



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB 2

LANDASAN TEORI

2.1 Arduino

Arduino adalah sebuah papan pengendali mikro (*microcontroller*) yang bersifat *open-source* (Arduino, 2017), yang artinya cetak biru dari pengendali mikro ini dapat di akses oleh publik dan menjadikan publik dapat memproduksi sebuah papan Arduino sejenis atau bahkan dengan sedikit modifikasi sesuai kebutuhan.

Arduino mampu mengenali masukan dari berbagai jenis sensor seperti sensor cahaya, sensor kelembapan, atau sensor temperatur, dan mengubahnya menjadi keluaran untuk mengaktifkan misalnya sebuah motor elektrik, atau lainnya sesuai dengan kebutuhan.

Arduino juga dilengkapi dengan *integrated development environment* (IDE) khusus untuk memrogram papan Arduino tersebut. IDE ini dapat dijalankan di banyak sistem komputer seperti Windows, Mac, bahkan Linux. IDE ini dibangun dengan menggunakan bahasa pemrograman Java, namun program yang ditulis untuk memrogram Arduino dapat menggunakan bahasa C atau C++.

Pada konteks penelitian ini, Arduino dilengkapi dengan sebuah sensor suara yang dapat menangkap gelombang suara dengan keluaran berupa nilai rentang 0 hingga 1023. Setelah mendapatkan masukan suara tersebut, masukan tersebut di

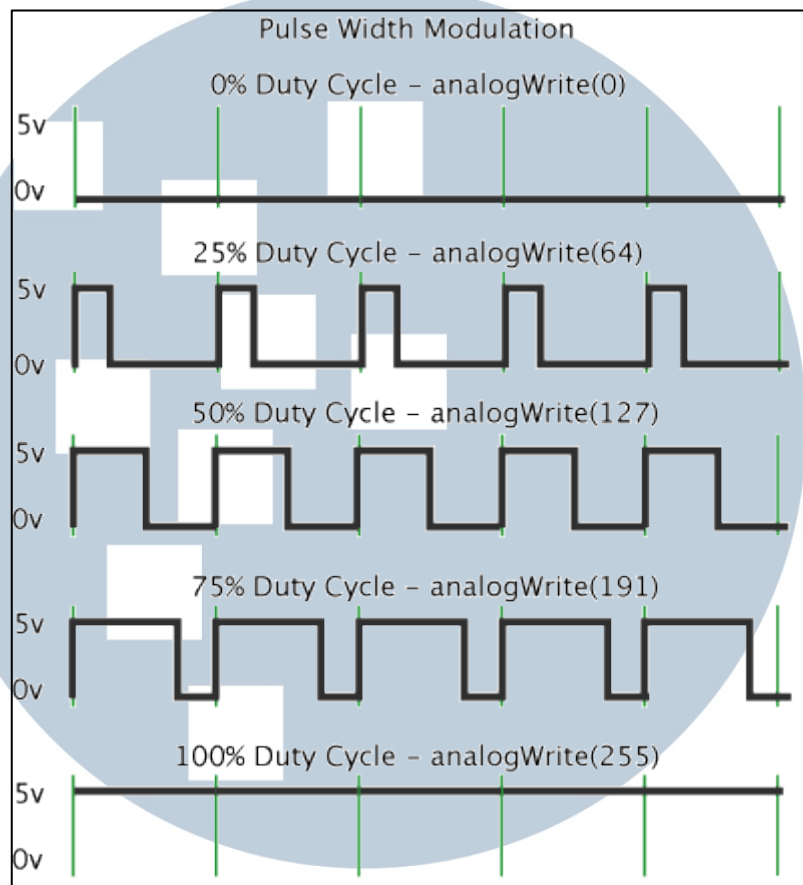
kirimkan secara terus menerus ke program Python dimana data tersebut akan diolah.

Jenis Arduino yang digunakan pada skripsi ini adalah Arduino versi standar, yaitu Arduino Uno. Pada sebuah papan Arduino, terdapat beberapa *port* masukan dan keluaran, 1 *port* USB untuk sumber tenaga sekaligus *port* masukan dan keluaran, dan 1 *port* tenaga DC apabila ingin menggunakan sumber tenaga selain dari USB. Pada salah satu sisi pada papan Arduino, terdapat 13 *digital pulse width modulation* yang dapat mengeluarkan sinyal listrik sebesar 5 volt untuk membentuk sebuah modulasi berbentuk kotak. Modulasi ini dapat di atur dengan cara mengatur lamanya waktu *port* ini menyediakan sinyal listrik tersebut.



