak, 1 sensor pH, 1 sensor tekanan, 1 sensor suhu. Robot ini juga memiliki 1 buah kamera yang berguna untuk pemetaan bawah laut secara visual. Seperti yang dijelaskan pada subbab sebelumnya, pemetaan yang dilakukan oleh STROMS tidak hanya pemetaan secara visual saja tetapi juga pemetaan terhadap pH, tekanan, dan suhu. Desain STROMS dibuat mirip dengan ikan, tentu saja robot ini memiliki sirip yang berguna untuk mempermudah gerak robot.

II.2. Teori Pendukung

II.2.1. Perancangan

Perancangan adalah usulan pokok yang mengubah sesuatu yang sudah ada menjadi sesuatu yang lebih baik, melalui tiga proses: mengidentifikasi masalah-masalah, mengidentifikasi metode untuk pemecahan masalah, dan pelaksanaan pemecahan masalah. Dengan kata lain adalah pemograman, penyusunan rancangan dan pelaksanaan rancangan (John Wade, 1997). Dan masih banyak pendapat dari para ahli yang lain seperti J.C Jones, E. Marchet, JB. Reswick dan masih banyak pendapat dari para ilmuwan yang lain. Dan yang terakhir yaitu perancang.

Perancang adalah orang yang merancang suatu ide-ide kreatif yang dapat diwujudkan melalui visual dengan konsep-konsep ideal. (Ahmad Rifai, 2013).

II.2.2. Implementasi

Implementasi adalah suatu tindakan atau pelaksanaan dari sebuah rencana yang sudah disusun secara matang dan terperinci. Secara sederhana implementasi bisa diartikan pelaksanaan atau penerapan. Majone dan Wildavsky mengemukakan implementasi sebagai evaluasi. Browne dan Wildavsky mengemukakan bahwa implementasi adalah perluasan aktivitas yang saling menyesuaikan. Pengertian implementasi sebagai aktivitas menyesuaikan juga di kemukakan oleh Mclaughlin. Adapun Schubert mengemukakan bahwa implementasi adalah sistem rekayasa.

II.2.3. ROV (*Remotely Operated Vehicle*)

ROV (*Remotely Operated Vehicle*) merupakan salah satu jenis robot bawah air yang tergolong ke dalam tipe robot mobile dengan pengaplikasian ditujukan untuk melakukan kegiatan bawah air. Dalam pengoperasiannya ROV dapat dikendalikan oleh operator karena didukung oleh perangkat kendali (*remote control*). Menurut Mooney, Jr., et al (1996:8), ROV yang kosong merupakan kapal selam mini dengan pengikat kabel inti yang membawa daya listrik, data sensor, dan perintah kontrol. Pilot atau operator bekerja dari tempat yang jauh tetapi mempertahankan kendali *vehicle* (ROV itu sendiri). ROV digunakan terutama dalam operasi minyak dan gas lepas pantai untuk berbagai pemeriksaan dan tugas manipulasi, serta telah banyak menggantikan para penyelam di berbagai pekerjaan industri. ROV juga banyak digunakan untuk peletakkan kabel bawah laut, dan

pengembangan lepas pantai serta bergerak ke perairan yang lebih dalam, ROV akan semakin dibutuhkan.

Christ dan Wernli (2007:2) secara sederhana menyatakan, sebuah ROV adalah kamera yang dipasang di kerangka atau bodi yang tahan air, dengan pendorong untuk bermanuver, yang terhubung pada kabel ke atas permukaan di mana sinyal video ditransmisikan.

ROV adalah wahana bawah air yang bertenaga listrik dan dikontrol melalui pusat, dapat bermanuver sesuai perintah manusia dengan pendorong (*thruster*) hidrolis atau elektrik, kamera yang dipasang dalam wadah tahan air, dengan pendorong untuk bermanuver, yang melekat pada kabel ke permukaan dimana sinyal video yang dikirim (H.A, Kusuma, 2012).

