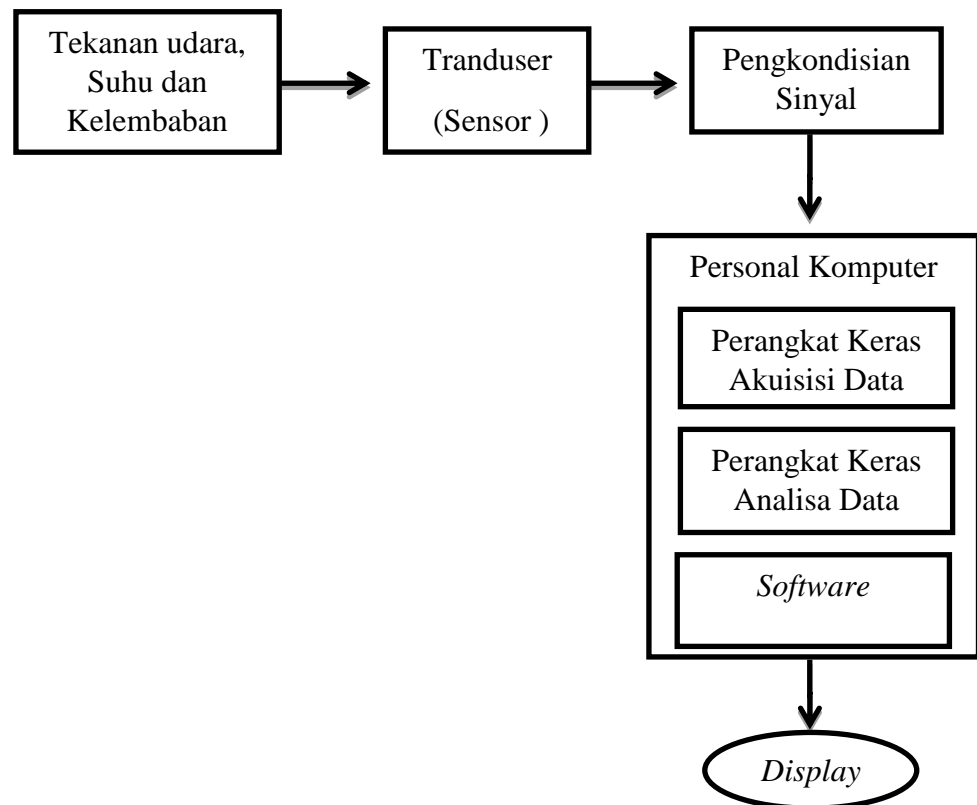


BAB II DASAR TEORI

2.1 SISTEM AKUISISI DATA^[1]

Data Acquisition System (DAS) atau sistem akuisisi data adalah suatu proses dalam pengambilan data sampel pada saat fenomena fisik seperti tekanan udara, suhu, kelembaban dan lain-lain. Serta mampu mengkonversikan data sampel menjadi nilai numerik yang dapat dimanipulasi oleh komputer. Atau dengan pengertian lain bahwa sistem akuisisi data adalah instrumen elektronik dari berbagai elemen-elemen elektronik yang secara bersama-sama bertujuan melakukan pengumpulan, penyimpanan, pengolahan data serta penyaluran data. Sehingga menjadi sebuah bentuk informasi yang berarti.



Gambar 2.1 Elemen-Elemen Akuisisi Data Berbasis PC

Elemen-elemen dasar pada sistem akuisisi data berbasis komputer (PC) terdiri dari tranduser, pengkondisian sinyal dan personal komputer yang terdiri dari perangkat keras akuisisi data, perangkat keras analisa data serta *software* sebagai pengembang dan *display*.

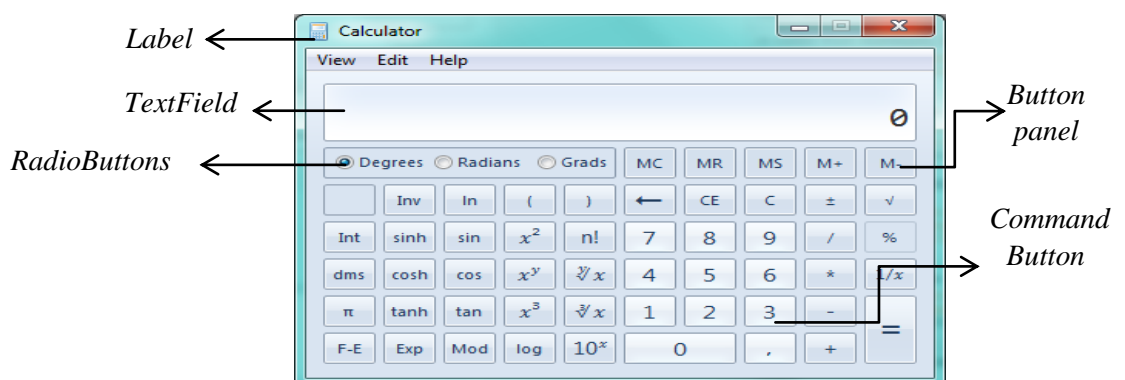
Parameter yang diukur pada sistem akuisisi data untuk Tugas Akhir ini adalah tekanan udara, suhu dan kelembaban. Transduser adalah alat yang digerakan oleh suatu energi pada sistem transmisi dan energi tersebut disalurkan ke sistem transmisi berikutnya. Dapat juga mengubah besaran fisis menjadi besaran listrik berupa tegangan dan arus dengan spesifikasi pada ketelitian, keandalan serta kecepatan. Pengkondisian sinyal adalah proses mengubah sinyal listrik menjadi besaran tegangan ke komputer untuk analisa, pemrosesan, penyimpanan data dari hasil pengukuran. Sinyal –sinyal yang telah dilakukan pengkondisian masih berupa sinyal analog. Pada sistem akuisisi data *hardware* sinyal-sinyal listrik yang telah dikondisikan akan dibaca oleh perangkat komputer yang hanya akan membaca sinyal digital. Sehingga pada aplikasi akuisisi data berbasis GUI untuk mengolah data dan diproses untuk dijadikan sistem *monitoring*. Aplikasi yang digunakan pada Tugas Akhir ini berupa *software Processing*.

2.2 GUI (GRAPHICAL USER INTERFACE)^[2]

GUI (*Graphical User Interface*) atau biasa disebut dengan antar pengguna grafis adalah suatu media virtual yang membuat pengguna memberikan perintah tanpa mengetik perintah tersebut di komputer. Namun lingkungan pengguna GUI menggunakan penyajian berupa gambar maupun teks. Yang berhubungan terhadap masukan dan keluaran dari struktur data dan aplikasi atau jenis lainnya yang menjadi tempat dimana suatu informasi disimpan. Untuk memulai fungsinya dengan GUI, suatu perintah dapat dikonversi menjadi ikon yang dapat diklik dalam suatu layar monitor. Untuk beberapa tombol, ikon dan Windows telah umum digunakan dalam GUI, dan kebanyakan perintah dilakukan dengan menggunakan perangkat jenis *pointing device* seperti *mouse*. Kelebihan menggunakan GUI selain *design* grafis yang lebih menarik juga memungkinkan pengguna untuk berinteraksi dengan komputer dengan cara lebih baik. Contoh untuk *platform* terkemuka yang menggunakan sistem GUI adalah Apple Macintosh dan Microsoft Windows. Dalam membuat aplikasi GUI akan sangat mempertimbangkan aspek kemudahan pengguna program dan keindahan tampilan. Pada gambar berikut terdapat contoh program yang

telah menggunakan GUI, contohnya adalah aplikasi kalkulator pada sistem operasi Windows dengan komponen-komponen pada GUI.

- *Label* = Menyatakan judul dari aplikasi tersebut.
- *Text Field* = Tempat penginputan text pada satu baris.
- *Button panel* = Tombol yang jika ditekan akan melakukan intruksi tertentu.
- *Radio Buttons* = Untuk menandai atau memilih satu pilihan.
- *Command Button* = Menjalankan perintah atau proses ketika ditekan.



Gambar 2.2 *Calculator* Sebagai Contoh Pemrograman GUI

2.3 TEKANAN UDARA, SUHU DAN KELEMBABAN

Dapat diketahui hubungan antara tekanan udara dan kelembaban berbanding lurus, sementara tekanan udara adalah semakin tinggi daerah maka tekanan udara akan semakin rendah, tetapi berbanding terbalik dengan suhu. Apabila terjadi peningkatan suhu maka kelembaban akan menurun dan kapasitas manampung uap air di udara meningkat. Apabila tingkat uap air menurun maka terjadi kelembaban karena penurunan suhu begitu juga dengan tekanan udara, apabila uap air mengembang ke atas udara sehingga kandungan uap air dalam udara akan lebih sedikit.

2.3.1 TEKANAN UDARA

Tekanan udara adalah tenaga menggerakkan berat massa udara pada suatu tempat. Tekanan udara sangat dipengaruhi tingkat kepadatan atau kerapatan massa udara. Kerapatan udara adalah massa persatuan volume gas atmosfer atau dilambangkan dengan huruf Yunani rho (ρ). Kepadatan udara tergantung pada suhu dan

tekanan udara. Satuan SI massa jenis udara adalah kilogram per meter kubik (kg m^{-3}).