Gambar 2.1. Arduino Uno

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 2.2. Diagram macam-macam modulasi

Pada sisi lainnya, terdapat 2 segmen yang berbeda. Satu segmen didedikasikan sebagai *power port* untuk menyediakan sumber energi kepada modul Arduino yang membutuhkan, dan juga sebuah *port* untuk *grounding*. Tenaga yang dapat disediakan dari *port* tersebut adalah sebesar 5 volt. Segmen lainnya merupakan *analog in*. *Analog in* memungkinkan Arduino untuk menerima masukan berupa sinyal analog dari modul-modul Arduino yang dapat menyediakan keluaran analog.

2.1.1 Modul Arduino

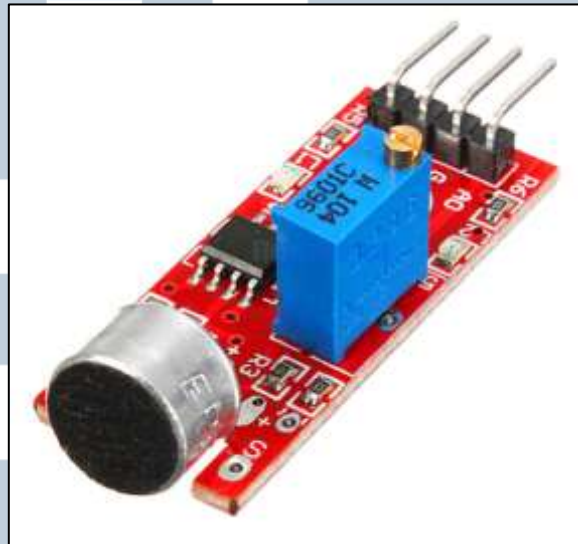
Ada banyak macam modul yang dapat mendukung fungsionalitas dari sebuah papan Arduino. Modul-modul ini dibuat selain oleh penyedia Arduino itu sendiri, juga dapat dibuat oleh pihak ketiga. Banyak macam dari modul Arduino ini. Contoh modul masukan pada Arduino adalah seperti sensor temperatur, sensor kelembapan, dan sensor cahaya. Contoh modul keluaran pada Arduino adalah seperti motor servo, lampu LED, *seven segment*, dan layar LCD kecil. Modul-modul yang akan digunakan pada penelitian ini adalah sebagai berikut.

1. Mikrofon SE019
2. Lampu LED, berwarna merah dan hijau

Mikrofon SE019 ini merupakan modul mikrofon untuk Arduino yang memiliki 4 pin masing-masing untuk *analog output*, *digital output*, *power*, dan *ground*. Mikrofon ini membutuhkan tenaga listrik sebesar 5 volt atau 3.3 volt untuk bisa bekerja. Mikrofon ini dilengkapi dengan sebuah potensiometer yang berfungsi untuk mengatur sensitivitas dari mikrofonnya. Mikrofon ini yang akan bertugas untuk menangkap gelombang suara. Gelombang suara ini kemudian dapat diteruskan ke papan Arduino melalui koneksi pin analog maupun digital.

Sementara dua LED tersebut masing-masing memiliki fungsi untuk menandakan ketika terdeteksi adanya kebisingan untuk yang

berwarna merah dan untuk menandakan bahwa Arduino sedang beroperasional untuk yang berwarna hijau.



Gambar 2.3. Mikrofon SC019.



Gambar 2.4. LED Arduino

2.2 Fast Fourier Transform

Fast fourier transform (FFT) adalah sebuah algoritma untuk mengekstrak frekuensi pembentuk sebuah gelombang suara. FFT ini merupakan turunan dari sebuah rumus pengolah sinyal yakni *discrete fourier transform* (DFT) (Bekele, 2013). Rumus DFT inilah yang pada awalnya dapat melakukan ekstraksi dari sebuah fungsi terhadap waktu menjadi frekuensi pembentuk dari fungsi tersebut. Namun pada perkembangannya dan penerapannya pada kalkulasi menggunakan komputer, rumus DFT ini didapati memerlukan waktu yang lama untuk memproses fungsi masukannya. Menemui kendala tersebut, rumus DFT dimodifikasi dengan cara menambahkan, salah satunya, algoritma *Cooley Tukey*.