II.2.4. Android

Android adalah sebuah sistem operasi untuk *smartphone* dan *tablet*. Sistem operasi dapat diilustrasikan sebagai jembatan atau penghubung antar piranti (*device*) dan penggunanya, sehingga pengguna dapat berinteraksi dengan *device*-nya dan menjalankan aplikasi-aplikasi yang tersedia pada *device*. Android merupakan sistem operasi yang bersifat *open source*. Disebut *open source* karena *source code* dari sistem operasi Android dapat dilihat, di-*download* dan dimodifikasi secara bebas. Paradigma *open source* ini memudahkan pengembangan teknologi Android, karena semua pihak yang tertarik dapat memberikan kontribusi, baik pada pengembangan sistem operasi maupun aplikasi. (Joni Maulindar et al, 2016)

Menurut Arifianto Teguh, android adalah sebuah *platform* pertama yang betul-betul terbuka dalam pengembangannya dan komprehensif untuk perangkat *mobile*, semua perangkat lunak yang ada difungsikan menjalankan sebuah *device mobile* tanpa memikirkan kendala kepemilikan yang menghambat inovasi pada teknologi *mobile*. Dalam definisi lain, Android merupakan subset perangkat lunak untuk perangkat *mobile* yang meliputi sistem operasi, *middleware*, dan aplikasi inti yang dirilis oleh Google. Sedangkan Android *SDK* (*Software Development Kit*) menyediakan *tools* dan API yang diperlukan untuk mengembangkan aplikasi pada *platform* Android dengan menggunakan bahasa pemrograman Java.

Aplikasi *Android* ditulis dalam bahasa pemrograman *java*, yaitu kode *java* yang terkompilasi bersama-sama dengan data dan *file-file* sumber yang dibutuhkan oleh aplikasi yang digabungkan oleh *app tools* menjadi paket aplikasi Android, sebuah file yang ditandai dengan akhiran *.apk*. file inilah yang didistribusikan sebagai aplikasi dan diinstal pada *handset Android*. File ini diunduh oleh pengguna ke perangkat *mobile* mereka. Semua kode dijadikan satu file *.apk*, dan kemudian kita sebut sebagai sebuah aplikasi. (Ahmad : 2015 : 190-200)

II.2.5. Mikrokontroler

Mikrokontroler adalah suatu *chip* dengan kepadatan yang sangat tinggi, di mana semua bagian yang diperlukan untuk suatu kontroler sudah dikemas dalam satu keping, biasanya terdiri dari CPU (*Central Proccesssing Unit*), RAM (*Random Access Memory*), EEPROM/ EPROM/ PROM/ROM, I/O, *Timer* dan lain sebagainya. Mikrokontroler AVR adalah mikrokontroler RISC 8 bit berdasarkan arsitektur *Harvard*, yang dibuat oleh Atmel pada tahun 1996. AVR memiliki

keunggulan dibandingkan dengan mikrokontroler lain, keunggulan AVR yaitu AVR memiliki kecepatan eksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock*, lebih cepat dibandingkan MCS51 yang membutuhkan 12 siklus *clock* untuk mengeksekusi 1 instruksi. Mikrokontroler yang digunakan dalam penelitian ini adalah mikrokontroler keluarga AVR memiliki fitur yang lengkap (ADC internal, EEPROM internal, *Timer/Counter*, *Watchdog Timer*, PWM, Port I/O, komunikasi serial, Komparator, I2C, dan lain – lain (Ardianto Pranata et al, 2015).

II.2.6. Mikrokontroler ATmega328

Mikrokontroler adalah sebuah sistem komputer fungsional dalam sebuah chip. Di dalamnya terkandung sebuah inti prosesor, memori (sejumlah kecil RAM, memori program, atau keduanya), dan perlengkapan input output (Hanafi ; 2015 : 154). Dengan kata lain, mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data. Sekedar contoh, bayangkan diri anda saat mulai belajar membaca dan menulis, ketika anda sudah bisa melakukan hal itu anda bisa membaca tulisan apapun baik buku, cerpen, artikel dan sebagainya, dan andapun bisa pula menulis halhal sebaliknya. Begitu pula jika anda sudah mahir membaca dan menulis data maka anda dapat membuat program untuk membuat suatu sistem pengaturan otomatis menggunakan mikrokontroler sesuai keinginan anda. 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. (Hanafi ; 2015 : 154)