Ada dua faktor yang sangat mempengaruhi tekanan udara yaitu suhu dan tinggi suatu daerah:

1. Tinggi suatu tempat berbanding terbalik dengan tekan udara di daerah tersebut.
2. Suhu udara, sangat mempengaruhi tekanan udaranya. Ketika suhu tinggi, molekul udara akan mengembang dan volume udara menjadi lebih besar. Jika volume udara di suatu tempat adalah tetap, maka ketika suhu udara naik massa udara total akan berkurang dan berat udara berkurang, demikian juga dengan tekanan udara. Dan sebaliknya, ketika suhu udara rendah maka tekanan udara akan semakin tinggi.

Pada umumnya dunia menggunakan hectopascal untuk menyatakan nilai tekanan udara. Namun masih ada satuan tekanan seperti cmHg pada barometer tua dengan konversi satuan $1 \text{ cmHg} = 1,103 \times 10^5 \text{ Pa} = 1 \times 1,013 \times 10^3 \text{ hectopascal}$ sebagai pengganti milibar. Hectopascal adalah satuan pengukuran langsung dari tekanan udara. $1 \text{ hectopascal} = 100 \text{ Pascal}$ sedangkan $1 \text{ bar} = 100.000 \text{ newton/m}^2 = 100.000 \text{ Pa}$.

Tekanan udara diberbagai tempat berbeda-beda tergantung pada tinggi rendahnya suatu daerah dari permukaan laut. Perbedaan tekanan udara ini yang mengakibatkan berbagai fenomena cuaca seperti angin topan, badai dan sebagainya. Berikut adalah rumus tekanan udara :

$$P_h = (P_u - h/100) \text{cmHg}^{[3]} \dots\dots\dots(1)$$

2.3.2 SUHU^[4]

Suhu adalah besaran yang menyatakan derajat dingin dan panas suatu benda, dan suhu memiliki perbedaan dengan kalor yang adalah salah satu bentuk energi. Benda yang bersuhu tinggi mempunyai kalor yang banyak dan sebaliknya benda yang memiliki suhu rendah mempunyai kalor yang sedikit. Terkait dengan hal tersebut untuk keadaan dingin, panas dan hangat adalah hasil yang

menyatakan ukuran derajat panas pada suatu benda dan menjadi indikator rendah atau tingginya suhu adalah indikator banyak atau sedikitnya kalor pada sebuah benda.

Suhu memiliki 4 jenis satuan atau besaran suhu yang digunakan dunia sampai sekarang, yaitu:

1. Celcius ($^{\circ}\text{C}$)

Skala Celcius dikembangkan oleh ahli astronomi Swedia Anders Celcius (1701-1744) pada tahun 1742, mengusulkan suatu skala sebagai patokan untuk mengukur suhu air membeku dan air mendidih.

2. Reamur ($^{\circ}\text{R}$)

Titik tetap bawah sebesar angka 0 dan titik tetap atas diberi angka 80. Diantara titik tetap bawah dan titik tetap atas dibagi menjadi 80.

3. Fahrenheit ($^{\circ}\text{F}$)

Titik tetap bawah diberi angka 32 dan titik tetap atas adalah 212. Antara titik tetap bawah dan titik tetap atas dibagi 180 skala.

4. Kelvin (K)

Titik terbawah Kelvin adalah nol. Titik ini disebut mutlak, dimana suhu terkecil yang dimiliki benda ketika energi total partikel benda tersebut nol. Kelvin menetapkan suhu es melebur dengan angka 273 dan suhu air mendidih dengan angka 373. Rentang antara titik tetap bawah dan titik tetap atas thermometer Kelvin dibagi menjadi 100 skala.

Suhu udara adalah rata-rata ukuran kinerja dari pergerakan molekul-molekul. Dalam 24 jam suhu di udara akan selalu mengalami perubahan. Banyak perubahan yang terjadi terhadap suhu diantaranya sifat-sifat suatu zat. Misalnya pada besi yang warnanya akan berubah ketika dipanaskan dan zat lain akan berubah ketika suhu dinaikan, tekanan dan daya hantar listrik juga mengalami perubahan apabila suhu berubah. Sifat zat akan berubah ketika terjadi perubahan sifat termometik zat. Sifat termometik tersebut dapat dimanfaatkan dalam melakukan pengukuran terhadap

besarnya suhu. Untuk suhu dan kelembaban yang relatif diterima manusia pada ruangan untuk kenyamanan manusia. Dalam ilmiah kurang lebih 20° hingga 25° Celcius, 68° sampai 77° Fahrenheit, 293 hingga 298 Kelvin, 528 sampai 537 Rankine. Namun dalam perhitungan yang lebih sering digunakan untuk suhu didalam ruangan yaitu angka 20° Celcius dan 293 Kelvin.

Tabel 2.1 Ukuran Masing-Masing Jenis Satuan Suhu

	Celcius	Reamur	Fahrenheit	Kelvin
Titik didih	100	80	212	373
Titik beku	0	0	32	273
Selisih kedua titik	100	80	180	100
Perbandingan	5	4	9	5

Tabel di atas adalah informasi satuan Celcius, Reamur, Fahrenheit dan Kelvin. Berikut rumus untuk mengkonversi suhu dari skala satu ke skala lainnya adalah:

- a. Konversi suhu dari Celcius (C) ke Reamur (R), Fahrenheit (F) dan Kelvin (K) adalah:

$$R = (4/5)C$$

$$F = (9/5)C + 32$$

$$K = C + 273$$

- b. Konversi suhu dari Reamur (R) ke Celcius (C), Fahrenheit (F) dan Kelvin (K) adalah:

$$C = (5/4)R$$

$$F = (9/4)R + 32$$

$$K = C + 273 = (5/4) R + 273$$

- c. Konversi suhu dari Fahrenheit (F) ke Celcius (C), Reamur (R) dan Kelvin (K) adalah:

$$C = 5/9 (F-32)$$

$$R = 4/9 (F-32)$$

$$K = 5/9 (F-32) + 273$$

- d. Konversi suhu dari Kelvin (K) ke Celcius (C), Reamur (R), Fahrenheit (F) adalah:

$$C = K-273$$

$$R = 4/5 (K-273)$$

$$F = 9/5 (K-273) + 32$$

2.3.3 KELEMBABAN^[5]

Kelembaban udara adalah tingkat keadaan lingkungan udara basah yang disebabkan oleh adanya uap air. Suhu sangat mempengaruhi tingkat kejenuhan. Secara matematis kelembaban relatif (RH) didefinisikan prosentase perbandingan antara tekanan uap air parsial dengan tekanan uap air jenuh. Pembacaan 100% RH berarti udara telah saturasi (udara penuh dengan uap air).

$$RH(\%) = \frac{P_{\text{uap air}}}{P_{\text{uap jenuh}}} \times 100\% \dots \dots \dots (2)$$

Dalam unsur kelembaban udara yang terdiri dari kelembaban relatif dan kelembaban absolut. Kelembaban relatif (nisbi) adalah jumlah prosentase kandungan air yang dihitung dengan dasar udara jenuh selalu harus 100%. Dengan pembacaan 100% kelembaban relatif yang berarti udara telah saturasi dengan kata lain udara penuh dengan uap air. Kelembaban udara relatif akan meningkat prosentasinya apabila udara diinginkan dan sebaliknya prosentase kelembaban menurun ketika dipanaskan. Dengan ketentuan udara tersebut memiliki jumlah kandungan air yang tetap. Perbandingan antara tekanan parsial aktual yang dapat diterima uap air pada suatu udara tertentu dengan tekanan parsial yang diterima uap air dengan kondisi saturasi terhadap suhu udara saat itu. RH dinyatakan dalam persen (%).

Kelembaban spesifik atau biasa disebut kelembaban absolut atau ratio kelembaban (w) adalah perbandingan antara berat uap air yang terkandung dalam udara dengan volume kapasitas udara maksimal diukur dalam (gr/kg). Dalam tekanan barometer tertentu, kelembaban spesifik adalah fungsi dari suhu titik embun. Tetapi pengaruh penurunan pada tekanan barometer menyebabkan kenaikan volume persatuan massa udara naik. Dan sebaliknya

kenaikan tekanan pada barometer akan menyebabkan kelembaban absolut menjadi turun.

2.4 TEMBOO^[14]

Temboo adalah layanan yang memungkinkan pengguna untuk menghubungkan *embeded* sistem ke dunia seperti dapat akses ke cuaca, mengirim sms, email dan layanan lainnya seperti Wolfram Alpha, USPS dan Xively. Temboo menggunakan kode yang secara konsisten terstruktur dan *website* mampu menghasilkan untuk *copy paste* secara langsung ke dalam I.D.E sehingga mudah dan intuitif untuk menggunakan dan menguji semua proses pintar atau sering disebut Choreos, Choreos adalah singkatan koreografi. Kode yang dijalankan pada Choreos yang selanjutnya akan memanggil *platform* Temboo dan terdapat kode rumit dibalik setiap Choreo dijalankan.