Algoritma *Cooley-Tukey* bekerja dengan cara mengelompokkan seluruh data yang hendak diproses menjadi 2 (Bekele, 2013), yakni kelompok data ganjil dan kelompok data genap untuk mengejar kemiripan simetri dari kedua kelompok sehingga komputasi dapat di lakukan dengan paralel (Understanding the FFT Algorithms, 2013). Kemiripan simetri ini kemudian dimanfaatkan untuk mempercepat proses kalkulasi *Fourier Transform* ini. Sehingga apabila masukan *array* untuk kalkulasi FFT ini adalah berukuran $N = 2^m$ dimana $m > 0$ maka waktu yang dibutuhkan untuk melakukan kalkulasi FFT adalah $O(N \log(N))$ (Bekele, 2013).

2.3 Frekuensi

Frekuensi, spesifik pada frekuensi suara, adalah jumlah getaran suara periodik yang terjadi pada satu detik ("Frequency", 2018). Frekuensi merupakan komponen yang menentukan nada (*pitch*) dari sebuah suara dan di tulis dengan

satuan hertz (Hz). *Audible frequency*, atau frekuensi yang dapat di kenali dan di dengar oleh manusia memiliki rentang dari 20Hz hingga 20.000Hz (Pilhofer & Day, 2015, p. 97).

2.4 Python

Python merupakan sebuah bahasa pemrograman tingkat tinggi yang diciptakan oleh Guido van Rossum yang dirilis pada tahun 1991. Bahasa pemrograman yang dapat dijalankan di banyak sistem operasi ini diyakini sebagai bahasa pemrograman yang menggabungkan kapabilitas, kemampuan, dan sintaksis kode yang sangat jelas serta didukung oleh komunitas pengguna bahasa pemrograman ini yang besar (Kuhlman, 2012).

Fungsi dan penerapan bahasa pemrograman Python pun banyak. Yang pertama yakni untuk pengembangan framework untuk pengembangan berbasis internet seperti Django dan Pyramid. Library dari Python pun tersedia dan dapat digunakan pada banyak protokol seperti HTML, JSON, pemrosesan surel, FTP, dan IMAP.

Contoh penerapan Python yang paling umum yakni untuk tujuan ilmiah dan perhitungan matematis. Beberapa produk terbesarnya yakni terdapat Scipy untuk kalkulasi matematis, Pandas untuk proses analisis dan modeling, IPython sebagai tampilan interaktif untuk proses suntingan dan perekaman proses bekerja serta mampu mendukung visualisasi data dan *parallel computing*.

Bahasa pemrograman Python sering kali dibandingkan dengan bahasa pemrograman serupa bernama R. Kedua bahasa pemrograman ini sering

dibandingkan karena kemiripannya dalam sintaksis dan dalam kemampuannya dalam menjalankan tugas analisa data dan visualisasi data.

2.4.1 Perbandingan Python dengan R

Bahasa pemrograman Python memiliki banyak kemiripan dengan bahasa pemrograman R. Kedua bahasa pemrograman ini sama-sama memiliki kemampuan untuk mengolah data, terutama data yang berjumlah besar untuk tujuan *data science*.

Namun, Python dipilih untuk digunakan pada penelitian ini karena Python merupakan bahasa pemrograman yang berbentuk lebih umum. Umum dengan artian bahwa gaya penulisan pada Python lebih *user-friendly* terhadap *programmer*. Dan Python ini selain memiliki kemampuan untuk melakukan olah data, juga dapat digunakan sebagai bahasa pemrograman untuk membuat program yang sederhana di luar konteks olah data.

Sementara itu, R, adalah bahasa pemrograman yang lebih cenderung digunakan untuk keperluan statistika. Bentuk dan gaya pemrograman pada bahasa pemrograman R juga dianggap lebih kaku. Walaupun seperti itu, untuk keperluan statistika bahasa pemrograman ini lebih sesuai seperti untuk membuat sebuah program yang dapat memecah sebuah paragraf menjadi *word cloud* atau frasa.