Mikrokonktroler digunakan dalam produk dan alat yang dikendalikan secara otomatis, seperti sistem kontrol mesin, remote controls, mesin kantor, peralatan rumah tangga, alat berat, dan mainan. Dengan mengurangi ukuran, biaya, dan komsumsi 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. Rancangan 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. Yang dimaksud dengan sistem minimal adalah sebuah rangkaian mikrokontroler yang sudah dapat digunakan untuk menjalankan sebuah aplikasi. Sebuah IC mikrokontroler tidak akan berarti bila hanya berdiri sendiri. Pada dasarnya sebuah

sistem minimal mikrokontroler AVR memiliki prinsip yang sama. (Hanafi ; 2015 : 154)

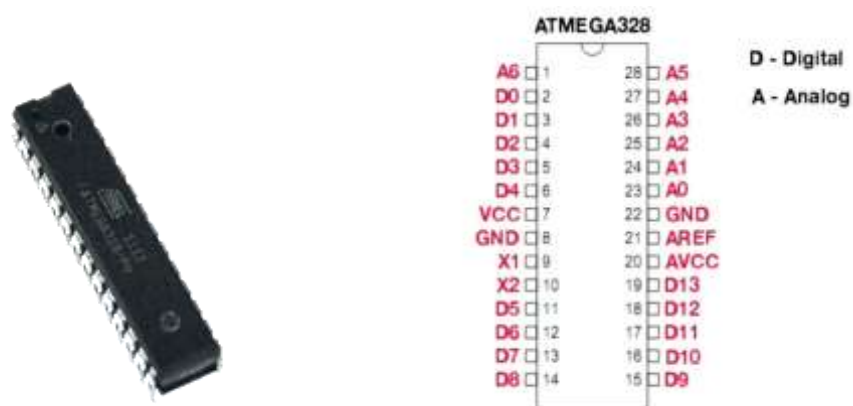
ATMega328 adalah mikrokontroller keluaran dari atmel yang mempunyai arsitektur RISC (*Reduce Instruction Set Computer*) yang dimana setiap proses eksekusi data lebih cepat dari pada arsitektur CISC (*Completed Instruction Set Computer*). Mikrokontroller ini memiliki beberapa fitur antara lain:

1. 130 macam instruksi yang hampir semuanya dieksekusi dalam satu siklus *clock*.
2. 32 x 8-bit register serba guna.
3. Kecepatan mencapai 16MIPS dengan *clock* 16MHz.
4. 32KB flash memory dan pada arduino memiliki *bootloader* yang menggunakan 2KB dari *flash memory* sebagai *bootloader*.
5. Memiliki EEPROM (*Electrically Erasable Programmable Read Only Memory*) sebesar 1KB sebagai tempat penyimpanan data semi permanent karena EEPROM tetap dapat menyimpan data meskipun daya dimatikan.
6. Memiliki SRAM (*Static Random Access*
7. *Memory*) sebesar 2KB
8. Memiliki pin I/O digital sebanyak 14 pin.

Mikrokontroller ATmega 328 memiliki arsitektur Harvard, yaitu memisahkan memori untuk kode program dan memori untuk data sehingga dapat memaksimalkan kerja dan *parallelism*. Instruksi-instruksi dalam memori program dieksekusi dalam satu alur tunggal, dimana pada saat satu instruksi dikerjakan

instruksi berikutnya sudah diambil dari memori program. Konsep inilah yang memungkinkan instruksi – instruksi dapat dieksekusi dalam setiap satu siklus clock. 32 x 8-bit register serba guna digunakan untuk mendukung operasi pada ALU (*Arithmetic Logic unit*) yang dapat dilakukan dalam satu siklus. 6 dari register serbaguna ini dapat digunakan sebagai 3 buah register pointer 16-bit pada mode pengalamatan tidak langsung untuk mengambil data pada ruang memori data. (Hanafi ; 2015 : 155)

Ketiga register pointer 16-bit ini disebut dengan register X (gabungan R26 dan R27), register Y (gabungan R28 dan R29), dan register Z (gabungan R30 dan R31). Hampir semua instruksi AVR memiliki format 16-bit. Setiap alamat memori program terdiri dari instruksi 16-bit atau 32-bit. Selain register serba guna di atas, terdapat register lain yang terpetakan dengan teknik *memory mapped* I/O selebar 64 byte. Beberapa register ini digunakan untuk fungsi khusus antara lain sebagai register control Timer/ Counter, Interupsi, ADC, USART, SPI, EEPROM, dan fungsi I/O lainnya. Register – register ini menempati memori pada alamat 0x20h – 0x5Fh (Hanafi ; 2015 : 155).



Gambar II.1. Mikrokontroler ATmega 328
(*Datasheet ATmega 328, www.atmel.com*)

II.2.7. Dasar Sistem Kendali (Navigasi)

Sistem kendali atau sistem kontrol merupakan suatu sistem yang keluarannya atau outputnya dikendalikan pada suatu nilai tertentu atau untuk merubah beberapa ketentuan yang telah ditetapkan dari masukan atau input ke sistem. Untuk merancang suatu sistem yang dapat merespon perubahan tegangan dan mengeksekusi perintah berdasarkan situasi yang terjadi, maka diperlukan pemahaman tentang sistem kendali (*control system*). Sistem kendali merupakan suatu kondisi dimana sebuah perangkat (*device*) dapat di kontrol sesuai dengan perubahan situasi.

II.2.8. *Software Android Studio*

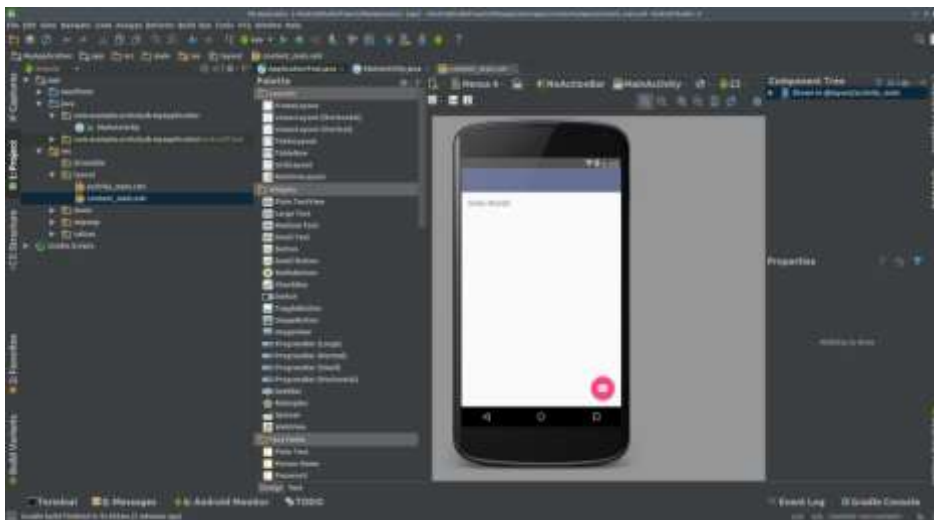
Android Studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi Android dan bersifat *open source* atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 mei 2013 pada *event* Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android.

Android Studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT *plugin* (*Android Development Tools*). Android Studio memiliki fitur :

1. Projek berbasis pada Gradle *Build*.
2. *Refactory* dan pembenahan *bug* yang cepat.
3. *Tools* baru yang bernama “Lint” dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
4. Mendukung *Proguard and App-signing* untuk keamanan.

5. Memiliki GUI aplikasi android lebih mudah.
6. Didukung oleh Google Cloud Platfrom untuk setiap aplikasi yang dikembangkan. (Andi Juansyah ; 2015)

Berikut adalah tampilan dari Android Studio :



Gambar II.2. Android Studio
(www.androidstudio.com)

II.2.9. Arduino

Arduino adalah kit elektronik atau papan rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama, yaitu sebuah *chip* mikrokontroller dengan jenis AVR dari perusahaan Atmel. Pembuatan Arduino dimulai pada tahun 2005, dimana sebuah situs perusahaan komputer Olivetti di Ivrea Italia. Pendiri dari Arduino adaah Massimo Banzi dan David Cuartielles sebagai founder. Saat ini ada bermacam-macam bentuk papan Arduino yang disesuaikan dengan peruntukannya antara lain Arduino USB, Arduino Serial, Arduino Fio, Arduino Lilypad, Arduino BT (Bluetooth), Arduino Nano, Arduino Mini, Arduino Uno R3, Arduino Mega. (Zuhadi, 2017).

Arduino merupakan mikrokontroler yang dirancang untuk bisa digunakan dengan mudah oleh para seniman dan desainer (yang memang bukan orang teknik). Dengan demikian, tanpa mengetahui bahasa pemrograman Arduino bisa digunakan untuk menghasilkan karya yang canggih. Hal ini seperti diungkapkan oleh Mike Schmidt. Menurut Massimo Banzi, salah satu pendiri atau pembuat arduino, Arduino merupakan sebuah platform hardware *open source* yang mempunyai *input/output (I/O)* yang sederhana (Mirfan, 2016 : 127).

Arduino Uno adalah sebuah *board* mikrokontroler yang berbasis ATmega328. Arduino memiliki 14 pin input/output yang mana 6 pin dapat digunakan sebagai output PWM, 6 analog input, *crystal* osilator 16 MHz, koneksi USB, *jack power*, kepala ICSP, dan tombol reset. Arduino mampu *men-support* mikrokontroler, dapat dikoneksikan dengan komputer menggunakan kabel USB. (Hanafi ; 2015 : 156)

Arduino adalah kit elektronik atau papan rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama, yaitu sebuah *chip* mikrokontroller dengan jenis AVR dari perusahaan Atmel. Pembuatan Arduino dimulai pada tahun 2005, dimana sebuah situs perusahaan komputer Olivetti di Ivrea Italia. Pendiri dari Arduino adaah Massimo Banzi dan David Cuartielles sebagai founder. Saat ini ada bermacam-macam bentuk papan Arduino yang disesuaikan dengan peruntukannya antara lain Arduino USB, Arduino Serial, Arduino Fio, Arduino Lilypad, Arduino BT (Bluetooth), Arduino Nano, Arduino Mini, Arduino Uno R3, Arduino Mega. (Zuhadi, 2017).

Arduino merupakan mikrokontroler yang dirancang untuk bisa digunakan dengan mudah oleh para seniman dan desainer (yang memang bukan orang teknik). Dengan demikian, tanpa mengetahui bahasa pemrograman Arduino bisa digunakan untuk menghasilkan karya yang canggih. Hal ini seperti diungkapkan oleh Mike Schmidt. Menurut Massimo Banzi, salah satu pendiri atau pembuat Arduino, Arduino merupakan sebuah platform hardware *open source* yang mempunyai *input/output (I/O)* yang sederhana (Mirfan, 2016 : 127).

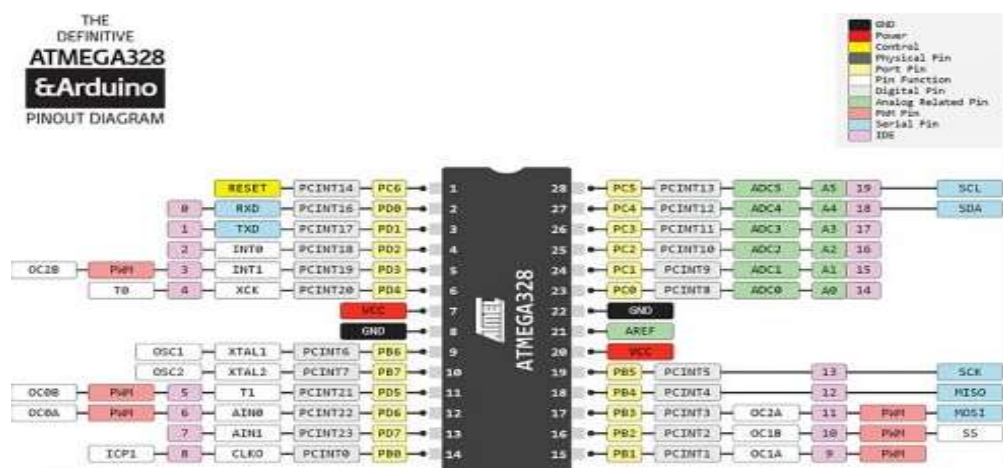


Gambar II.3. Board Arduino Uno ATmega328
(Arduino Board, <http://www.arduino.cc/>)

Arduino memiliki kelebihan tersendiri dibanding *board* mikrokontroler yang lain selain bersifat *open source*, Arduino juga mempunyai bahasa pemrogramannya sendiri yang berupa bahasa C. Selain itu dalam board Arduino sendiri sudah terdapat *loader* yang berupa USB sehingga memudahkan kita ketika kita memprogram mikrokontroler di dalam Arduino. Sedangkan pada kebanyakan board mikrokontroler yang lain yang masih membutuhkan rangkaian loader terpisah untuk memasukkan program ketika kita memprogram mikrokontroler. Port USB tersebut selain untuk loader ketika memprogram, bisa juga difungsikan sebagai port komunikasi serial. (Hanafi ; 2015 : 156)

Arduino menyediakan 20 pin I/O, yang terdiri dari 6 pin input analog dan 14 pin digital input/output. Untuk 6 pin analog sendiri bisa juga difungsikan sebagai output digital jika diperlukan output digital tambahan selain 14 pin yang sudah tersedia. Untuk mengubah pin analog menjadi digital cukup mengubah konfigurasi pin pada program. Dalam board kita bisa lihat pin digital diberi keterangan 0-13, jadi untuk menggunakan pin analog menjadi output digital, pin analog yang pada keterangan board 0-5 kita ubah menjadi pin 14-19. Dengan kata lain pin analog 0-5 berfungsi juga sebagai pin output digital 14-16 (Hanafi ; 2015 : 156).

Sifat *open source* arduino juga banyak memberikan keuntungan tersendiri untuk kita dalam menggunakan *board* ini, karena dengan sifat *open source* komponen yang kita pakai tidak hanya tergantung pada satu merek, namun memungkinkan kita bisa memakai semua komponen yang ada dipasaran.



Gambar II.4. Konfigurasi Pin Arduino Uno

(<https://arduino-info.wikispaces.com/>)

II.2.10. Software Arduino IDE

Sketch IDE (Integrated Development Environment) merupakan perangkat lunak yang dirancang untuk memenuhi penggunaan papan arduino dengan bahasa

pemrograman sendiri. Pemrograman untuk arduino dengan menggunakan perangkat lunak ini akan memudahkan para pengguna karena bahasa pemrograman yang dirancang untuk lebih mudah dimengerti. Selain itu keunggulan pada perangkat lunak ini adalah tersedianya beberapa contoh pemrograman untuk sejumlah perangkat seperti *blink*, *motor servo*, LCD (*Liquid Cristal Display*), sensor ultrasonik dan lain sebagainya. Beberapa program yang telah dibuat pada perangkat lunak ini dapat langsung ditanam pada *chip* mikrokontroller pada papan arduino dengan menggunakan kabel USB (*Universal Serial Bus*) dengan syarat tidak adanya kesalahan dalam penelitian programnya dan sudah terhubung. Istilah menanam program ke *chip* mikrokontroller inilah yang sering disebut dengan proses *upload*. (Supegina & Munandar, 2014 : 123). Berikut merupakan gambaran tampilan arduino IDE :

The image shows a screenshot of the Arduino IDE software. The window title is "Blink | Arduino 1.0". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for opening files, saving, and running. The main text area contains the following code:

```
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 * This example code is in the public domain;
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards;
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);            // wait for a second
  digitalWrite(13, LOW); // set the LED off
  delay(1000);           // wait for a second
}
```

The status bar at the bottom indicates "1" and "Arduino Uno on /dev/ttyACM1".

Gambar II.5. Tampilan Arduino IDE
(Arduino IDE, <http://www.arduino.cc/>)

II.2.11. Bahasa Pemrograman C

Bahasa BCPL yang dikerjakan oleh Martin Richards pada tahun 1967 merupakan awal dari lahirnya bahasa C. Ken Thompson memulai pengembangan bahasa BCPL yaitu bahasa B pada tahun 1970. Perkembangan selanjutnya dari bahasa B dikembangkan menjadi bahasa C oleh Dennis Ritchie beberapa bulan berikutnya di Bell Telephone Laboratories Inc. (sekarang AT&T Bell Laboratories). (Mujtahid Aktanto ; 2016 : 7)

Beberapa alasan mengapa Bahasa C banyak digunakan, diantaranya adalah sebagai berikut :

1. Bahasa C hampir tersedia di semua jenis komputer
2. Bahasa C adalah bahasa yang terstruktur
3. Memiliki dukungan pustaka yang banyak
4. Proses eksekusi program lebih cepat
5. Kode Bahasa C sifatnya adalah portable dan fleksibel
6. Bahasa C hanya menyediakan sedikit kata-kata kunci, hanya terdapat 32 kata kunci. (Mujtahid Aktanto ; 2016 : 7)

Struktur penulisan bahasa C secara umum terdiri atas empat blok, yaitu:

1. *Header*,
2. Deklarasi konstanta global dan atau variabel,
3. Fungsi dan atau prosedur (bisa di bawah program utama),
4. Program utama.

Secara umum, pemrograman C paling sederhana dilakukan dengan hanya menuliskan program utamanya saja, yaitu:

```

/* fungsi utama */
void main()
{
    Statemen-statemen;
}

/*fungsi-fungsi lain yang ditulis oleh pemrogram komputer*/
Fungsi_fingsi_lain()
{
    Statemen-statemen;
}

```

II.2.12. Flowchart

Diagram alir atau *flowchart* adalah gambaran dalam bentuk diagram alir dari algoritma - algoritma dalam suatu program yang menyatakan arah alur program tersebut. Dalam diagram alir ini akan diketahui jalur dari program secara keseluruhan. Diagram alir merupakan suatu model logika data yang dibuat untuk menggambarkan darimana asal data dan tujuan data yang keluar sistem, proses apa yang menghasilkan data tersebut (Endyatna Puthut Bagus Pratama et al ; 2013 : 26 – 27).

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* menolong analis dan programmer untuk memecahkan masalah kedalam segmen segmen yang lebih kecil dan menolong dalam menganalisis alternatif - alternatif lain dalam pengoperasian (Caesar Pats Yahwe et al ; 2016 : 103).

Dalam penulisan *Flowchart* dikenal dua model, yaitu Sistem *Flowchart* dan Program *Flowchart*.

1. *System Flowchart*

Sistem *Flowchart* yaitu bagan yang memperlihatkan urutan *prosedure* dan proses dari beberapa proses di dalam media tertentu. Melalui *Flowchart* ini terlihat jenis media penyimpanan yang dipakai dalam pengolahan data.

- a. Selain itu juga menggambarkan file yang dipakai sebagai input dan output,
- b. Tidak digunakan untuk menggambarkan urutan langkah untuk memecahkan masalah,
- c. Hanya untuk menggambarkan prosedur dalam sistem yang dibentuk,

2. *Program Flowchart*

Program *Flowchart* yaitu bagan yang memperlihatkan urutan dan hubungan proses dalam suatu program. Dua jenis metode penggambaran program *Flowchart*:

- a. *Conceptual Flowchart*, menggambarkan alur pemecahan masalah secara global,
- b. *Detail Flowchart*, menggambarkan alur pemecahan masalah secara rinci.