Temboo berisi kelas Temboo Choreo yang digunakan oleh sketsa untuk mengeksekusi Tembo Choreos. Kelas Temboo Choreo adalah kelas dari kumpulan *Bridge* yang dikirimkan dengan Arduino I.D.E, kemudian parameter program Temboo klien akan mengambil dan memberikan kode akses untuk membangun permintaan eksekusi Choreo untuk dikirim ke klien Temboo.

Temboo klien adalah program kecil yang berada pada sistem *file* linino dan akan mengetahui bagaimana berkomunikasi dengan *platform* Temboo. Temboo juga menyediakan akses ke beberapa web dengan cara antar muka web *REST*, artinya pengguna berkomunikasi dengan layanan menggunakan protokol yang sama dan menggunakan teknik *web browser* untuk mengakses situs *web* biasa.

2.5 TCP/ IP^[15]

TCP/IP (*Transmission Control Protocol/Internet Protocol*) adalah standar komunikasi data yang digunakan dalam proses tukar menukar data dari satu komputer ke komputer lainnya didalam jaringan Internet. Protokol ini berupa kumpulan protokol (*protocol suite*) dimana data protokol diimplementasikan dalam bentuk perangkat lunak (*software*) sistem

operasi. Protokol ini bersifat *routable* yang berarti cocok untuk menghubungkan sistem-sistem berbeda seperti Microsoft Windows dan keluarga UNIX.

Berikut ini beberapa layer yang dimiliki oleh TCP/IP, diantaranya:

- a. IP (*Internet Protocol*) digunakan saat mentransmisikan paket data dari node ke node. IP mendahului setiap paket data berdasarkan 4 *byte* untuk versi IPv4 alamat tujuan dengan nomer IP. Pada IP bekerja dimesin *gateway* yang memindahkan data dari departemen ke organisasi kemudian ke region dan terakhir keseluruh dunia.
- b. TCP (*Transmission Control Protocol*) berperan memperbaiki pengiriman data yang benar dari satu klien ke server. TCP dapat mendeteksi jaringan *error* atau data yang hilang ditengah-tengah jaringan dan selanjutnya melakukan transmisi ulang sampai data diterima dengan benar dan lengkap.
- c. *Socket* adalah nama untuk paket-paket sebagai penyedia akses ke TCP/IP pada kebanyakan sistem.

2.6 KOMUNIKASI SERIAL^[16]

Komunikasi serial adalah pengiriman data serial dimana data dikirim satu persatu secara berurutan sehingga komunikasi ini lebih lambat dari komunikasi paralel. Komunikasi serial digunakan sebagai pengganti komunikasi paralel jalur data 8-bit dengan cara mengubah data *byte* ke dalam bit-bit serial menggunakan *shift register parallel-in serial-out* dan dikirim hanya pada satu jalur saja. Keuntungan dari penggunaan komunikasi serial ini selain memakan biaya yang lebih murah, namun dapat digunakan untuk menghubungkan dua peralatan yang sangat jauh

Berikut adalah beberapa contoh komunikasi serial yang umumnya dikenal :

- RS-232 (menggunakan UART)
- Sistem *management bus* (SMBus)
- Serial ATA (SATA)
- USART (*Universal Synchronous & Asynchronous Receiver Transmitter*)

- CAN (*Controller Area Network*)
- TWI (*Two Wire Interface*)
- SPI (*Serial Peripheral Interface*)
- *Proprietary* (Protokol yang dibuat oleh pabrikan tertentu)

Untuk memilih protokol komunikasi terdapat beberapa standar komunikasi serial sebagai pertimbangan dan pada beberapa MCUs beberapa *standart support* pada satu IC. Salah satu standarisasi yang cukup dikenal adalah RS232 atau UART adalah serial *interface synchronous* menggunakan 3 sampai 4 kabel untuk berkomunikasi antar perangkat. Data terdiri dari *input*, *output*, *clock*. Kabel ke empat digunakan *slave* sementara untuk memilih *chip* lebih dilakukan oleh BUS.

SMBus terdiri dari satu kabel untuk *input* dan *output* dan satunya untuk *clock*. Serial ATA adalah bentuk serial dari bus yang paralel yang digunakan pada PC seperti koneksi pada *hardisk*.

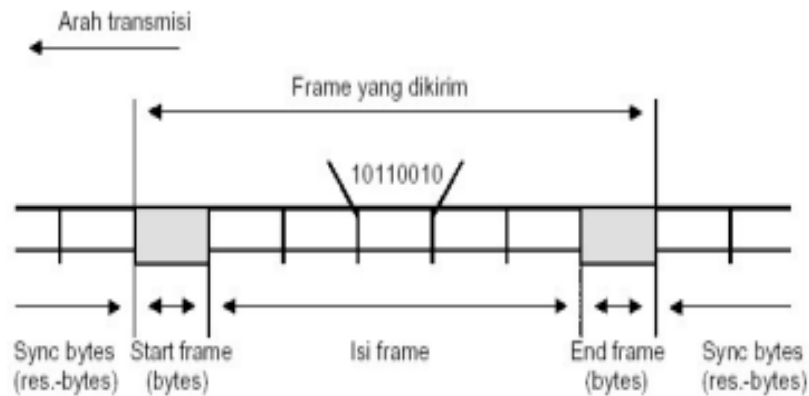
Kecepatan transfer data pada komunikasi data serial diukur dalam BPS (*Bits PerSecond*) atau *baud rate*. Sementara untuk kecepatan transfer data (*data transfer rate*) pada komputer tergantung pada jenis komunikasi yang dilakukan.

2.6.1 Komunikasi Serial Sinkron^[16]

Pada komunikasi serial akan terjadi proses transmisi sinkron seperti (Gambar 2.3) yang adalah transmisi antara pengirim dan penerima berada pada waktu yang sinkron, biasanya dimulai dengan sinyal SYN untuk melakukan sinkronisasi berkomunikasi, kemudian menyusul sinyal STX (*start-of-text*) yang menyatakan awal dari transmisi data, kemudian sejumlah (blok) data dikirim dan ditutup dengan ETX (*end-of-text*), terakhir ada sinyal BCC (*Block-Check-Character*.) yang digunakan untuk mengecek kesalahan dalam penerimaan data.

Komunikasi serial *synchronous* adalah pengiriman data beberapa *byte* atau karakter sebelum meminta konfirmasi memastikan data sudah diterima dengan baik atau tidak dan

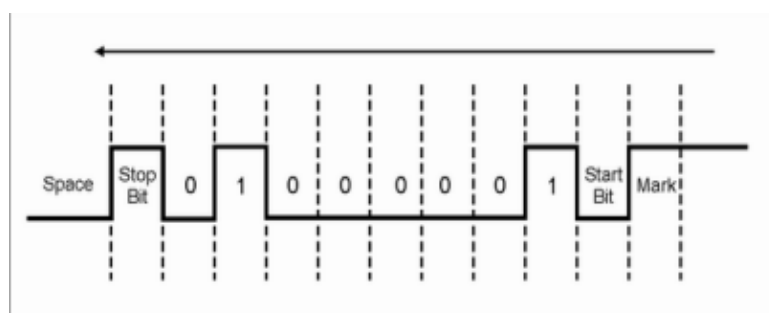
dibutuhkan adanya konfirmasi penerimaan data. Dalam protokol yang berbeda *synchronous* memerlukan sinyal tambahan yang digunakan untuk men-*synchron*-isasi setiap denyut dari proses transfer. Pada metode *asynchronous* digunakan untuk transmisi yang berrorientasi blok.



Gambar 2.3 Sinyal Transmisi Sinkron

2.6.2 Komunikasi Serial Asinkron^[16]

Komunikasi serial *asynchronous* data dikirim satu *byte* setiap pengiriman dan biasanya tidak dibutuhkan konfirmasi penerimaan data. Pada komunikasi *asynchronous* setiap karakter ditempatkan berada diantara bit *start* dan bit *stop*, untuk bit *start* selalu satu bit tetapi *stop* bit dapat satu bit atau dua bit. Untuk *start* bit selalu 0 (*low*) sementara *stop* bit selalu 1 (*high*).



Gambar 2.4 Pembingkai Karakter ASCII "A" (41h)

Pada gambar di atas adalah karakter A (01000001 biner) dibingkai (dikurung) oleh *start* bit dan satu *stop* bit. Sistem karakter ASCII masih terbatas pada 7-bit namun saat ini ASCII *extended* sudah mampu menggunakan lebar data 8-bit.

2.6.3 USART

Komunikasi serial *Universal Synchronous and Asynchronous Serial Receiver and Transmitter* (USART) adalah komunikasi dua arah antara mikro komputer (PC) dengan sistem lainnya seperti mikropengendali baik secara sinkron atau asinkron dengan pengiriman serial melalui pentransfer data bit sampai membentuk satu *frame* data yang diawali dengan *start* bit dan diakhiri dengan *stop* bit, misalnya yang dimiliki oleh ATMega8535.

Kecepatan transfer data (*baud rate*) memiliki nilai-nilai tertentu sehingga mikropengendali sebaagai perangkat yang fleksibel akan mengikuti nilai *baudrate* yang dimiliki komputer. Beberapa nilai standar *baudrate* diantaranya 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 (bit/detik) dan seterusnya. Dalam komunikasi serial, *baud rate* dari PC dan sistem lainnya yang saling terhubung harus diukur dalam kecepatan sama dan selanjutnya harus ditentukan panjang data (6, 7 atau 8 bit), paritas (genap, ganjil atau tanpa paritas) dan jumlah bit “*stop*” (1, 1 ½ atau 2 bit).^[17]

Memiliki fleksibilitas yang tinggi mode komunikasi serial USART dapat berfungsi dalam melakukan transfer data antar mikropengendali serta dengan modul-modul eksternal termasuk pada PC dengan fitur UART (*Universal Asynchronous Receiver Transmitter*). Mikropengendali AVR ATMega16, pin PDO dan PD1 digunakan untuk komunikasi serial USART. Komunikasi serial tersebut mendukung dalam komunikasi dua arah (*full duplex*).

Dengan komunikasi serial USART mampu melakukan transmisi data baik secara *synchronous* maupun *asynchronous*, sehingga sudah dipastikan beroperasi dengan UART. Pengaturan mode *synchronous* maupun *asynchronous* pada ATMega8535 sama, perbedaannya adalah hanya pada sumber *clock* yang digunakan secara bersama-sama sedangkan mode *asynchronous* memiliki sumber *clock* sendiri untuk masing-masing peripheral. Untuk itu secara *hardware* mode *synchronous* membutuhkan 3 pin yaitu

TXD, RXD dan XCK. Sedangkan untuk mode *asynchronous* hanya membutuhkan pin TXD dan RXD.^[18]

2.7 TWITTER^[19]

Twitter adalah jaringan sosial berlogo burung biru yang didirikan pada Maret 2006 oleh perusahaan rintisan *Obvious Corp* yang adalah sebuah layanan *microblogging* dengan mengirimkan pesan singkat terbatas hanya sampai 140 karakter. Mikroblog adalah suatu blog yang memungkinkan pengguna untuk menulis postingan singkat dan memublikasikannya untuk dilihat semua orang ataupun kelompok terbatas yang dipilih oleh pengguna.

Twitter akan bekerja setelah pengguna membuat *account* di *Twitter*. Setiap *tweet* akan mengidentifikasi *Twitterer* dengan nama yang muncul di layar adalah *link* ke profil halaman orang yang bersangkutan. Setelah sudah terdaftar dan membuat hubungan teman maka *homepage Twitter* akan menunjukkan *tweet* yang di-*posting account* lain. Begitu juga pengguna mampu mengirim pesan pribadi ke lainnya.

Twitter terhubung dengan blog dan halaman web lainnya, menyediakan aplikasi *flash* dan pilihan kode *JavaScript* untuk meng-*update* halaman web *Twitter*. *Twitter* juga berperan sebagai *virtual air cooler* dimana orang berbicara mengenai pekerjaan, cuaca dan olahraga atau hal lain yang muncul.

Twitter diterbitkan dalam bentuk API (*Application Programming Interface*) dan dikembangkan secara teratur pada *platform* tersebut. API adalah kumpulan *routine*, struktur data, *object classes* atau protokol yang disediakan oleh *library* atau layanan *operating* sistem untuk mendukung aplikasi yang dibangun. *Twitter* membatasi jenis dan jumlah informasi yang di-*upload* bertujuan untuk menyeimbangkan jumlah konten yang tersedia.

2.8 PERANGKAT KERAS

Perangkat keras (*hardware*) yang digunakan pada pembuatan alat sistem akuisisi data tekanan udara, suhu dan kelembaban dengan

penampilan hasil pengukuran melalui *Twitter* adalah Arduino Nano, sensor tekanan udara yaitu BMP085, sensor suhu dan kelembaban yaitu DHT22.

2.8.1 MIKROPENGENDALI^[6]

Mikropengendali pertama kali dibuat pada tahun 1971 dan diperkenalkan oleh Texas Instrumen dengan seri TMS 1000 yang adalah mikropengendali 4 bit. Mikropengendali adalah sebuah elektronika digital yang terdiri dari *input* dan *output* serta kendali sebuah program.

Mikropengendali adalah sebuah sistem komputer fungsional yang berbentuk sebuah *chip*. Pada *chip* terdapat berbagai komponen pendukung lainnya yaitu terdapat sebuah inti prosesor, memori RAM dan sistem *input-output*. Mikropengendali adalah salah satu dari bagian sistem komputer yang terbangun dari elemen-elemen dasar yang sama seperti sistem komputer. Sistem kerja dari mikropengendali dengan cara memasukan satuan *input*-an berupa program sehingga mikropengendali mampu menjalankan perintah-perintah yang diberikan. Berikut adalah beberapa mikropengendali yang secara umum digunakan, yaitu :

a. MCS51

Mikropengendali ini termasuk jenis mikropengendali CISC.

b. AVR

AVR (*Alv and Vegrd's Risc processor*) adalah mikropengendali RISC 8 bit dengan kode instruksinya dikemas dalam satu siklus *clock*. Secara umum AVR dikelompokkan dalam empat kelas diantaranya ATTiny, AT90Sxx, ATmega dan AT86RFxx. Yang membedakan keempat kelas tersebut adalah memori, *peripheral* dan fungsinya.

c. PIC

PIC termasuk dalam keluarga mikropengendali yang berarsitektur *Hardvard* yang dibuat oleh *Mikrochip Technology*.

d. Arduino

Arduino adalah papan *open source* yang didalamnya terdapat komponen utama yaitu sebuah *chip* mikropengendali jenis AVR.

e. ARM Cortex-M0

ARM adalah sebuah prosesor dengan arsitektur set intruksi 32 bit RISC (*Reduced Instruction Set Computer*) yang dikembangkan oleh ARM *Holdings*.

2.8.2 PENGERTIAN ARDUINO^[7]

Arduino adalah sebuah *platform hardware open source* yang mempunyai *input/output (I/O)* yang sederhana dan pada umumnya mudah digunakan. *platform* tersebut terdiri dari tiga bagian diantaranya :

- a. Aduino *board* memiliki *chip* yang dasar sebuah mikropengendali Atmel AVR. Mikropengendali disini berperan sebagai otak pengendali *input*, proses dan *output* dari sebuah rangkaian elektronik.
- b. *Shield* adalah sebuah papan yang dipasang dibagian atas Arduino *board* berfungsi untuk menambah kemampuan dari Arduino *board*.
- c. Bahasa pemrograman Arduino adalah bahasa yang digunakan untuk membuat perangkat lunak yang ditanamkan kedalam Arduino. Bahasa pemrograman yang dipakai mirip dengan C++. Arduino *Development Environment* adalah perangkat lunak untuk menulis dan meng-*compile* program Arduino.

Secara Umum, Arduino terdiri dari dua bagian utama, yaitu :

1. Bagian H/D

Pada bagian H/D berupa papan yang berisi I/O

Bagian-bagian papan Arduino, yaitu :

a. 14 pin I/O digital (0-13)

Pada pin ini memiliki fungsi sebagai pin *input* dan *output*, masing-masing diantaranya untuk 6 pin (3, 5, 6, 9, 10) adalah pin analog *output*, yang tegangannya dapat diatur.

b. USB

USB berfungsi untuk komunikasi serial dari *board* Arduino ke PC, untuk mentransfer program dari PC ke *board* Arduino dan untuk *supply* tegangan ke *board*.

c. SV1

SV1 adalah penghubung untuk *power supply board*. Dan penghubung ini digunakan untuk *power supply* dari sumber eksternal atau menggunakan USB.

d. Tombol *reset* S1

Tombol *reset* akan me-*reset* program agar kembali ke awal lagi, namun tombol ini tidak akan menghapus program yang ada pada mikropengendali.

e. Q1-kristal (*quartz crystal oscillator*)

Quartz crystal oscillator atau jantung dari *board* berfungsi untuk mengirimkan data yang dikirim ke mikropengendali. Komponen ini yang sering dipilih adalah 16 Mhz, *quartz crystal oscillator* mengirimkan 16 juta data perdetik.

f. *In-Circuit Serial Programming* (ICSP)

Port ini berfungsi untuk memprogram mikropengendali tanpa melalui *bootloader*.

g. IC1-mikrokontroler ATmega

Komponen ini adalah otak dari *board* Arduino yang didalamnya terdapat CPU, ROM dan RAM.

h. X1 atau sumber daya eksternal

Pada *board* Arduino dapat diberikan tegangan DC maksimal sebesar 9-12 V.

i. 6 Pin analog (0-5)

Pin ini berfungsi untuk *input* dan *output* analog dan difungsikan mengubah sinyal analog yang masuk menjadi nilai digital.

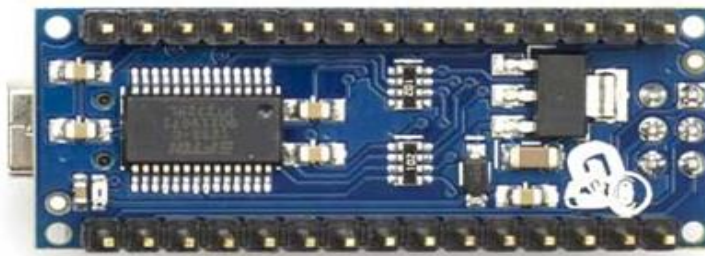
2.8.3 ARDUINO NANO^[8]

Arduino Nano berbentuk papan kecil yang memiliki ukuran lebih kecil dan memiliki fungsi yang sama dari Arduino

Duemilanove tetapi dalam paket yang berbeda. Pada Arduino Nano tidak hanya terdapat *port* listrik DC namun juga bekerja dengan kabel USB Mini-B bukan satu standar. Untuk komunikasi pada Arduino Nano dengan komputer, Arduino lain dan mikropengendali lainnya memiliki fasilitasnya yang tersedia pada pin digital 0 (RX) dan 1 (TX). Gambar 2.5 dan Gambar 2.6 menunjukkan gambar *board* Arduino Nano.



Gambar 2.5 Arduino Nano Tampak Depan



Gambar 2.6 Arduino Nano Tampak Belakang

Arduino memiliki pin yang masing-masing dari 14 pin digital yang digunakan sebagai *input* atau *output*. Memiliki 8 *input* analog yang masing-masing menyediakan 10 bit resolusi atau 1024 nilai yang berbeda. Untuk analog pin 6 dan 7 tidak dapat digunakan sebagai pin digital. Menggunakan *pinMode()*, *digitalWrite()* dan *digitalRead()* yang beroperasi pada 5V yang mana setiap pin mampu memberikan atau menerima maksimum 40 mA dan memiliki resistor *pull-up internal* yang terputus secara *default* dari 20-50 KOhms.

Arduino Nano adalah salah satu dari banyak jenis produk *board* mikropengendali keluaran Arduino. Arduino Nano adalah *board* Arduino terkecil yang menggunakan mikropengendali ATmega328 untuk Arduino Nano 3.x dan ATmega168 untuk Arduino Nano 2.x. Arduino Nano tidak dilengkapi dengan soket

catu daya, tetapi tersedia pin untuk catu daya luar yang dapat diberikan pada pin 30 (+) dan pin 29 (-) untuk tegangan 5V atau dapat menggunakan catu daya dari mini USB *port*. Arduino Nano didesain dan diproduksi oleh Gravitech.

Beberapa pin memiliki fungsi sebagai berikut :

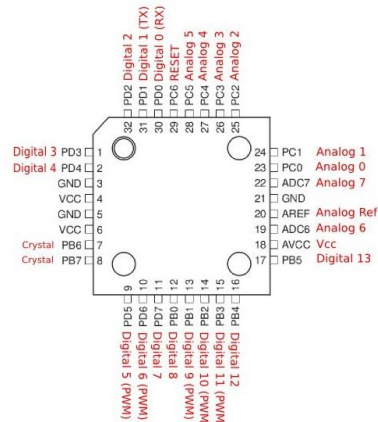
- a. Serial: 0 (RX) dan 1 (TX) digunakan untuk menerima dan mengirimkan TTL data serial. Pin ini terhubung ke pin yang sesuai dari FTDI USB-to-TTL Serial *Chip*.
- b. Pin 2 dan 3 adalah pin interupsi eksternal yang dapat dikonfigurasi untuk memicu interupsi pada nilai rendah, naik, turun atau bahkan mengubah nilai.
- c. Pin 3, 5, 6, 9, 10 dan 11 memberikan 8 bit PWM keluaran dengan fungsi *analogWrite()*.
- d. SPI: 10 (SS), 11 (MOSI), 12(MISO), 13 (SCK) pin ini mendukung untuk komunikasi SPI.
- e. LED 13: Terdapat *built-in* LED yang terhubung ke pin digital 13. Ketika pin tinggi maka LED menyala dan ketika pin rendah LED mati (*off*).

Pada Arduino Nano ini menggunakan ATmega328P yang adalah salah satu anggota keluarga 8-bit mikropengendali yang didasarkan pada arsitektur AVR. Masing-masing kumpulan komponen memiliki jumlah yang berbeda dari RAM, ROM, i/o *Port* dan lain-lain, yang tergantung pada jumlah eksternal PIN yang diperlukan. Mereka dapat datang dalam paket lebih dari seratus pin atau lebih sedikit seperti delapan. Banyak intruksi yang dieksekusi dalam siklus satu jam untuk memberikan *throughput* hampir 20 MIPS di 20 MIPS di 20MHz.

ATmega328P adalah yang dipilih untuk kelas EE 459 untuk berbagai alasan, seperti:

1. Ketersediaan *chip* dan pengembangan perangkat lunak.
2. Tersedia dalam 28-pin DIP (*Dual-Inline Packet*) yang sesuai ke dalam soket IC yang tersedia.

3. Cukup TTL kompatibel i/o pin (21) digunakan untuk menangani sebagai EE 459 L proyek tugas.
4. Memori *flash* untuk mudah dan cepat memogram ulang.



Gambar 2.7 Pin ATmega328P Arduino Nano



Gambar 2.8 ATmega328P Arduino Nano

Terdapat beberapa pin lain yang terdapat pada papan, yaitu:

- a. AREF adalah referensi tegangan untuk *input* analog yang digunakan dengan `analog Reference()`.
- b. Pin *reset* yang berfungsi untuk me-*reset* mikropengendali dan biasanya untuk menambah tombol *reset*. Adapun data yang terdapat pada *board* Arduino Nano ditunjukkan pada Tabel 2.2.

Pada tabel di bawah adalah kumpulan data yang direpresentasikan berdasarkan keadaan Arduino Nano secara umum.

Tabel 2.2 Spesifikasi Arduino Nano

Parameter	Spesifikasi
Mikropengendali	Atmel ATmega168 atau ATmega328
Tegangan operasi	5V
Tegangan masukan (dianjurkan)	7-12 V
Tegangan masukan (batas)	6-20 V
Digital I/O Pins	14
Pins masukan Analog	8
DC Current per I/O Pin	40 mA
Flash Memory	16 KB (ATmega168) atau 32 kbyte (ATmega328), 2 kbyte digunakan bootloader
SRAM	1 KB (ATmega 168) atau 2 kbyte (ATmega328)
EEPROM	512 byte (ATmega168) atau 1 kbyte (ATmega328)
Kecepatan Jam	16 MHz
Ukuran	0.73'' x 1.70''
Panjang	45 mm
Lebar	18 mm
Bobot	3 g

Pada Arduino Nano terdapat pin untuk mengeluarkan tegangan 5V untuk mode *high* (logika 1) dan 0 V untuk mode *low* (logika 0) jika dikonfigurasi sebagai pin *input*, maka 14 pin ini dapat menerima tegangan 5 V untuk mode *high* dan 0 V untuk mode *low* (logika 0). Besar arus listrik yang diijinkan untuk melewati pin digital I/O juga mempunyai fungsi khusus yaitu pin D0 dan pin D1 berfungsi sebagai pin TX dan RX untuk komunikasi data serial. Pin D2 dan D3 berfungsi sebagai pin interupsi eksternal. Kedua pin ini dapat dikonfigurasi untuk pemicu interupsi dari sumber eksternal. Pin D3, pin D4, pin D5, pin D6, pin D9, pin D10 dan pin D11 digunakan sebagai pin PWM (*Pulse Width Modulator*). Pin D10, pin D11, pin D12 dan pin D13 digunakan untuk komunikasi mode SPI. Pin D13 terhubung ke sebuah LED.

Arduino Nano dilengkapi juga dengan 8 buah pin analog, yaitu A0, A1, A2, A3, A4, A5, A6 dan A7. Pin analog terhubung ke ADC (*Analog to Digital Converter*) internal yang terdapat di dalam mikropengendali. Fungsi khusus untuk pin analog antara

lain adalah pin A4 untuk pin SDA, pin A5 untuk pin SCL, pin ini mampu digunakan untuk komunikasi 12C. Pin A_{ref} digunakan sebagai pin tegangan referensi dari luar untuk mengubah *range* ADC. Pin *reset*, pin ini digunakan untuk *me-reset board* Arduino Nano selain dapat di-*reset* melalui pin *reset*, juga dapat di-*reset* dengan menggunakan tombol *reset* yang terpasang pada *board* Arduino Nano.

Untuk komunikasi Arduino Nano sudah terdapat di Arduino Nano untuk berkomunikasi dengan komputer (PC atau laptop), atau dengan *board* mikropengendali lainnya. ATmega168 dan ATmega328 dilengkapi dengan komunikasi serial UART TTL (5V), yang terdapat pada pin D₀ dan pin D₁. *Board* juga dilengkapi dengan sebuah IC FTDI 232 RI yang dapat dihubungkan langsung ke komputer untuk menghasilkan sebuah virtual *com-port* pada sistem operasi. *Software* Arduino (*sketch*) yang digunakan sebagai Arduino I.D.E juga dilengkapi dengan serial monitor yang memungkinkan programmer menampilkan data serial sederhana yang dapat dikirim atau diterima dari *board* Arduino Nano akan berkedip jika terjadi komunikasi data serial antara PC dengan Arduino Nano. Selain mampu berkomunikasi menggunakan data serial melalui virtual *com-port*, Arduino Nano juga dilengkapi dengan mode komunikasi 12C (TWI) dan SPI untuk komunikasi antar *hardware*.

Pada Arduino Nano dapat dengan mudah melakukan pemrograman dengan menggunakan *software* Arduino (*sketch*). Pada menu program, pilih *tool-board* kemudian pilih jenis *board* yang akan digunakan dalam memrogram. Untuk memrogram *board* Arduino dapat memilih tipe *board* Arduino Diecimila atau Duemilanove atau dapat langsung memilih Nano W/ATmega168 atau Nano W/ATmega328. Pada Arduino sudah dilengkapi program *bootloader*, sehingga programmer dapat langsung meng-*upload* kode program *board* Arduino tanpa melalui *board* perantara atau *hardware* lain. Komunikasi ini menggunakan

protokol STK500 keluaran ATMEL. Dapat juga meng-*upload* program ke *board* Arduino Nano tanpa menggunakan *boatloader*, tetapi melalui (*in-circuit serial programming*) *header* yang sudah tersedia di *board* Arduino Nano.

2.8.4 SENSOR BMP085^[9]

Sensor BMP085 adalah sensor pengukur tekanan udara dengan memanfaatkan tekanan udara berdasarkan ketinggian terhadap permukaan laut. Prinsip kerja dari sensor BMP085 adalah mendeteksi ketinggian objek dengan memanfaatkan tekanan udara pada saat berada di atas udara dalam suatu wilayah tertentu. Semakin tinggi suatu tempat maka semakin sedikit jumlah udara di atasnya dan tekanan menjadi sedikit.



Gambar 2.9 Sensor BMP085

Keluaran pengukuran dari sensor ini berupa satuan Pa (Pascal). Untuk mendapatkan hasil ketinggian harus memasukkan nilai tekanan udara yang didapat dari sensor BMP085. Pada *library* sensor ini memudahkan mendapat nilai ketinggian.

Alat untuk mengukur tekanan udara disebut barometer dengan satuan Pa (Pascal). Berikut adalah persamaan ketinggian dalam satuan meter :

$$\text{Altitude} = 44330 * \left(1 - \left(\frac{P}{P_0} \right)^{\frac{1}{5.255}} \right) \dots \dots \dots (3)$$

Pada (Tabel 2.3) dapat dilihat fungsi dari masing-masing pin BMP085 yang adalah sensor tekanan udara.

Tabel 2.3 Fungsi PIN BMP085

PIN BMP085	FUNGSI PIN
VCC	Power (1.8V-3.6V)
GND	Ground
EOC	End of Conversion output
XCLR	Master Clear (low-active)
SCL	Serial Clock I/O
SDA	Data Serial I/O

Persamaan di atas adalah persamaan non-linear yang mengubah tekanan untuk ketinggian dengan p_0 adalah tekanan dipermukaan laut 101325 Pa dan p adalah tekanan yang diukur. Setelah menemukan ketinggian maka selanjutnya hanya perlu menambahkan parameter untuk menghitung ketinggian.

2.8.5 SENSOR DHT22^[10]

DHT22 atau AM2302 adalah sensor suhu dan kelembaban untuk ruangan. Memiliki rentang pengukuran suhu dan kelembaban yang lebar. Menyala pada tegangan minimal 3.3V hingga maksimal 5V dan memiliki rentang deteksi suhu -40C sampai +80C.

Sensor DHT22 adalah gabungan dari beberapa unsur seperti sinyal digital suhu dan kelembaban sensor output yang dikalibrasi sinyal digital. DHT22 dilengkapi dengan keandalan yang sangat tinggi dan stabilitas jangka panjang yang sangat baik. DHT22 termasuk dalam sensor kapasitif kelembaban dan suhu NTC mengukur elemen yang dihubungkan dengan kinerja tinggi mikropengendali 8 bit, dihasilkan berkualitas tinggi, waktu respon super cepat, kuat anti gangguan kemampuan dan sangat hemat biaya. DHT22 digital sensor suhu dan kelembaban dirancang untuk antar muka analog sensor. Antar muka analog digunakan sebagai satu digital yang tidak akan menempati antar muka digital lainnya di Arduino. Dapat juga menggunakan analog ke digital dengan mengkonversi kabel untuk menghubungkan DHT22 ke antar muka digital.



Gambar 2.10 Sensor DHT22

DHT22 adalah sensor suhu dan kelembaban yang dua parameternya digunakan sebagai tolak ukur pada berbagai aplikasi yang memiliki rentang jangkauan pengukuran yang luas mulai dari 0% hingga 100% untuk tingkat kelembaban dan -40°C hingga 125°C untuk suhu. Selain itu sensor DHT22 juga dilengkapi dengan satu buah *output* digital (*single bus*) yang mampu memberikan hasil dengan tingkat ketepatan pengukuran yang tinggi. Dalam merancang *source code* (*sketch*) Arduino untuk suhu dan kelembaban membutuhkan *library* DHT.

Di bawah ini adalah tabel keterangan dari bagian-bagian pin DHT22.

Tabel 2.4 Pin DHT22

Pin	Fungsi
1	vcc
2	data
3	<i>no connect</i>
4	GND

DHT22 memiliki akurasi yang lebih baik. Sangat mudah menghubungkan sensor ke *breadboard* karena dengan kaki yang cocok untuk spasi lubangnya. Untuk konfigurasi yang dimiliki DHT22 terdapat 4 pin dari sensor DHT22 yang terhubung ke Arduino adalah pin 1 untuk VCC 3 Volt sampai 5 Volt, Pin 2 untuk data pengukuran suhu dan kelembaban, pin 3 untuk *no connect* dan pin 4 untuk *ground*. Komponen ini terhubung dengan mikropengendali pada *port B* pin 0.

Tabel di bawah ini adalah parameter dan spesifikasi yang mempresentasikan keadaan pada sensor suhu dan kelembaban.

Tabel 2.5 Spesifikasi Teknis DHT22/AM-2302^[2]

Parameter	Spesifikasi
Rentang catu daya	3.3-6 Volt DC (tipikal 5VDC)
Arus saat pengukuran	1 hingga 1,5 mA
Arus pada moda siaga	40-50uA
Sinyal keluaran	Digital lewat bus dengan kecepatan 5ms/operasi (MSB)
Elemen pendeteksi	Kapasitor polimer
Jenis sensor	Kapasitif
Rentang deteksi kelembapan	0-100% RH (akurasi $\pm 2\%$ RH)
Rentang deteksi suhu	-40° - +80°C (akurasi 0,5°C)
Resolusi sensitivitas	0,1%RH; 0,1°C
Pengulangan	$\pm 1\%$ RH; $\pm 0,2^\circ\text{C}$
Histeris kelembapan	$\pm 0,3\%$ RH
Stabilitas jangka panjang	$\pm 0,5\%$ RH/tahun
Periode pemindaian rata-rata	2 detik
Ukuran	25,1 x 15,1 x 7,7 mm

Penampilan nilai suhu dan kelembaban dengan format :

- Suhu : $T = (\text{Nilai_Suhu})^\circ\text{C}$

- Kelembaban : $H = (\text{Nilai_Kelembaban})\%$

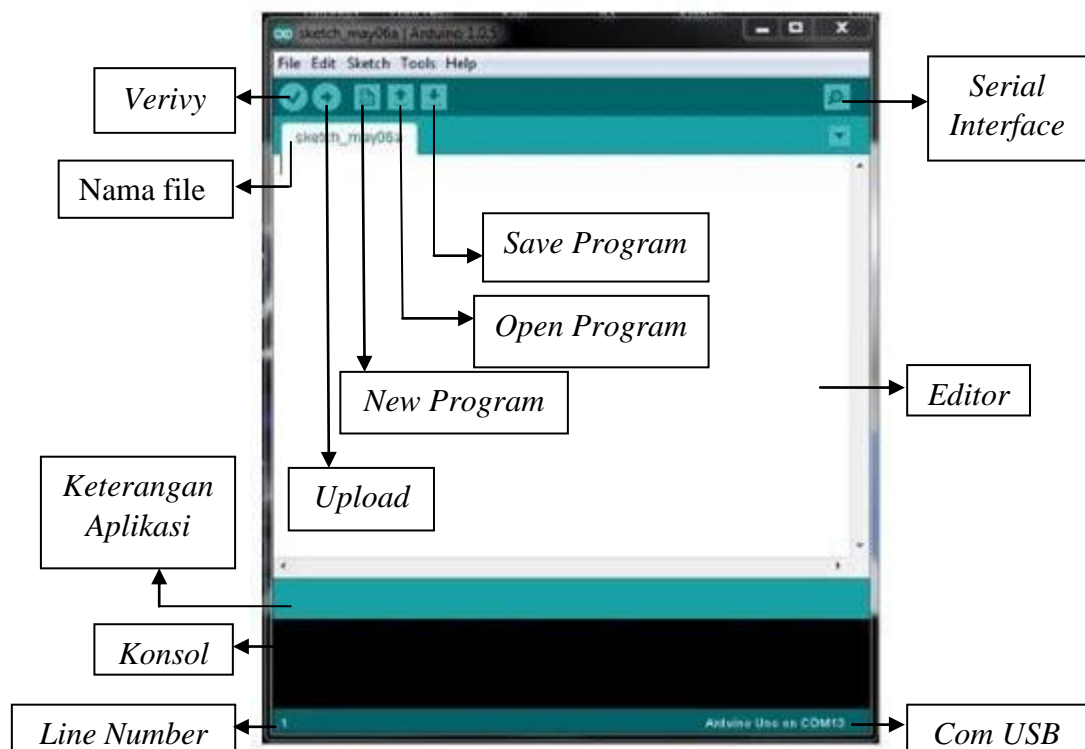
Keluaran berupa sinyal digital dengan konversi dan perhitungan dilakukan oleh MCU 8-bit terpadu dengan kalibrasi tersimpan secara akurat dengan kompensasi suhu diruang penyesuaian dengan nilai koefisien kalibrasi tersimpan dalam memori OTP terpadu. DHT22 memiliki rentang pengukuran suhu dan kelembaban yang lebih lebar dan mampu mentransmisikan sinyal keluaran melewati kabel yang panjang hingga 20 meter.

2.9 PERANGKAT LUNAK

2.9.1 ARDUINO I.D.E^[20]

Arduino I.D.E adalah *software* yang berfungsi untuk mengembangkan dan mengisi program ke dalam Arduino. Arduino I.D.E juga dapat digunakan di OS Windows, Mac OS dan Linux. Untuk penggunaannya sesuaikan dengan OS yang digunakan di komputer masing-masing. Pada Arduino I.D.E menggunakan dasar pemrograman C. Tugas lain dari *software* Arduino I.D.E adalah

menghasilkan sebuah *file* berformat *hex* yang akan di-*download* pada papan Arduino atau papan sistem mikropengendali lainnya.



Gambar 2.11 Aplikasi Arduino I.D.E

Bagian ini berupa *software* Arduino yang meliputi I.D.E (*Integrated Development Enviroment*) untuk menulis program. *Arduino Development Enviroment* adalah perangkat lunak untuk menulis dan meng-*compile* program Arduino. Bahasa pemrograman Arduino adalah bahasa yang digunakan untuk dapat membuat perangkat lunak yang akan ditanamkan pada Arduino. Bahasa pemrogramannya mirip dengan C++, *Arduino I.D.E* adalah *software* yang sangat canggih ditulis dengan menggunakan Java. *Arduino I.D.E* terdiri dari:

1. *Editor* Program

Sebuah *Windows* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.

2. *Compiler*

Modul yang berfungsi mengubah kode program menjadi kode biner bagaimanapun sebuah mikropengendali tidak akan dapat memahami bahasa *Processing*.

3. *Uploader*

Sebuah modul yang memuat kode biner dari komputer ke dalam memori didalam papan Arduino.

Dalam bahasa pemrograman Arduino ada tiga bagian utama yaitu struktur, variabel dan fungsi:

1. Struktur Program Arduino

a. Kerangka Program

Kerangka program Arduino sangat sederhana, yaitu terdiri atas dua blok. Blok pertama adalah *void setup()* dan blok kedua adalah *void loop*.

1). Blok *Void Setup* ()

Berisi kode program yang hanya dijalankan sekali saat setelah Arduino dihidupkan atau di-*reset*. Adalah bagian persiapan atau instalasi program.

2). Blok *Void loop* ()

Berisi kode program yang akan dijalankan terus menerus. Dan tempat untuk program utama.

b. Sintaks Program

Baik blok *void setup loop()* maupun blok *function* harus diberi tanda kurung kurawal buka “{” sebagai tanda awal program diblok itu dan kurung kurawal tutup “}” sebagai tanda akhir program.

2. Variabel

Sebuah program secara garis besar dapat didefinisikan sebagai instruksi untuk memindahkan angka dengan cara yang cerdas dengan menggunakan sebuah variabel.

3. Fungsi

Pada bagian ini meliputi fungsi *input* digital, *input output* analog, *advanced I/O*, fungsi waktu, fungsi matematika serta fungsi komunikasi.

Pada proses *uploader* dimana pada proses ini mengubah bahasa pemrograman yang nantinya di-*compile* oleh avr-gcc (avr-gcc *compiler*) yang hasilnya akan disimpan kedalam papan Arduino.

Avr-gcc *compiler* adalah suatu bagian penting untuk *software* bersifat *open source*. Dengan adanya avr-gcc *compiler*, maka akan membuat bahasa pemrograman dapat diterjemahkan oleh mikropengendali. Proses terakhir ini sangat penting, karena dengan adanya proses ini akan membuat proses pemrograman mikropengendali menjadi sangat mudah.

Berikut ini adalah gambaran siklus yang terjadi dalam melakukan pemrograman Arduino:

1. Menghubungkan papan Arduino dengan komputer melalui *port* USB.
2. Menuliskan sketsa rancangan suatu yang akan dimasukkan ke dalam papan Arduino.
3. Mengunggah sketsa program ke dalam papan Arduino melalui kabel USB dan kemudian tunggu beberapa saat untuk melakukan *restart* pada papan Arduino.^[17]

Papan Arduino akan mengeksekusi rancangan sketsa program yang telah dibuat dan di-*upload* ke papan Arduino.

2.9.2 PROCESSING^[11]

Processing adalah bahasa pemrograman berbasis Java dengan fungsi *multiplatform* yang dapat digunakan pada GNU/Linux, Windows, Mac OS X dan Map ataupun Monitor Server dan dapat menampilkannya dalam bentuk histogram. Secara umumnya sintaks pemrograman dalam *Processing* hampir mirip dengan bahasa pemrograman C atau C++ dan Java.

Processing juga adalah bahasa pemrograman *open source* yang dapat digunakan dalam pengembangan aplikasi Arduino I.D.E (*Integrated Development Environment*) yang dibangun dengan tujuan mengajarkan dasar-dasar pemrograman komputer dalam konteks visual dan dasar sketsa elektronik. Sistem ini difasilitasi teknik mengajar komputer grafik dan banyak interaksi seperti *vector* atau *rester* gambar, pengolahan citra, model

warna, *mouse* dan *keyboard*, jaringan komunikasi dan pemrograman berorientasi objek (*object oriented programming*).

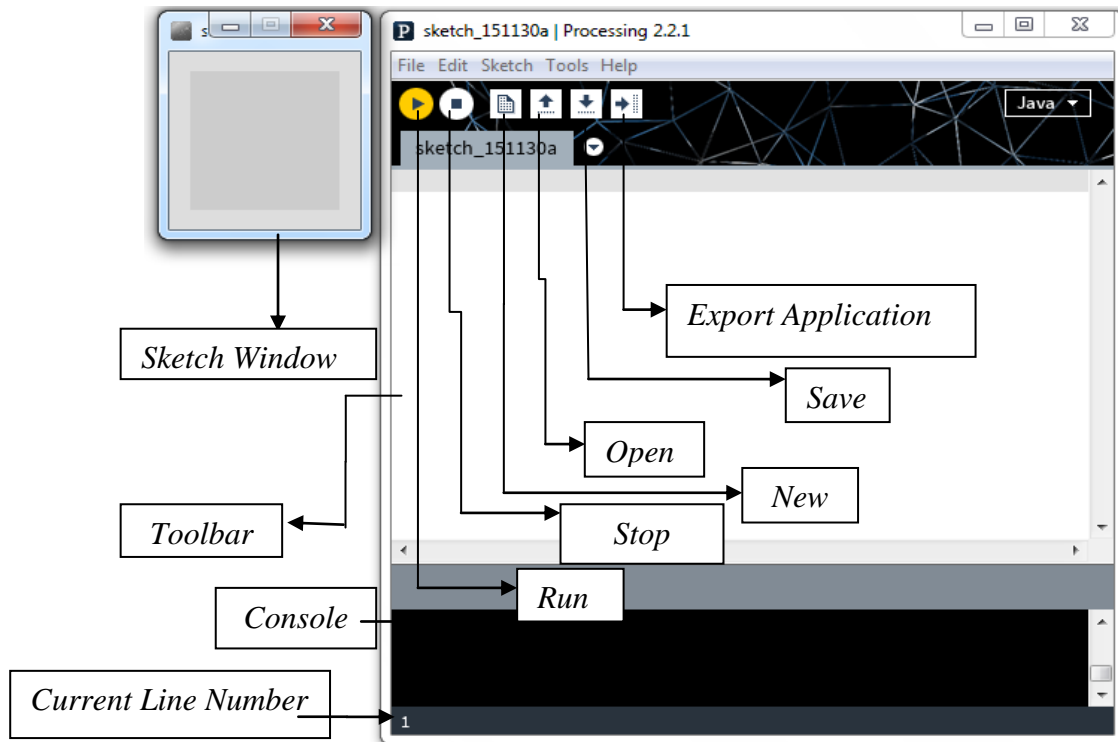
Silsilah pemrograman *Processing* yang mempunyai sistem seperti bahasa manusia. Bahasa pemrograman ini diurutkan berdasarkan level relasi dengan bahasa pemrograman lainnya. Bahasa *Processing* ini berbasis Java yang memiliki dialek dan formula sintak yang hampir identik. Tetapi bahasa *Processing* memiliki *custom features* tambahan yang berhubungan dengan grafik dan intraksi. Grafik elemen *Processing* tersebut merujuk ke *PostScript* dan *OpenGL*. Maka dari itu mempelajari bahasa *Processing* adalah dasar dari bahasa pemrograman lainnya. Di bawah ini akan ditunjukkan bahasa *Processing* menjadi dasar pada lingkungan bahasa pemrograman lainnya.

Pada tampilan *Processing* terdapat editor teks (*text editor*) yang adalah salah satu bagian dari *Processing Development Environment* (PDE) yang terdiri area untuk menulis *code*, area pesan (*message area*), konsol teks (*text console*), tab pengelola *file*, toolbar dengan tombol untuk melakukan tindakan dalam serangkaian menu. Terdapat *display Windows* yang adalah jendela baru ketika program dijalankan.

Processing diciptakan didalam sebuah lingkaran pemrograman yang sederhana dimana pengembangannya untuk aplikasi yang berorientasi visual dengan sarana ide dalam kode yaitu *sketsa*. Penekanan pada aplikasi ini adalah animasi yang menyediakan umpan balik melalui interaksi bagi pengguna. *Processing* berevolusi yang digunakan untuk *prototipe* alat dalam skala yang besar adalah visualisasi data kompleks, instalasi dan grafis gerak serta menjadi sebuah *full-bown* desain.

1. Metode atau perintah (*A collection of function*) membentuk antar muka atau inti pemrograman, menggambar melalui *OpenGL*, menyelamatkan kompleks pencitraan dalam format *PDF*, dan membaca *file XML* terhadap perpustakaan yang mendukung fitur lanjutannya.
2. Sintaks bahasa yang identik dengan Java, tetapi dengan beberapa modifikasi.

3. *The Processing Development Environment* (PDE) adalah *Integrated Development Environment* (IDE) terdiri dari fitur minimalis yang perancangannya sebagai pengantar sederhana pada pemrograman. Perangkat lunak akan berjalan ketika menekan dua kali *icon* pemrosesan.



Gambar 2.12 PDE (*Processing Development Environment*)

2.9.2.1 Sketchbook^[12]

Sketsa adalah sebuah program *Processing* yang disimpan pada *sketchbook*, dan folder yang digunakan untuk lokasi *default* dalam penyimpanan projek yang telah dibuat. Dalam direktori buku sketsa atau *sketchbook* terdiri dari dua sub direktori yaitu, tempat penyimpanan hasil pekerjaan disebut direktori *default* dan direktori *examples* adalah tempat menyimpan contoh-contoh program yang telah dipelajari. Sektsa dengan pengelolaannya terlampir ide yang bertujuan dalam penulisan kode dengan mengadopsi proses *scripting* dan untuk pembuatan program dengan bergaya Java seperti skrip.

2.9.2.2 Struktur Program Processing^[12]

Di dalam dasar pemrograman *Processing* terdapat beberapa notasi yaitu berupa tipe kembalian, nama fungsi, deklarasi dan pernyataan (*statement*). Nama fungsi berhubungan dengan aturan penamaan tentang karakter apa saja yang digunakan. Deklarasi parameter yang akan dijalankan pada suatu program. Persamaan (*statement*) pada program dipergunakan untuk melakukan suatu tindakan, dalam memperkenalkan nama dari variabel atau yang lainnya dan disertakan tipe data. Struktur program di *Processing* dapat dilihat pada contoh program sederhana untuk membuat kotak dengan *listing* program seperti dibawah ini:

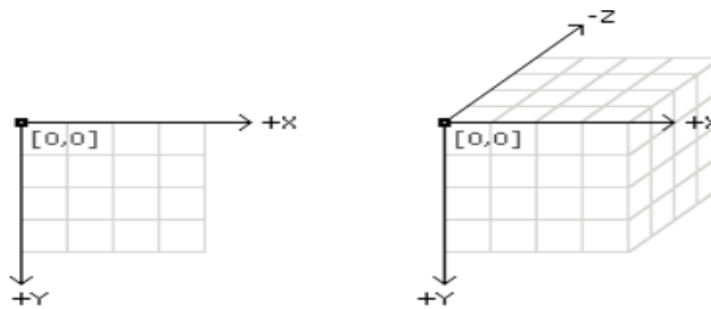
```
void setup()
{
  size(300,300);
  smooth();
}
void draw()
{
  background(128);
  fill(255);
  strokeWeight(3);
  stroke(0);
  rect(50,50,200,200);
}
```

Struktur pemrograman pada *Processing* terdiri dari dua kode fungsi dasar yaitu *void setup()* dan *void draw()*. Untuk kode *void setup()* berfungsi sebagai pengaturan program perintah yang didalamnya dijalankan hanya satu kali ketika program mulai berjalan diawal. Sementara *void draw()* berfungsi sebagai fungsi dasar untuk menggambar sebuah pada layar.

2.9.2.3 Sitem Koordinat Processing^[12]

Gambar di bawah adalah koordinat *Processing* menggunakan sistem koordinat kartesian dengan titik asal terletak di sudut kiri atas. Apabila program berukuran lebar 320 piksel dan lebar 240 piksel,

maka koordinat $[0,0]$ terletak di kiri-atas dan koordinat $[320, 240]$ terletak di kanan bawah.



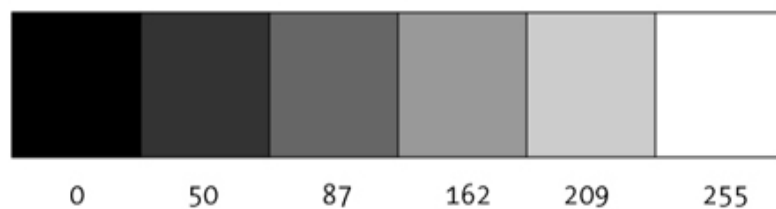
Gambar 2.13 Sistem Koordinat (dimensi X,Y,Z) *Processing*.

2.9.2.4 Sistem Pewarnaan *Processing*^[13]

Warna adalah efek yang diciptakan dalam visual manusia yang disebabkan oleh cahaya atau eksitasi radiasi elektromagnetik. Pada *design* grafis dan untuk proses pengolahan aplikasi komponen yang sangat penting ialah pewarnaan sangat mempengaruhi estetika aplikasi yang dirancang. Sistem pewarnaan pada *Processing* terbagi menjadi *grayscale* dan RGB. Untuk lebih memahami terdapat pada penjelasan di bawah ini.

2.9.2.4.1 *Graycale*^[13]

Graycale adalah sistem warna sebagai nilai antara 0 dan 255, untuk 0 adalah warna hitam dan 255 adalah warna yang paling ringan yaitu putih. Sebuah *image digital* dengan 256 warna abu-abu yaitu antara 0 hingga 255.



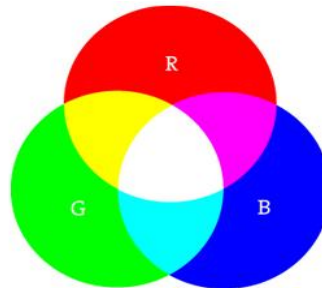
Gambar 2.14 Sistem Pewarnaan *Grayscale*

2.9.2.4.2 RGB (*Red, Green, Blue*)^[13]

Tiga komponen warna seperti merah, hijau dan biru yang biasa disebut dengan RGB dapat menentukan warna suatu objek. Berdasarkan teori *colorimetry modern*

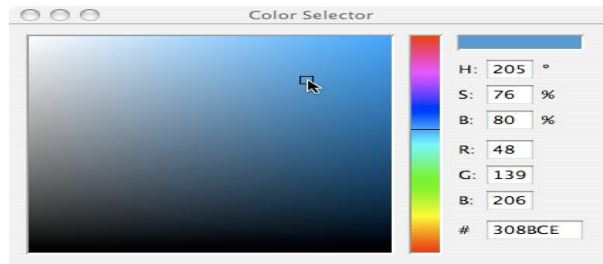
untuk suatu warna didasarkan terhadap perbedaan eksitasi terhadap tiga reseptor cahaya dalam retina. Perbedaan warna muncul berdasarkan tingkat intensitas dari komponen RGB. Layar komputer terdapat bintik kecil terdapat komponen warna RGB, ketika masing-masing intensitasnya ditentukan dan dikombinasikan akan membentuk warna-warna tertentu.

Ketiga komponen itu dapat digambarkan sebagai koordinat tiga dimensi atau kubus RGB. Koordinat berdasarkan warna RGB atau merah, hijau dan biru. Untuk penyimpanannya pada setiap komponen memerlukan 1 *byte*, sehingga komponen dibutuhkan 3 *byte*. Jika masing-masing intensitas telah ditentukan maka akan membentuk warna-warna tersendiri dan untuk kombinasi warna pada sistem ini setara dengan 16 juta warna atau 2^4 . *True Color* adalah format penyimpanan warna RGB. Untuk sistem pewarna RGB memiliki ciri khas tersendiri, selain lebih jelas dan terang, dapat menghasilkan kapasitas *file* yang lebih kecil.



Gambar 2.15 Sistem Pewarna RGB

Pada *Processing* untuk memudahkan pemilihan warna juga dapat diakses melalui *tools* pada menu bar *color selector*. Berikut adalah tampilannya seperti gambar di bawah ini.



Gambar 2.16 *Color Selector* Pada *Processing*.