2.5 Baud-rate

Baud-rate adalah sebuah simbol untuk menggambarkan laju kecepatan dari sebuah pertukaran data yang terjadi pada kanal data (Phongchit, 2016). *Baud-rate*

sendiri erat hubungannya dengan *bit rate*, namun tidak sepenuhnya sama. *Bit rate* pada umumnya memiliki penyebutan sendiri yaitu *bits per second*. Namun karena *bit* hanya memiliki dua simbol, 0 (nol) dan 1 (satu), maka *baud-rate* dan *bit per second* menjadi serupa.

2.6 Mendeteksi kebisingan

Pada skripsi ini, sebuah kebisingan akan dianggap ada apabila sebuah suara yang di tangkap oleh Arduino memiliki frekuensi dominan bernilai di antara 85Hz hingga 255Hz dan melebihi *threshol*d yang telah ditentukan dari hasil pengolahan 152 data suara.

Akurasi alat ini diukur apabila berhasil mengidentifikasi suara manusia. Proses identifikasi dilakukan dengan menyeleksi frekuensi dominan pada suara masukkan apakah bernilai di antara 85Hz hingga 255Hz. Nilai frekuensi tersebut dipilih karena nilai 85Hz hingga 255Hz adalah di mana suara manusia berbicara pada umumnya.

2.7 Penelitian Sebelumnya

1. Library Sound Meter (Nathan David et al., 2013)

Penelitian ini bertujuan untuk membuat sebuah alat untuk mendeteksi sebuah kebisingan di dalam perpustakaan, dan akan mengeluarkan sebuah peringatan untuk mengendalikan kebisingan tersebut menggunakan sebuah pengeras suara yang akan memutar sebuah audio yang sudah direkam sebelumnya.

Penelitian ini menggunakan sebuah mikrokontroler yang dipasang sebuah mikrofon yang keduanya akan menjadi sebuah alat pendeteksi kebisingannya. Hasil deteksi akan ditampilkan pada sebuah layer LCD terpisah. Setelah terdeteksi adanya kebisingan, maka mikrokontrolernya akan memicu sebuah pengolah suara yang kemudian akan memberikan peringatan berupa suara.

2. Perancangan dan Pembuatan Alat Pendeteksi Tingkat Kebisingan Bunyi Berbasis Mikrokontroler (Suyatno, et al., 2010).

Penelitian ini memiliki tujuan untuk membuat sebuah alat yang mampu mengeluarkan sebuah *output* yang menunjukkan kondisi kebisingan di mana alat itu diaktifkan. Pada penelitian ini, kebisingan dikategorikan ke dalam 3 jenis kebisingan yakni kebisingan yang aman, batas bahaya, dan sangat bahaya yang juga akan menyalakan lampu peringatan.

Penelitian ini didasari oleh perhatian dari peneliti bahwa tingkat kebisingan di dalam lingkungan kerja dan ingin membantu untuk mendeteksi apakah kebisingan yang ada sudah termasuk berbahaya atau belum dengan membuat sebuah alat yang berbasis sebuah mikrokontroler AVR.

3. Perancangan dan Realisasi Alat Pendeteksi Kebisingan Suara Kendaraan Dalam Satuan Desibel (Yoga Persada, et al., 2012)

Penelitian ini bertujuan untuk membuat alat yang dapat mengukur tingkat kebisingan. Dengan menghadirkan alat tersebut, para peneliti berharap dapat membantu tegaknya peraturan lalu lintas yang menyinggung tentang tingkat kebisingan yang dihasilkan dari kendaraan bermotor.

Penelitian ini memiliki fokus untuk mengukur suara bising yang dikeluarkan dari kendaraan bermotor seperti suara knalpot terutama knalpot dari sepeda motor yang telah dimodifikasi di kalangan anak muda khususnya sepeda motor yang memiliki ukuran mesin 80cc – 175cc. Dan tidak terhindar juga dari kebisingan yang terjadi karena aktivitas di bandar udara, atau aktivitas di sebuah perusahaan manufaktur yang menggunakan mesin-mesin besar.

UMMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA