



TUGAS AKHIR - TE 141599

PENENTU POSISI DRONE BERDASARKAN SINYAL SUARA

Fardian Abdi Putra
NRP. 07111545000018

Dosen Pembimbing
Dr. Muhammad Rivai, ST., MT.
Ir. Tasripan, MT.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



TUGAS AKHIR - TE 141599

PENENTU POSISI DRONE BERDASARKAN SINYAL SUARA

Fardian Abdi Putra
NRP 07111545000018

Dosen Pembimbing
Dr. Muhammad Rivai, ST., MT.
Ir. Tasripan, MT.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018

Halaman Ini Sengaja Dikosongkan



FINAL PROJECT - TE 141599

Drone Positioning Based On Voice Signal

Fardian Abdi Putra
NRP 07111545000018

Advisor Lecturer
Dr. Muhammad Rivai, ST., MT.
Ir. Tasripan, MT.

*DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018*

Halaman Ini Sengaja Dikosongkan

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “**Penentu Posisi Drone Berdasarkan Sinyal Suara**” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri. Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Januari 2018

Fardian Abdi Putra
NRP. 2215105018

Halaman Ini Sengaja Dikosongkan

PENENTU POSISI DRONE BERDASARKAN SINYAL SUARA

TUGAS AKHIR

**Diajukan Untuk Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik**

Pada

**Bidang Studi Elektronika
Departemen Teknik Elektro
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Muhammad Rival, ST., MT.

NIP. 1969 04 26 1994 03 1003

Ir. Tasripan, MT.

NIP. 1962 04 18 1990 03 1004

**SURABAYA
JANUARI, 2018**

Halaman Ini Sengaja Dikosongkan

PENENTU POSISI DRONE BERDASARKAN SINYAL SUARA

Nama : Fardian Abdi Putra
Dosen Pembimbing 1 : Dr. Muhammad Rivai, ST., MT.
Dosen Pembimbing 2 : Ir. Tasripan, MT.

ABSTRAK

Perkembangan drone saat ini semakin maju baik untuk kepentingan militer maupun sipil. Namun semakin majunya teknologi maka penyalahgunaan drone meningkat. Drone dapat digunakan sebagai alat pengintai yang sangat berbahaya dan melanggar aturan. Oleh karena itu maka perlu dikembangkanlah sistem drone jammer yang bisa melumpuhkan sistem kendalinya. Pada penelitian ini telah dilakukan rancang bangun sistem pendeteksi posisi drone berdasarkan sinyal suara. Sistem ini terdiri dari 4 buah mikrofon sebagai sensor suara dan mikrokontroler Arduino Uno. Tegangan keluaran dari sensor suara akan dikonversi oleh ADC pada mikrokontroler untuk menghasilkan sudut azimuth dan elevasi dari posisi drone yang kemudian ditampilkan pada LCD atau komputer. Berdasarkan hasil pengujian prototipe ini mempunyai kemampuan untuk mendeteksi posisi suara drone dengan kenaikan sudut azimuth sebesar 60^0 dan sudut elevasi sebesar 30^0 dengan jarak sumber suara di bawah 100 cm dan intensitas suara sebesar 84 dB. Tingkat error rata-rata yaitu sebesar 19,41%. Alat ini diharapkan dapat diaplikasikan pada sistem drone jamming sebagai alutsista dalam bidang militer untuk mendeteksi keberadaan drone.

Kata Kunci: *Arduino Uno*, Mikrofon, Posisi drone.

Halaman Ini Sengaja Dikosongkan

DRONE POSITIONING BASED ON VOICE SIGNAL

Name : Fardian Abdi Putra
1st Advisor : Dr. Muhammad Rivai, ST., MT.
2nd Advisor : Ir. Tasripan, MT.

ABSTRACT

The development of drones is now advancing for both military and civil interests. However, the more advanced the drone technology, the more the drone abuse. Drones can be employed as dangerous law breaking spying devices. Therefore, it is necessary to develop a drone jammer system which is able to disable its control system. In this research, drone position detection system has been made based on sound signal. The system consists of 4 microphones as sound sensors and Arduino Uno microcontroller. The output voltage of the sound sensors are converted by the ADC on the microcontroller to determine the azimuth and elevation angles of the drone position which are displayed on the LCD or computer. Based on experiments, this prototype is capable to locate the drone sound with an azimuth angle of 60° and the elevation angle of 30° with the distance of the sound source below 100 cm and the sound intensity of 84 dB. The average error rate is 19.41%. This device is expected to be applied to the jamming drone system as an army defense equipment to detect the presence of drones.

Keywords: *Arduino Uno, Drone Position, Microphone Array.*

Halaman Ini Sengaja Dikosongkan

KATA PENGANTAR

Segala puji syukur penulis panjatkan atas kehadiran Allah SWT yang selalu memberikan rahmat serta hidayah-Nya sehingga penelitian ini dapat terselesaikan dengan baik. Shalawat serta salam selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Pada kesempatan ini penulis ingin mengucapkan ucapan terima kasih yang sebesar- besarnya kepada beberapa pihak yang telah memberikan dukungan selama proses pengerjaan penelitian ini, antara lain:

1. Dr. Muhammad Rivai, ST., MT. dan Ir. Tasripan, MT. selaku dosen pembimbing pertama dan kedua atas bimbingan, inspirasi, pengarahan, dan motivasi yang diberikan selama pengerjaan penelitian ini.
2. Dr. Ir. Hendra Kusuma, M.Eng, Sc, Ir. Harris Pirngadi, MT, Ir.Totok Mujiono, M.IKom, Dr.Eng, dan Dr. Astria Nur Irfansyah, ST, M.Eng. selaku dosen penguji atas pengarahan dan masukan yang diberikan selama pengerjaan buku laporan penelitian ini.
3. Dr. Eng. Ardyono Priyadi, ST., M.Eng. Selaku Ketua Departemen Teknik Elektro Fakultas Teknologi Elektro.
4. Seluruh dosen bidang studi Elektronika Jurusan Teknik Elektro Fakultas Teknologi Elektro ITS.
5. Keluarga penulis ayahanda Usman Hasan, ibunda Ramayani, Istri Syafitri Apriyuni & anak-anak sebagai penyemangat, beserta seluruh keluarga yang selalu memberikan doa, dukungan, motivasi, perhatian dan kasih sayangnya.
6. Seluruh rekan-rekan laboratorium B402 yang telah banyak membantu dalam penyelesaian penelitian ini.
7. Seluruh rekan-rekan yang ikut berperan dalam membantu dan memberikan arahan dalam penyelesaian penelitian ini.

Penulis menyadari bahwa dalam penelitian ini terdapat banyak kekurangan. Akhir kata semoga melalui tulisan ini dapat bermanfaat dan dapat berbagi ilmu bagi pembacanya. Aamiin.

Surabaya, Januari 2018

Penulis

DAFTAR ISI

HALAMAN JUDUL	iii
PERNYATAAN KEASLIAN	vii
LEMBAR PENGESAHAN	ix
ABSTRAK	xi
ABSTRACT	xiii
KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Metodologi.....	2
1.6 Sistematika Penulisan	3
1.7 Relevansi	3
BAB II TEORI PENUNJANG	
2.1 <i>Drone</i>	5
2.2 <i>Sound Localization</i>	6
2.2.1 Microphone array.....	6
2.3 Mikrofon	7
2.3.1 Sensor Mikrofon MAX9814.....	8
2.4 LCD	9
2.5 Arduino Uno	10
2.5.1 ATmega328.....	11
2.6 <i>Neural Network</i>	13
2.6.1 Konsep dasar <i>Neural Network</i>	14
2.6.2 Karakteristik <i>Neural Network</i>	14
2.6.3 Faktor Bobot.....	14
2.6.4 Fungsi Aktivasi.....	15
2.6.5 Model <i>Neural Network</i> Backpropagation	16

2.7	<i>Microsoft Visual Studio</i>	17
2.7.1	Pengenalan Integrated Development Environment (IDE) VB.NET 2013	17
BAB III PERANCANGAN DAN REALISASI ALAT		
3.1	Perancangan Sistem	21
3.2	Blok Diagram.....	22
3.3	Perancangan <i>Hardware</i>	23
3.3.1	Mikrofon MAX9814.....	23
3.3.2	Perancangan Desain Antena Microphone Array	24
3.4	Perancangan <i>Software</i>	27
3.4.1	Pengolahan Data Mikrofon	29
3.4.2	<i>Microsoft Visual Studio</i>	32
3.5	Realisasi Alat	34
BAB IV PENGUJIAN DAN ANALISA SISTEM		
4.1	Pengujian <i>hardware</i>	35
4.1.1	Pengujian mikrofon.....	35
4.1.2	Pengujian ADC Arduino Uno	37
4.2	Pengujian <i>Software</i>	39
4.2.1	Pengujian <i>Neural Network</i>	39
4.2.2	Jarak suara drone 10 cm dengan sudut elevasi 0 ⁰	43
4.2.3	Jarak suara drone 20 cm dengan sudut elevasi 0 ⁰	44
4.2.4	Jarak suara drone 10 cm dengan sudut elevasi 30 ⁰	44
4.2.5	Jarak suara drone 20 cm dengan sudut elevasi 30 ⁰	45
4.2.6	Jarak suara drone 10 cm dengan sudut elevasi 60 ⁰	46
4.2.7	Jarak suara drone 20 cm dengan sudut elevasi 60 ⁰	47
BAB V PENUTUP		
5.1	Kesimpulan	53
5.2	Saran	53
DAFTAR PUSTAKA		55
LAMPIRAN		
BIODATA PENULIS		

DAFTAR GAMBAR

Gambar 2.1 Drone	5
Gambar 2.2 Konsep microphone array	6
Gambar 2.3 Mikrofon kondensor	7
Gambar 2.4 Sensor MAX9814	8
Gambar 2.5 Liquid crystal display (LCD)	9
Gambar 2.6 Board arduino uno	11
Gambar 2.7 Pin mapping ATmega 328	12
Gambar 2.8 Pin <i>Mapping</i> ATmega 328 SMD	13
Gambar 2.9 <i>Neural Network</i> Sederhana	13
Gambar 2.10 Fungsi <i>Sigmoid Biner</i>	16
Gambar 2.11 Tampilan IDE Visual Basic NET 2013	18
Gambar 2.12 Toolbox MS Visual Studio	19
Gambar 3.1 Skema perancangan sistem	21
Gambar 3.2 Blok diagram sistem	22
Gambar 3.3 Rangkaian Skematik Mikrofon MAX9814	24
Gambar 3.4 Mikrofon tampak atas & mikrofon tampak samping	25
Gambar 3.5 (a) Bagian dalam antenna microphone array (b) Bagian atas antenna mikrofon array	26
Gambar 3.6 Letak sensor mikrofon	27
Gambar 3.7 Flowchart arduino uno	28
Gambar 3.8 Flowchart visual studio	33
Gambar 3.9 (a) Antena microphone array tampak depan (b) Antena microphone array tampak atas	34
Gambar 4.1 (a) pengujian mikrofon jarak 10 cm, (b) pengujian mikrofon jarak 20 cm, (c) pengujian mikrofon jarak 30 cm, (d) pengujian mikrofon jarak 40 cm	36
Gambar 4.2 Perbandingan Nilai ADC pada data Power Supply dan data Multimeter	38
Gambar 4.3 Grafik error nilai ADC	38
Gambar 4.4 Tampilan learning Neural Network	42
Gambar 4.5 Grafik Error Learning Neural Network	42
Gambar 4.6 Tampilan pengujian sudut azimuth 0^0	48
Gambar 4.7 Tampilan pengujian sudut azimuth 60^0	49
Gambar 4.8 Tampilan pengujian sudut azimuth 120^0	50

Gambar 4.9 Tampilan pengujian sudut azimuth 180^0	50
Gambar 4.10 Tampilan pengujian sudut azimuth 240^0	51
Gambar 4.11 Tampilan pengujian sudut azimuth 300^0	52

DAFTAR TABEL

Tabel 2.1 Tabel fungsi LCD	10
Tabel 4.1 Pengujian ke 4 mikrofon.....	35
Tabel 4.2 Pembacaan nilai ADC	37
Tabel 4.3 Pengujian sudut azimuth 0^0	39
Tabel 4.4 Pengujian sudut azimuth 60^0	40
Tabel 4.5 Pengujian sudut azimuth 120^0	40
Tabel 4.6 Pengujian sudut azimuth 180^0	40
Tabel 4.7 Pengujian sudut azimuth 240^0	41
Tabel 4.8 Pengujian sudut azimuth 300^0	41
Tabel 4.9 Hasil pengujian sudut azimuth dengan jarak suara drone 10 cm dan sudut elevasi 0^0	43
Tabel 4.10 Hasil pengujian sudut azimuth dengan jarak suara drone 20 cm dan sudut elevasi 0^0	44
Tabel 4.11 Hasil pengujian sudut azimuth dengan jarak suara drone 10 cm dan sudut elevasi 30^0	45
Tabel 4.12 Hasil pengujian sudut azimuth dengan jarak suara drone 20 cm dan sudut elevasi 30^0	45
Tabel 4.13 Hasil pengujian sudut azimuth dengan jarak suara drone 10 cm dan sudut elevasi 60^0	46
Tabel 4.14 Hasil pengujian sudut azimuth dengan jarak suara drone 20 cm dan sudut elevasi 60^0	47

Halaman Ini Sengaja Dikosongkan

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pesawat tak berawak atau yang biasa disebut drone dengan cepat menjadi bahan pokok masyarakat modern. Beragam penggunaan teknologi drone adalah mulai dari penggunaan komersial, militer dan rekreasi [1]. Isu tersebut telah menunjukkan bahwa dalam banyak hal begitu banyak pihak terkemuka yang mengakui akan keunggulan penggunaan drone, dan kemajuan mereka menjadi teknologi yang canggih dan serbaguna, menghadirkan ancaman terbesar bagi masyarakat serta ketersediaan dan kemampuan masyarakat yang harus menyesuaikan dengan perkembangan teknologi modern sekarang ini.

Pernyataan ini mulai diakui di seluruh dunia, dengan penyalahgunaan teknologi drone yang meningkat. Contoh penyalahgunaan drone yang menonjol berasal dari upaya penjahat untuk menghindari sistem pertahanan yang ada di lingkungan yang aman. Sejumlah kejadian terungkap mengilustrasikan drone ini digunakan untuk mencoba menyelundupkan selundupan ke penjara, drone juga telah semakin terbiasa memasuki wilayah udara yang terbatas, seperti bandara, dan pangkalan militer sebagai mata-mata yang tentunya sangat membahayakan.

Dalam tugas akhir ini telah dilakukan pengembangan alat untuk mengatasi pencegahan tentang penyalahgunaan drone yang lebih efektif dan efisien dengan cara mendeteksi posisi dari drone berdasarkan sinyal suara menggunakan 4 buah sensor mikrofon, tegangan keluaran dari sensor suara dikonversi oleh ADC pada mikrokontroler Arduino Uno kemudian data sudut yang didapat akan ditampilkan pada LCD atau tampilan pada komputer.

1.2 Rumusan Masalah

Sehubungan dengan latar belakang yang telah dijelaskan sebelumnya, terdapat beberapa masalah yang akan dibahas antara lain sebagai berikut :

1. Bagaimana suara drone dapat dideteksi.
2. Bagaimana cara mengolah sinyal suara drone untuk menjadi nilai *input* data dan metode apa yang digunakan.
3. Bagaimana menentukan posisi dari suara drone.

1.3 Batasan Masalah

Batasan masalah yang akan dibahas dalam Tugas Akhir ini adalah :

1. Penggunaan mikrofon yang berjumlah 4 buah dan dianggap yang terdengar hanya suara drone.
2. Penggunaan Arduino Uno sebagai pengolah data dan metode decision tree yang diaplikasikan pada mikrokontroler..
3. Posisi suara drone yang dideteksi yaitu sudut 0^0 , 60^0 , 120^0 , 180^0 , 240^0 , 300^0 untuk sudut azimuth dan 30^0 , 60^0 , dan 90^0 untuk sudut elevasi.

1.4 Tujuan

Tujuan dari penelitian ini adalah merancang dan merealisasikan sebuah prototype antena microphone array agar memiliki kemampuan mendeteksi suara drone dan dapat dikembangkan untuk keperluan alutsista militer.

1.5 Metodologi

Metodologi yang digunakan dalam penyusunan penelitian ini sebagai berikut :

1. Studi literatur dan diskusi, yaitu studi yang bersumber pada jurnal-jurnal, buku referensi, *paper* dan *datasheet* komponen yang digunakan dalam penelitian ini serta berdiskusi dengan dosen pembimbing dan narasumber.
2. Perancangan dan realisasi alat, pada tahap ini dimulai dari desain antena mikrofon sebagai sensor suara dan Arduino uno sebagai mikrokontroler untuk memerhitungkan arah sudut dari suara drone serta penggunaan LCD dan Microsoft Visual Studio sebagai output display pada komputer.

3. Tahap pengujian sistem dan analisa, pada tahapan ini dimulai dengan memberi suara drone pada 4 mikrofon dengan jarak – jarak tertentu dan membandingkan data nilai yang diterima pada masing-masing mikrofon, kemudian perbandingan tersebut dapat menentukan posisi sudut dari drone.
4. Pembuatan laporan mengacu pada perancangan dan realisasi alat, serta hasil dari pengujian alat.

1.6 Sistematika Penulisan

Sistematika penulisan pada penelitian ini dibagi menjadi beberapa bab dengan rincian :

BAB I : PENDAHULUAN

Berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, metodologi dan sistematika penulisan.

BAB II : TEORI PENUNJANG

Berisi tentang penjelasan dan dasar teori dari drone, *Sound Localization*, mikrofon, LCD, Arduino Uno, *Neural Network*, dan Microsoft Visual Studio.

BAB III : PERANCANGAN DAN REALISASI ALAT

Berisi tentang tahap-tahap perancangan perangkat hardware dan software.

BAB IV : PENGUJIAN DAN ANALISA

Bab ini membahas mengenai pengujian dari rancangan yang telah diimplementasikan pada prototipe pendeteksi posisi drone.

BAB V : PENUTUP

Berisi tentang kesimpulan dan saran yang diperoleh dalam penelitian ini.

1.7 Relevansi

Adapun manfaat yang diharapkan dengan penelitian ini adalah terciptanya suatu alat yang dapat mendeteksi drone berdasarkan sinyal suara dengan menggunakan teori mikrofon array serta menggunakan 4 buah sensor mikrofon dan mikrokontroller Arduino Uno.

Hasil yang dicapai diharapkan dapat menjadi salah satu referensi dalam bidang militer untuk pengembangan alutsista sistem pendeteksi posisi drone.

Halaman Ini Sengaja Dikosongkan

BAB II

TEORI PENUNJANG

2.1 Drone

Drone merupakan pesawat tanpa pilot. Pesawat ini dikendalikan secara otomatis melalui program komputer yang dirancang, atau melalui kendali jarak jauh dari pilot yang terdapat di dataran atau di kendaraan lainnya [1,2]. Awalnya drone merupakan pesawat yang dikendalikan jarak jauh, namun sistem otomatis kini mulai banyak diterapkan. Menurut sejarahnya, ide pengembangan pesawat tanpa pilot sudah ada sejak 22 Agustus 1849. Waktu itu, Austria berusaha menyerang kota Venesia di Italia dengan menggunakan balon tak berawak yang penuh akan bahan peledak. Cara kerja drone sederhana ini tidak sepenuhnya berhasil. Beberapa balon mengenai sasaran, tetapi ada pula yang terjebak angin dan berubah arah. Perkembangan teknologi membuat drone juga mulai banyak diterapkan untuk kebutuhan sipil, seperti untuk pemantauan lalu lintas, pemetaan lahan, investigasi dan sebagainya.

Teknologi drone sendiri sudah berkembang jauh sebelum perang dunia pertama untuk kepentingan militer dan terus mengalami perkembangan yang signifikan hingga saat ini. Istilah Drone akhirnya menjadi tenar untuk menyebut UAV. Drone atau pesawat tanpa awak penumpang secara umum disebut sebagai unmanned aerial vehicle (UAV) yang bisa terbang secara otomatis dengan sistem pengendali di dalamnya. Drone tersebut bisa dikendalikan untuk mengenai jarak tempuh serta ketinggiannya. Namun dalam perkembangannya penyalahgunaan drone sangat meningkat misalnya drone digunakan sebagai alat pengintai, hal ini tentu sangat berbahaya dan melanggar peraturan dalam penggunaan drone.



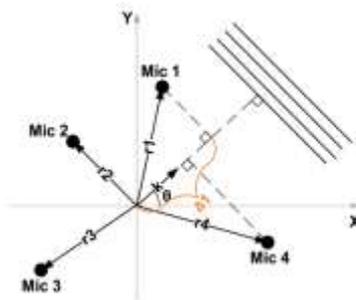
Gambar 2.1 Drone

2.2 Sound Localization

Sound localization adalah proses menentukan lokasi sumber suara berdasarkan beberapa observasi yang diterima pada sinyal suara [3]. Salah satu permasalahan dalam aplikasi deteksi sumber suara adalah kesulitan untuk mengestimasi sudut datangnya gelombang dari *planewave*. Hal ini berkaitan dengan permasalahan mencari arah penjarangan gelombang atau estimasi *DOA (Direction Of Arrival)*. Karena permasalahan tersebut, maka topik tentang estimasi *DOA* mendapat perhatian dari berbagai peneliti beberapa tahun ini. Metode yang telah dilakukan untuk mengatasi permasalahan tersebut adalah dengan memanfaatkan komponen transduser seperti *array microphone* dalam mencari sudut datangnya gelombang atau *DOA*.

2.2.1 Microphone Array

Microphone array merupakan proses penyaluran suara dengan menggunakan banyak mikrofon yang penempatan letaknya berbeda-beda [4,5]. Algoritma yang digunakan dalam modul pemrosesan didasarkan pada penentuan dua mikrofon, mana yang memiliki waktu tunda terbaik di antara mereka dalam merekam sinyal suara yang masuk dari drone. Begitu kedua mikrofon ditentukan, arah dengung dapat disimpulkan sebagai garis lurus yang mengarah ke arah mikrofon yang menerima sinyal suara terlebih dahulu. Ini pertama kali dilakukan antara 3 mikrofon pada dasar array, menentukan sudut azimuth, dan kemudian di antara masing-masing mikrofon di dasar dan 1 mikrofon tambahan di atas, tentukan sudut elevasi.



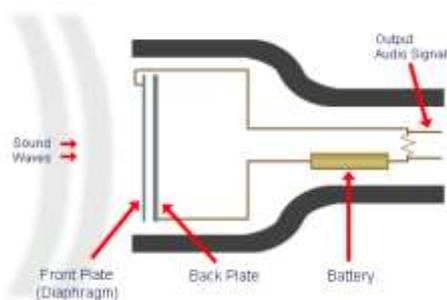
Gambar 2.2 Konsep Microphone Array

2.3 Mikrofon

Mikrofon adalah alat yang digunakan untuk mengubah besaran suara menjadi besaran listrik. Mikrofon memiliki banyak macam, seperti: mikrofon kristal, mikrofon elektrodinamik, dan mikrofon kondensor seperti yang ditunjukkan pada Gambar 2.3. Mikrofon kondensor merupakan komponen elektronik yang menyimpan energi dalam medan elektrostatis [6]. Mikrofon jenis ini menggunakan bahan dasar berupa kapasitor. Diperlukan catu daya bahan berupa baterai untuk pengoperasian mikrofon jenis ini.

Dalam mikrofon kondensor terdapat dua keping plat yang memiliki beda tegangan. Salah satu dari keping plat terbuat dari bahan yang ringan sehingga bertindak sebagai diafragma. Fungsi dari diafragma adalah mendeteksi getaran suara dari lingkungan dengan beresonansi dengan getaran tersebut. Ketika beresonansi dengan getaran suara, jarak antar dua keping plat berubah-ubah sehingga menghasilkan nilai kapasitansi yang berubah-ubah. Nilai kapasitansi berbanding lurus dengan jumlah muatan pada kapasitor. Jumlah muatan berbanding lurus dengan besar arus yang dihasilkan oleh mikrofon kondensor. Pada bagian keluaran mikrofon terdapat resistor yang menyebabkan tegangan berubah ketika mendapatkan perubahan arus.

Mikrofon kondensor lebih sensitif terhadap suara jika dibandingkan dengan mikrofon dinamis. Akibat dari sensitifitas yang tinggi mikrofon kondensor rentan terhadap derau.

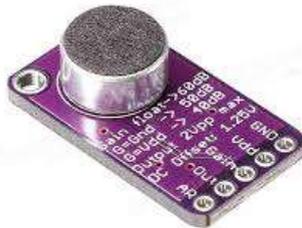


Gambar 2.3 Mikrofon Kondensor

2.3.1 Sensor Mikrofon Max9814

Sensor MAX9814 adalah penguat mikrofon berbiaya rendah dan berkualitas tinggi dengan kontrol gain otomatis (AGC) dan bias mikrofon dengan noise rendah. Perangkat ini dilengkapi dengan preamplifier low noise, amplifier penguat variabel (VGA), penguat keluaran, generator tegangan bias mikrofon dan sirkuit kontrol AGC [7].

Preamplifier dengan noise rendah memiliki gain 12 dB tetap, sedangkan VGA gain secara otomatis menyesuaikan dari 20 dB menjadi 0 dB, tergantung pada tegangan output dan ambang AGC. Penguat keluaran menawarkan keuntungan yang dapat dipilih dari 8 dB, 18 dB, dan 28 dB. Dengan tidak adanya kompresi, runcing amplifier menghasilkan keuntungan keseluruhan 40 dB, 50 dB, atau 60 dB. Program input digital trilevel menghasilkan penguat keluaran. Pembagi resistif eksternal mengendalikan ambang AGC dan satu unit kapasitor menyerang / melepaskannya. Program input digital trilevel rasio waktu serangan-ke-pelepasan. Waktu penahanan AGC ditetapkan pada 30 ms. Generator dengan tegangan bias rendah-bias-bias bisa menghasilkan sebagian besar mikrofon electret. MAX9814 tersedia dalam paket TDFN 14-pin hemat-ruang. Perangkat ini ditentukan selama rentang suhu diperpanjang -40°C sampai $+85^{\circ}\text{C}$. Di bawah ini adalah bentuk fisik dari MAX9814.



Gambar 2.4 Sensor Mikrofon MAX9814

2.4 LCD

LCD merupakan singkatan dari Liquid Crystal Display yang dapat digunakan untuk menampilkan berbagai hal berkaitan dengan aktivitas mikrokontroler, salah satunya adalah menampilkan teks yang terdiri dari berbagai karakter [8]. LCD banyak digunakan karena fungsinya yang bervariasi, dan juga pemrogramannya yang mudah. Untuk dapat menghubungkan LCD dengan mikrokontroler, PORT pada LCD perlu dihubungkan dengan PORT yang sesuai dengan PORT pada mikrokontroler.

Material dari LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan seven-segment dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan sandwich memiliki polarizer cahaya vertikal depan dan polarizer cahaya horisontal belakang yang diikuti dengan lapisan reflektor. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segmen yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan. Bentuk fisik dari LCD 2x16 bisa dilihat pada gambar 2.5 dibawah ini.



Gambar 2.5 Liquid crystal display

Tabel 2.1 Tabel Fungsi LCD

PIN ASSIGNMENT		
Pin no	Symbol	Function
1	Vss	Power Supply (GND)
2	Vdd	Power Supply (+)
3	Vo	Contrast Adjust
4	RS	Register select signal
5	R/W	Data read / write
6	E	Enable signal
7	DB0	Data bus line
8	DB1	Data bus line
9	DB2	Data bus line
10	DB3	Data bus line
11	DB4	Data bus line
12	DB5	Data bus line
13	DB6	Data bus line
14	DB7	Data bus line
15	A	Power supply for LED B/L (+)
16	K	Power supply for LED B/L (-)

Pada LCD dengan 16 pin, dapat dilihat fungsi-fungsi setiap pin yang dijelaskan pada Tabel 2.1.

2.5 Arduino Uno

Arduino merupakan *board microcontroller* yang berbasis pada *microcontroller* tipe Atmega, Arduino diprogram dengan menggunakan bahasa khusus, dan dengan struktur yang sama dengan bahasa C, dengan bahasa tersebut maka pemrograman menjadi lebih mudah. Arduino tipe uno merupakan versi terbaru, berbasis atmega 328, yang menyempurnakan tipe sebelumnya, *Duemilanove*. Perbedaan mendasarnya ialah tidak menggunakan IC FTDI USB to Serial sebagai driver komunikasi USBnya tetapi menggunakan *microcontroller* atmega 8U2 yang diprogram sebagai *converter* USB ke serial [9]. Uno sendiri diambil dari bahasa italia yang artinya satu. Berikut adalah bentuk fisik dari board Arduino Uno seperti pada gambar 2.6 dibawah ini:



Gambar 2.6 Board Arduino Uno

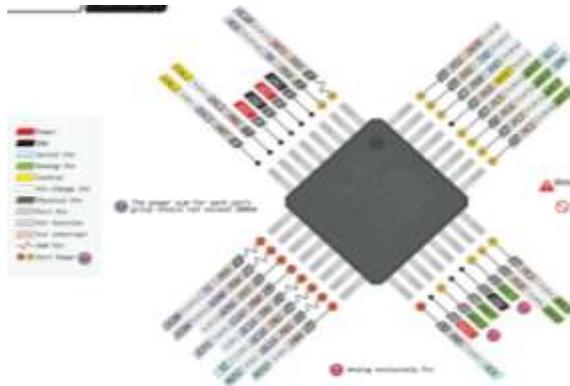
2.5.1 ATmega328

ATmega328 adalah mikrokontroler 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*) [10].

ATmega328 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 16 MIPS dengan *clock* 16 MHz.
4. 32 KB *Flash memory* dan pada arduino memiliki *bootloader* yang menggunakan 2 KB dari flash memori 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 catu daya dimatikan.
6. Memiliki SRAM (*Static Random Access Memory*) sebesar 2KB.
7. Memiliki pin I/O digital sebanyak 14 pin 6 diantaranya PWM (*Pulse Width Modulation*) output.
8. Master / Slave SPI *Serial interface*.

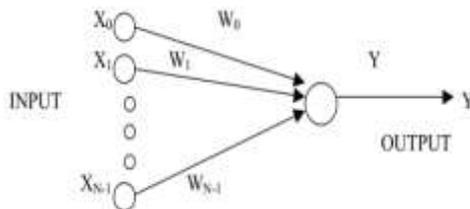
Mikrokontroler ATmega 328 memiliki arsitektur Harvard, yaitu memisahkan memori untuk kode program dan memori



Gambar 2.8 Pin Mapping ATmega 328 SMD (Surface Mount Device)

2.6 Neural Network

Neural network merupakan sistem analisa berdasarkan operasi jaringan neural biologi dengan prinsip kerja mirip dengan cara kerja otak manusia. *Backpropagation* merupakan metode pembelajaran yang sering digunakan karena kemudahan perancangan dengan ketelitian yang tinggi. Menurut Kevin Gurney, sebuah *neural network* adalah pertemuan saling berhubungan dari elemen pemrosesan sederhana, unit atau node, yang fungsi secara bebas didasarkan pada neuron hewan. Itu pengolahan kemampuan jaringan disimpan dalam antar-satuan kekuatan sambungan, atau bobot, diperoleh dengan proses adaptasi, atau belajar dari, satu set pelatihan pola [11,12].



Gambar 2.9. Neural Network Sederhana

2.6.1 Konsep Dasar Neural Network

Neural network terdiri dari beberapa neuron dan ada hubungan antar neuron-neuron seperti pada otak manusia. Neuron/sel saraf adalah sebuah unit pemroses informasi yang merupakan dasar operasi *neural network*. *Neural network* terdiri atas beberapa elemen penghitung tak linier yang masing-masing dihubungkan melalui suatu pembobot dan tersusun secara paralel. Pembobot inilah yang nantinya akan berubah (beradaptasi) selama proses pelatihan.

Pelatihan perlu dilakukan pada suatu *neural network* sebelum digunakan untuk menyelesaikan masalah. Hasil pelatihan *neural network* dapat diperoleh tanggapa yang benar (yang diinginkan) terhadap masukan yang diberikan. *Neural network* dapat memberikan tanggapan yang benar walaupun masukan yang diberikan terkena derau atau berubah oleh suatu keadaan [12].

2.6.2 Karakteristik Neural Network

Penyelesaian masalah dengan *neural network* tidak memerlukan pemrograman. *Neural network* menyelesaikan masalah melalui proses belajar dari contoh-contoh pelatihan yang diberikan. Biasanya pada *neural network* diberikan sebuah himpunan pola pelatihan yang terdiri dari sekumpulan contoh pola. Proses belajar *neural network* berasal dari serangkaian contoh-contoh pola yang diberikan. metode pelatihan yang sering dipakai adalah metode belajar terbimbing. Selama proses pembelajaran pola masukan disajikan bersama-sama dengan pola keluaran yang diinginkan. Jaringan akan menyesuaikan nilai bobotnya sebagai tanggapan atas pola masukan dan sasaran yang disajikan tersebut.

2.6.3 Faktor Bobot

Bobot merupakan suatu nilai yang mendefinisikan tingkat atau kepentingan hubungan antara suatu node dengan node yang lain. Semakin besar bobot suatu hubungan menandakan semakin pentingnya hubungan kedua node tersebut. Bobot merupakan suatu hubungan berupa bilangan real maupun integer, tergantung dari jenis permasalahan dan model yang digunakan. Bobot-bobot tersebut bisa ditentukan untuk berada

didalam interval tertentu. selama proses pelatihan, bobot tersebut dapat menyesuaikan dengan pola-pola input.

Jaringan dengan sendirinya akan memperbaiki diri terus-menerus karena adanya kemampuan untuk belajar. Setiap ada suatu masalah baru, jaringan dapat belajar dari masalah baru tersebut, yaitu dengan mengatur kembali nilai bobot untuk menyesuaikan karakter nilai.

2.6.4 Fungsi Aktivasi

Setiap neuron mempunyai keadaan internal yang disebut level aktivasi atau level aktivitas yang merupakan fungsi input yang diterima. Secara tipikal suatu neuron mengirimkan aktivitasnya kebeberapa neuron lain sebagai sinyal. Yang perlu diperhatikan adalah bahwa neuron hanya dapat mengirimkan satu sinyal sesaat, walaupun sinyal tersebut dapat dipancarkan ke beberapa neuron yang lain.

Ada beberapa pilihan fungsi aktivasi yang digunakan dalam metode *backpropagation*, seperti fungsi sigmoid biner, dan sigmoid bipolar. Karakteristik yang harus dimiliki fungsi aktivasi tersebut adalah kontinyu, diferensiabel, dan tidak menurun secara monoton. Fungsi aktivasi diharapkan dapat mendekati nilai-nilai maksimum dan minimum secara baik.

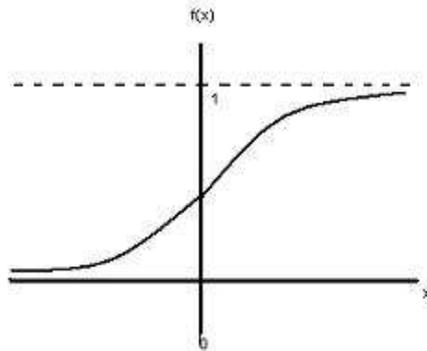
2.6.5 Fungsi Sigmoid Biner

Fungsi ini digunakan untuk *neural network* yang dilatih dengan menggunakan metode *backpropagation*. Fungsi sigmoid biner memiliki nilai pada range 0 sampai 1. Fungsi ini sering digunakan untuk *neural network* yang membutuhkan nilai output yang terletak pada interval 0 sampai 1. Definisi fungsi sigmoid biner adalah sebagai berikut:

$$f_1(x) = \frac{1}{(1+e^{-x})} \dots\dots\dots(2.1)$$

dengan turunan :

$$f_1'(x) = f_1(x)(1 - f_1(x)) \dots\dots\dots(2.2)$$



Gambar 2.10. Fungsi *Sigmoid Biner*

2.6.6 Model Neural Network Backpropagation

Pelatihan pada *neural network backpropagation, feedforward* (umpan maju) dilakukan dalam rangka perhitungan bobot sehingga pada akhir pelatihan akan diperoleh bobot-bobot yang baik [13]. Selama proses pelatihan, bobot-bobot diatur secara iteratif untuk meminimumkan *error* (kesalahan) yang terjadi. *Error* (kesalahan) dihitung berdasarkan rata-rata kuadrat kesalahan (MSE). Rata-rata kuadrat kesalahan juga dijadikan dasar perhitungan untuk kerja fungsi aktivasi. Sebagian besar pelatihan untuk jaringan *feedforward* (umpan maju) menggunakan gradien dari fungsi aktivasi untuk menentukan bagaimana mengatur bobot-bobot dalam rangka meminimumkan kinerja. Gradien ini ditentukan dengan menggunakan suatu teknik yang disebut *backpropagation*. Algoritma pelatihan standar *backpropagation* akan menggerakkan bobot dengan arah gradien negatif. Prinsip dasar dari algoritma *backpropagation* adalah *memperbaiki* bobot-bobot jaringan dengan arah yang membuat fungsi aktivasi menjadi turun dengan cepat.

2.7 Microsoft Visual Studio

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasi lainnya dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web [14]. Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe. Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam native code (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun managed code (dalam bentuk Microsoft Intermediate Language di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows Mobile (yang berjalan di atas .NET Compact Framework).

2.7.2 Pengenalan Integrated Development Environment (IDE) VB.NET 2013

Untuk mengembangkan aplikasi VB.NET dapat menggunakan editor Visual Studio IDE (Integrated Development Environment). Penggunaan IDE bertujuan untuk mempermudah mengembangkan aplikasi sehingga lebih cepat dan produktif. Visual Studio merupakan IDE yang sangat Lengkap dan dapat menjadi lebih produktif, namun fitur-fitur yang ada di Visual Studio harus dipelajari secara detail agar dapat memanfaatkannya secara lebih maksimal.



Gambar 2.11 Tampilan IDE Visual Basic .NET 2013

Visual Studio IDE memiliki beberapa menu, antara lain:

a. Title Bar

Berfungsi untuk menampilkan nama project yang aktif atau sedang dikembangkan.

b. Menu Bar

Bagian menu utama terdapat 11 menu yang mana tidak semua menu itu terpakai oleh kita dalam merancang program, berikut ini beberapa menu yang sering kita pakai :

- a) File : Berisikan tentang manipulasi file seperti project, file dan komponen lainnya.
- b) Edit : Berisikan tindakan-tindakan untuk melakukan perubahan terhadap sebuah objek.
- c) View : Berisikan dengan fungsi-fungsi untuk melihat kondisi dan properties objek-objek.

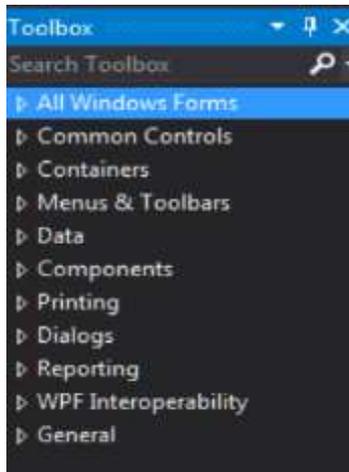
- d) Tools : Berisikan dengan penyediaan alat-alat untuk melakukan pemrograman atau yang berhubungan dengan pemrograman.
- e) Windows : Berisikan tentang pencantuman dan pemilihan window-window yang ditampilkan.
- f) Help : Berisikan fasilitas bantuan.

c. Form

Objek utama berfungsi sebagai tempat meletakkan objek-objek atau komponen yang digunakan dalam merancang sebuah tampilan antar muka pada program aplikasi.

d. Toolbox

Terdapat disebelah kiri tampilan IDE, berisi objek kontrol yang digunakan untuk mendesain antar muka grafis. Pada Visual Studio 2013 pengorganisasian objek kontrol lebih rapi dan teratur sehingga lebih mudah untuk menemukan objek kontrol yang diinginkan.



Gambar 2.12 Toolbox MS Visual Studio

Halaman ini sengaja dikosongkan

BAB III PERANCANGAN DAN REALISASI ALAT

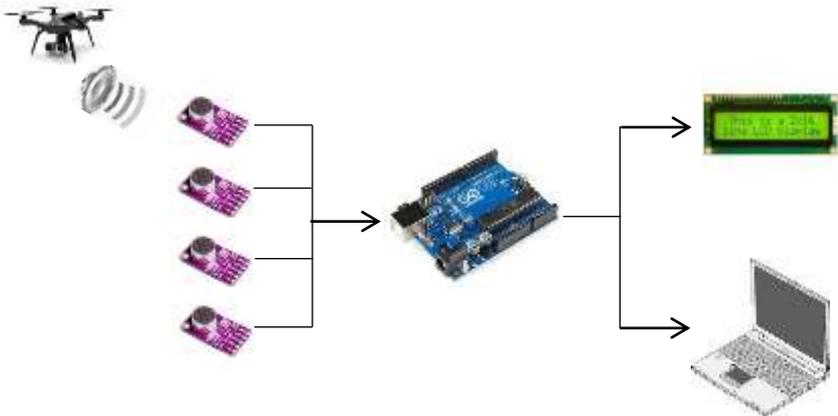
Bab ini membahas mengenai perancangan keseluruhan bagian dan realisasi alat yang meliputi perancangan sistem, diagram blok, perencanaan perangkat keras (*hardware*) dan perangkat lunak (*software*). Perancangan *hardware* alat akan dibahas perbagian. Sedangkan penjelasan *software* akan dijelaskan mengenai pembuatan program untuk mendeteksi suara dari drone dan output berupa display pada LCD dengan menggunakan program arduino serta tampilan di komputer menggunakan Microsoft Visual Studio.

Pada pembahasan bab ini akan dibagi menjadi tiga bagian yaitu:

1. Diagram blok sistem sebagai alur dari cara kerja sistem.
2. Perancangan *hardware* dengan membuat desain alat.
3. Perancangan *software* meliputi *flowchart* program arduino, *flowchart* program visual studio , display pada LCD, dan mencari nilai sudut azimuth posisi drone yang ditampilkan pada komputer,

3.1 Perancangan Sistem

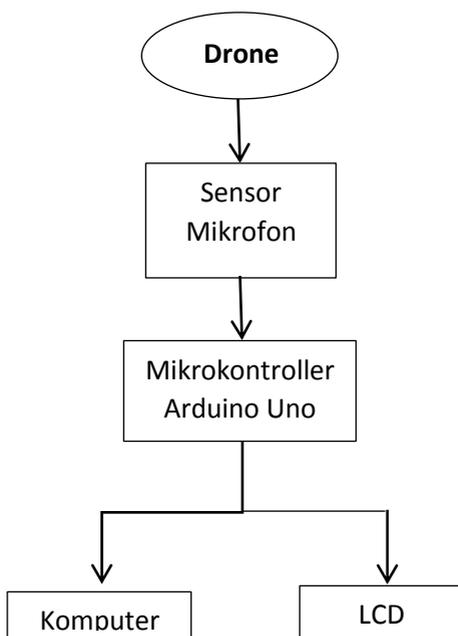
Secara umum sistem yang akan dibangun pada penelitian ini dapat dilihat pada gambar 3.1 berikut ini:



Gambar 3.1 Skema Perancangan Sistem

Pada gambar 3.1 ditunjukkan bahwa suara dari drone merambat melalui udara yang akan ditangkap oleh keempat mikrofon dengan jarak tertentu, lalu akan terdapat perbedaan atau selisih amplitudo pada masing-masing mikrofon karena jarak dari sumber suara drone dengan keempat mikrofon berbeda, jika mikrofon 1 lebih dekat dari sumber suara daripada mikrofon 2, mikrofon 3, atau mikrofon 4 maka waktu sampainya suara dari sumber suara menuju mikrofon 1 akan lebih cepat dibanding mikrofon 2, mikrofon 3, atau mikrofon 4 begitu juga sebaliknya, setelah perbedaan amplitudonya didapat maka digunakanlah perbandingan besarnya amplitudo pada masing-masing mikrofon, dari perbandingan tersebut terdapat mikrofon yang amplitudonya lebih dominan dari yang lain sehingga sudut dari posisi drone dapat dideteksi .

3.2 Blok Diagram



Gambar 3.2 Blok Diagram Sistem

Pada blok diagram tersebut pemrosesan sinyal suara drone yang diterima oleh sensor mikrofon dibaca oleh ADC pada mikrokontroler sehingga sinyal analog dapat digunakan untuk proses perbandingan selisih amplitudo sehingga sudut azimuth dan sudut elevasinya dapat diketahui, setelah sudutnya didapat mikrokontroler akan mengeluarkan output berupa tampilan hasil sudut akhir pada LCD dan pada computer yang menunjukkan sudut azimuth & sudut elevasi dari posisi drone tersebut.

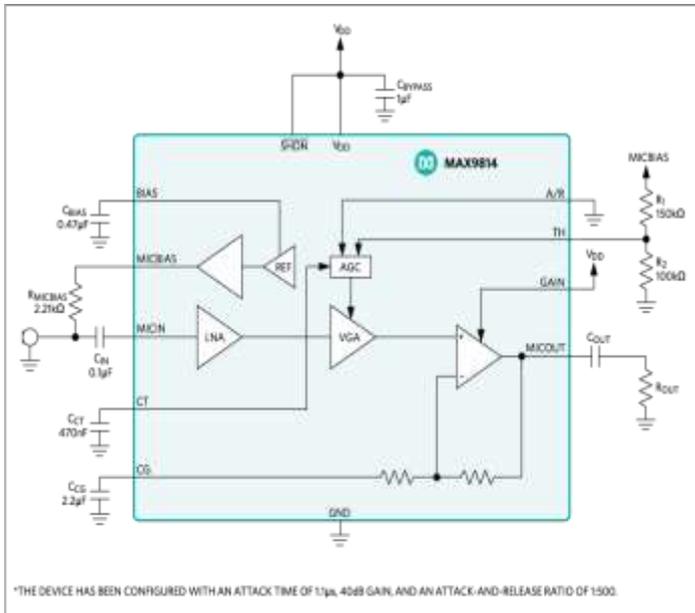
3.3 Perancangan *Hardware*

Perancangan *hardware* akan dilakukan dengan pembuatan rangkaian skematik controller pada software eagle, desain antena microphone array untuk penggabungan komponen agar bisa diletakkan pada satu tempat yang sama.

3.3.1 Mikrofon MAX9814

Mikrofon MAX9814 adalah mikrofon berbiaya rendah dan berkualitas tinggi amplifier dengan automatic gain control (AGC) dan Bias mikrofon dengan noise rendah. Fitur perangkat a low noise preamplifier, variable gain amplifier (VGA), output amplifier, microphone-bias-voltage generator, dan AGC sirkuit control Preamplifier dengan noise rendah memiliki gain 12 dB tetap, sementara VGA gain secara otomatis menyesuaikan dari 20 dB ke 0 dB, tergantung pada tegangan output dan ambang AGC.

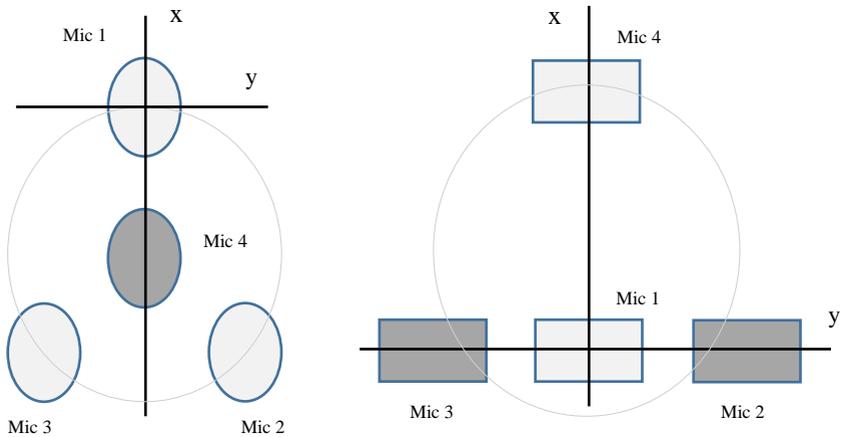
Penguat keluaran menawarkan keuntungan yang dapat dipilih dari 8 dB, 18 dB, dan 28 dB. Tanpa kompresi, kaskade dari Amplifier menghasilkan keuntungan keseluruhan 40 dB, 50 dB, atau 60 dB. Program input digital trilevel menghasilkan penguat keluaran, Berikut adalah rangkaian skematik dari mikrofon MAX9814



Gambar 3.3 Rangkaian Skematik Mikrofon MAX9814

3.3.2 Perancangan Desain Antena Microphone Array

Pada perancangan desain ini penulis menggunakan desain antena microphone array yang sudah ada sebagai referensi. Akan tetapi desain juga disesuaikan dengan bentuk dan ukuran komponen yang digunakan. Dalam perencanaannya, bentuk dasar sistem berupa sebuah prototipe antena microphone array dengan 3 dimensi yang memiliki 4 ruas penampang sensor, dengan 3 mikrofon dibawah dengan posisi horizontal dan 1 mikrofon dengan posisi vertikal. Sedangkan pada pengujian nanti akan menggunakan kemampuan untuk mendeteksi posisi suara drone dengan kenaikan sudut azimuth sebesar 60° yaitu sudut 0° , 60° , 120° , 180° , 240° , 300° , 360° dan arah sudut elevasi dengan kenaikan sebesar 30° yaitu sudut 0° , 30° , 60° , dan 90° . Berikut adalah ilustrasi yang menggambarkan tata letak dan dimensi mikrofon:



Gambar 3.4 Mikrofon tampak atas & mikrofon tampak samping

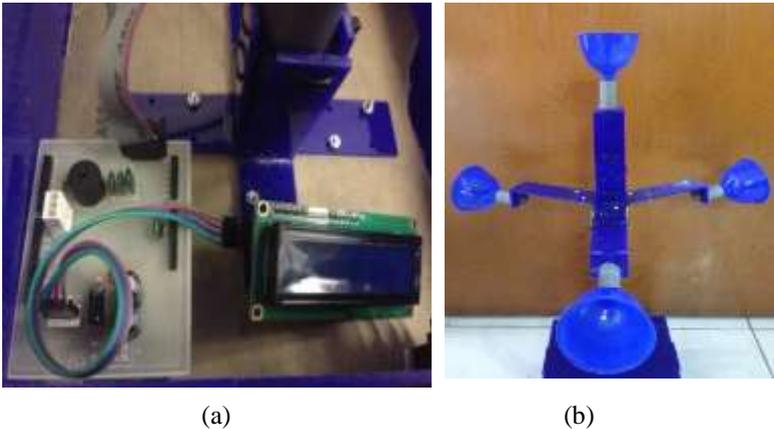
Tata letak mikrofon secara tetrahedral diadopsi pada prototipe, yaitu dengan salah satu mikrofon ditempatkan di atas bidang yang dibentuk oleh tiga mikrofon lainnya. Hal ini diputuskan menyusul penemuan penelitian yang menunjukkan bahwa tetrahedron adalah tata letak 3D yang optimal untuk empat mikrofon (B Yang, 2005). Tata letak ini secara signifikan memperbaiki keakuratan sudut elevasi yang dihitung, karena penundaan waktu antara mikrofon atas dan tiga lainnya cukup besar.

Untuk mengurangi latar belakang gangguan suara dari luar, maka sistem menyaring noise di luar rentang frekuensi yang diharapkan. Untuk menentukan rentang ini, analisis spektrum frekuensi sampel suara drone dilakukan pada aplikasi FFT pada smartphone dengan beberapa kali pengujian.

Empat kali pengujian suara drone digunakan untuk menganalisis spektrum frekuensi, yang secara konsisten menghasilkan kelompok puncak yang ditemukan pada 4,2 kHz. Oleh karena itu, rentang frekuensi operasi diputuskan dari 3,7 kHz – 4,7 kHz, hal ini dilakukan untuk mengaplikasikan kriteria desain yang merekomendasikan perangkat beroperasi lebih dari

rentang 4,7 kHz, memungkinkan drone yang berbeda, dengan berbagai ukuran, sehingga dapat terdeteksi.

Pada bagian bawah antenna akan dibuat sebuah tempat berbentuk kotak yang berfungsi sebagai penopang dari antenna dan dapat menampung beberapa komponen seperti arduino uno, rangkaian filter dan LCD 16x2. Sedangkan pada bagian atas antenna dibuat sesuai desain yaitu memiliki 4 ruas penampang sensor mikrofon dengan panjang ± 15 cm dan tiang antenna dengan panjang ± 30 cm serta pada salah satu ruas terdapat PCB yang terdiri dari kapasitor polyster yang berfungsi sebagai filter pada prototipe, namun fungsi dari kapasitor tersebut kurang maksimal pada ujung sensor diberikan corong agar suara yang diterima oleh mikrofon lebih fokus dan tepat sasaran. Berikut gambar 3.6 adalah bagian atas dari antenna microphone array.



Gambar 3,5 (a) Bagian dalam antenna mikrofon array
(b) Bagian atas antenna mikrofon array

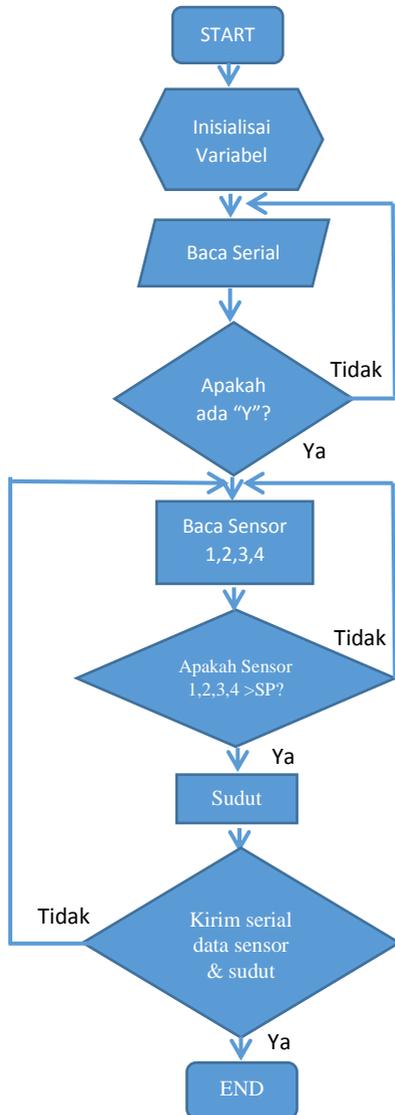


Gambar 3.6 Letak sensor mikrofon

Sedangkan gambar 3.6 adalah letak pemasangan sensor mikrofon. Sensor mikrofon diletakkan diujung masing-masing ruas antenna, sesuai dengan desain yaitu 3 sensor mikrofon dengan posisi horizontal & 1 sensor mikrofon dengan posisi vertikal tegak lurus terhadap posisi mikrofon yang dibawahnya.

3.4 Perancangan Software

Pada tahap perancangan *software* ini, arduino uno akan mendapatkan data dari 4 sensor mikrofon yang sebelumnya telah difilter dahulu kemudian akan diolah dan dikonversi oleh arduino untuk sebagai nilai *input* interface pada komputer. Untuk mengetahui cara kerja program yang digunakan pada sistem ini berikut *flow chart* dari program utama dari alat ini.



Gambar 3.7 Flowchart Arduino Uno

3.4.1 Pengolahan Data Mikrofon

a. *Decision Tree*

Untuk opsi yang pertama yaitu dengan metode Decision tree, data yang diterima dari sensor mikrofon sangat banyak, maka dilakukan *parsing data* atau pemecah data, karena data yang dibutuhkan berupa data sudut dari posisi drone. Setelah data sensor mikrofon didapatkan maka akan dilakukan perhitungan untuk mendapatkan sudut azimuth dan sudut elevasi dari posisi drone.

Sudut azimuth dan sudut elevasi didapatkan dengan menggunakan metode *decision tree*. *Decision Tree* adalah salah satu metode klasifikasi yang menggunakan representasi suatu struktur pohon yang berisi alternatif-alternatif untuk pemecahan suatu masalah. Berikut adalah penulisan program berdasarkan metode decision tree untuk menentukan sudut dari posisi drone.

```
if(sensorValue0 > bagi && sensorValue1 > bagi &&
sensorValue2 < bagi){
    sudut = 60;
} else if(sensorValue0 < bagi && sensorValue1 > bagi &&
sensorValue2 > bagi){
    sudut = 180;
} else if(sensorValue0 > bagi && sensorValue1 < bagi &&
sensorValue2 > bagi){
    sudut = 300;
} else{
    if(sensorValue0 > sensorValue1 && sensorValue0 >
sensorValue2){
        sudut = 0;
    } else if(sensorValue1 > sensorValue0 && sensorValue1 >
sensorValue2){
        sudut = 120;
    } else if(sensorValue2 > sensorValue0 && sensorValue2 >
sensorValue1){
        sudut = 240;
    }

if(sensorValue3 > 400){
    sudut_y = 90;
```

```

}else if(sensorValue3 > 300 && sensorValue3 < 400){
    sudut_y = 60;
}else if(sensorValue3 < 300){
    sudut_y = 30;
}

```

Setelah sudut azimuth dan sudut elevasi dari posisi dari drone diketahui maka data ini akan ditampilkan pada LCD dan komputer.

a. **Neural Network**

Opsi yang kedua menggunakan kontrol *Neural Network*. Proses indentifikasi suara drone dibuat dengan perangkat lunak Matlab. Untuk proses identifikasi terdapat dua sub proses, yaitu proses pengambilan hasil pembacaan ADC dari mikrokontroler dan pengiriman data ke server. Berikut proses identifikasi suara drone secara keseluruhan:

- 1) Pertama kali suara drone diuji dengan berbagai sudut ke sensor, sensor akan mendeteksi besaran masing masing sensor mikrofon berdasarkan nilai tegangan masing masing. Setelah terhubung perangkat lunak dapat membaca nilai dari tegangan ADC. Kemudian dilanjutkan dengan pengenalan pola pada proses *neural network*.
- 2) Proses pengiriman data ke *server*, proses ini merupakan proses pengiriman data tegangan ADC ke server untuk disimpan di *data base* dan sebagai inputan untuk pengenalan suara drone pada proses *neural network*.
- 3) Proses *neural network*, proses ini merupakan pengenalan pola dari suara drone yang diuji yang sebagai masukan atau disebut *input layer*, pada *layer* ini data yang didapatkan dari hasil pembacaan ADC yang diambil sebanyak lima kali (atau variasi sampel sebanyak lima), kemudian nilai yang didapat dirata-rata tiap sampelnya per samplingnya. Kemudian dilakukan proses pelatihan yang bertujuan untuk mendapatkan nilai nilai kesalahan yang sangat kecil

dan berfungsi untuk mendapatkan nilai beban (*weight*) dan nilai *bias* untuk proses selanjutnya. Setelah mendapatkan nilai *weight* dan *bias*, lalu dilakukan dengan cara mengaktifkan lagi sistem keseluruhan, kemudian memberi masukan berupa pembacaan nilai ADC untuk semua sensor mikrofon yang digunakan, pada proses pengenalan pola yang berfungsi untuk mendapatkan hasil proses *neural network* ini dengan menghasilkan nilai akhir pendeteksian sudut dari drone dan menghasilkan kesimpulan dari sistem keseluruhan.

Pada iterasi pertama ini dilakukan perhitungan dengan input dari data sensor yang didapatkan. *Output* dari iterasi pertama ini adalah *hidden layer* yang kedua. Hiden layer kedua terdapat 10 node. Berikut adalah program perhitungan untuk mendapatkan nilai pada iterasi pertama:

```
float ANN_Process() {
    // Reset all Variable to zero
    FlushVar();
    // First Iteration
    for(int i = 0; i < HiddenLayer1; i++) {
        for(int j = 0; j < JumlahData; j++) Synaps[i] += W1[j +
(i*JumlahData)] * InputData[j];
        Synaps[i] += B1[i];
    }
    // Log Sigmoid l Activation
    for(int i = 0; i < HiddenLayer1; i++) Telden[HiddenLayer1]
= 1/(1 + exp(-Synaps[i]));
```

Pada iterasi yang kedua nilai input dari hasil perhitungan aktivasi log sigmoid. Perhitungan pada *hidden layer* yang pertama menggunakan 5 input dan 5 output. Berikut program perhitungan untuk mendapatkan nilai dari aktivasi *log sigmoid*:

```

// Second Iteration
for(int i = 0; i < HiddenLayer2; i++) {
    for(int j = 0; j < HiddenLayer1; j++) Synaps2[i] += W2[j +
(i*HiddenLayer1)] * Telden[HiddenLayer2];
    Synaps[i] += B2[i];
}
// Log Sigmoid 1 Activation
for(int i = 0; i < HiddenLayer2; i++) Telden2[HiddenLayer2]
= 1/(1 + exp(-Synaps2[i]));

//Final Iteration
for(int i = 0; i < HiddenLayer2; i++) Node += W3[i] *
Telden2[HiddenLayer2];
Node += B3;

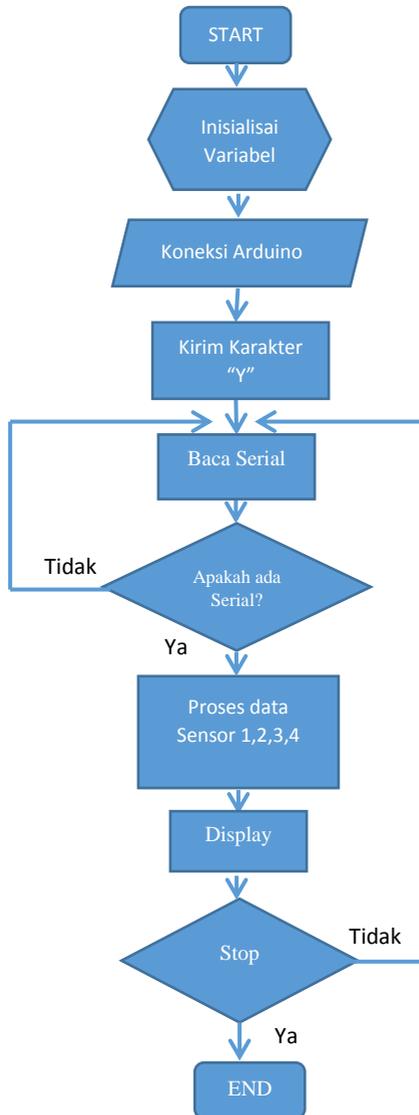
return Node;
}

void FlushVar() {
    memset(Synaps, 0, sizeof(Synaps));
    memset(Telden, 0, sizeof(Telden));
    memset(Synaps2, 0, sizeof(Synaps2));
    memset(Telden2, 0, sizeof(Telden2));
    Node = 0.0;
}

```

3.4.2 Microsoft Visual Studio

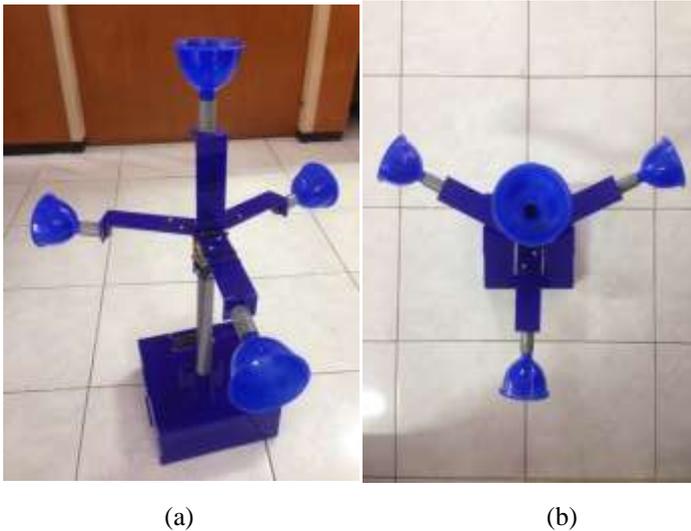
Pada tahap ini, arduino akan memberikan nilai inputan atau data terhadap interface pada PC berdasarkan hasil pengolahan data mikrofon dari antenna microphone array. Arduino uno akan mengirimkan karakter “Y” kemudian dibaca oleh interface dan ditampilkan pada komputer berupa grafik pembacaan sensor serta sudut azimuth dan sudut elevasi dari drone. Berikut flowchart program untuk visual studio.



Gambar 3.8 Flowchart Visual Studio

3.5 Realisasi Alat

Pada tahap ini akan merealisasikan sistem sesuai dengan hasil perancangan *hardware* dan *software* yang telah dijelaskan sebelumnya. Antena microphone array yang sudah didesain sebelumnya kemudian dicetak menggunakan akrilik. Kemudian bagian-bagian yang sudah selesai dicetak, dilakukan perakitan terhadap komponen yang akan digunakan. Berikut pada gambar 3.10 merupakan bentuk Antena microphone array.



Gambar 3.9 (a) Antena microphone array tampak depan
(b) Antena microphone array tampak atas

BAB IV PENGUJIAN DAN ANALISA

Pada bab ini menjelaskan tentang hasil pengujian dan analisa dari perancangan sistem yang telah dilakukan dan dijelaskan pada bab sebelumnya. Pengujian dari sistem pendeteksi posisi arah suara drone ini dilakukan menjadi dua tahap, yaitu pengujian *hardware* dan *software*. Pengujian *hardware* disini berupa pengujian mikrofon dan pengujian ADC mikrokontroler. Sedangkan pengujian *software* disini berupa pengujian dari *software* arduino uno dan *software* visual studio.

4.1 Pengujian Hardware

Pada pengujian *hardware* proyek ini yang diuji adalah mikrofon dan ADC pada mikrokontroler Arduino Uno, pengujian mikrofon disini mencakup pembacaan sinyal suara drone dengan intensitas suara yang berbeda. Selanjutnya pengujian ADC arduino uno dengan pembacaan sinyal suara drone, lalu data analog yang dihasilkan dijadikan grafik data.

4.1.1 Pengujian mikrofon

Dalam proyek ini *hardware* yang digunakan adalah mikrofon kondensor sebagai sensor suara, mikrokontroler Arduino Uno sebagai prosesor utama serta LCD dan komputer sebagai *output*. Mikrofon yang digunakan adalah mikrofon kondensor dengan preamplifier modul MAX 9814 sehingga output langsung berupa sinyal suara yang dapat diolah pada mikrokontroler, tabel 4.1 disini merupakan hasil pengukuran keempat mikrofon dalam membaca sinyal suara drone dengan pengujian sampai dengan jarak 2 meter. Suara drone yang dihasilkan menggunakan *smartphone* yang sebelumnya telah direkam terlebih dahulu.

Tabel 4.1 Pengujian ke 4 mikrofon

No	Jarak (Cm)	Vpp	Vmin	Vmax
1	10	2,04 V	-1,18 V	860 mV
2	20	1,56 V	-780 mV	780 mV
3	30	1,42 V	-780 mV	640 mV
4	40	1,28 V	-780 mV	500 mV



(a)

(b)



(c)

(d)

Gambar 4.1 (a) pengujian mikrofon jarak 10 cm, (b) pengujian mikrofon jarak 20 cm, (c) pengujian mikrofon jarak 30 cm, (d) pengujian mikrofon jarak 40 cm.

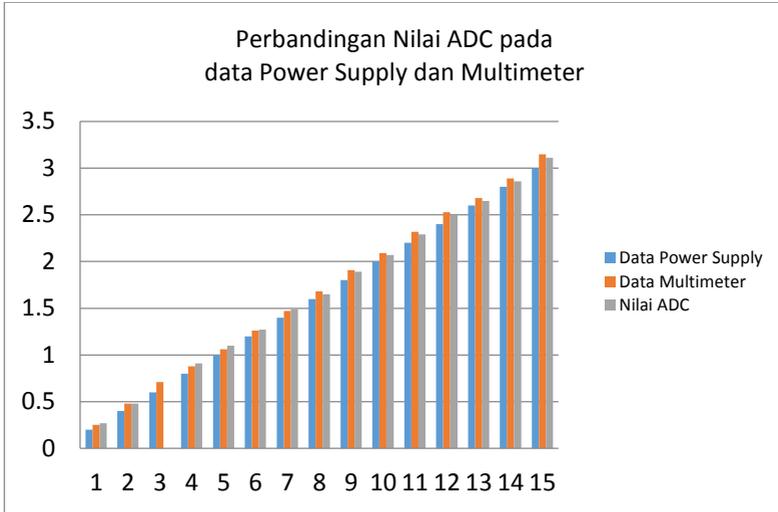
Pengujian mikrofon ini dilakukan dengan menggunakan osiloskop agar bentuk dari sinyal suara drone pada pengujian jarak yang berbeda akan terlihat. Untuk melihat kemampuan mikrofon dalam menangkap suara drone hasilnya dapat dilihat pada gambar 4.1. Pada pengujian mikrofon dengan menggunakan osiloskop dapat dilihat pada jarak yang relatif dekat sinyal drone yang ditangkap masih besar dan pengujian dengan jarak yang jauh sinyal suara dari drone berangsur-angsur mengecil, ini menandakan sensor mikrofon berfungsi sesuai dengan semestinya.

4.1.2 Pengujian ADC Arduino Uno

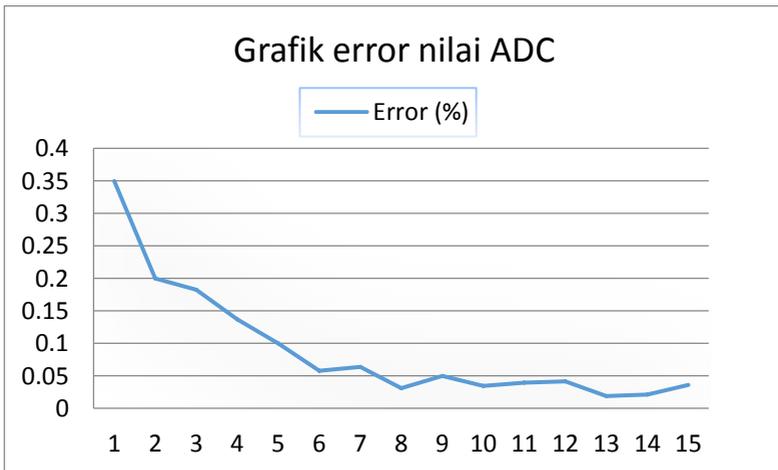
Pengujian selanjutnya adalah melakukan uji coba pada pembacaan data ADC pada mikrokontroler arduino uno yang memiliki resolusi 10 bit. Hal ini diperlukan untuk mengetahui keakuratan pengukuran dan pembacaan ADC. Dalam pengujian ini ADC diberikan input tegangan dengan nilai 0-3 volt yang bersumber dari power supply. Dan hasil pembacaan akan dicocokkan dengan pembacaan multimeter. Dari pengujian yang dilakukan didapat data pembacaan ADC sebagai berikut:

Tabel 4.2 Pembacaan nilai ADC

No	Vin (volt)	Multimeter (volt)	Nilai ADC	Error (volt)	Error (%)
1	0.2	0.25	0.27	0.07	0.35
2	0.4	0.48	0.48	0.08	0.2
3	0.6	0.71	0.71	0.11	0.183
4	0.8	0.88	0.91	0.11	0.1375
5	1	1.06	1.1	0.1	0.1
6	1.2	1.26	1.27	0.07	0.0583
7	1.4	1.47	1.49	0.09	0.0642
8	1.6	1.68	1.65	0.05	0.0312
9	1.8	1.91	1.89	0.09	0.05
10	2	2.09	2.07	0.07	0.035
11	2.2	2.32	2.29	0.09	0.04
12	2.4	2.53	2.5	0.1	0.0416
13	2.6	2.68	2.65	0.05	0.0192
14	2.8	2.89	2.86	0.06	0.0214
15	3	3.15	3.11	0.11	0.0366



Gambar 4.2 Perbandingan Nilai ADC pada data Power Supply dan data Multimeter



Gambar 4.3 Grafik error nilai ADC

Dari pengujian di atas dapat kita lihat selisih dari nilai tegangan antara power supply, multimeter dan perhitungan ADC sangat sedikit, dan dapat dilihat pada gambar 4.3 yaitu grafik error dari pembacaan ADC tidak melebihi dari 0,35%. Ini menandakan bahwa mikrokontroler arduino uno masih sangat baik bila digunakan dan error pembacaan ADC-nya masih dalam batas toleransi.

4.2 Pengujian Software

Pada tahap ini akan dilakukan pengujian terhadap software dari arduino uno dan visual studio membuktikan keakuratan dari *neural network* sebagai metode untuk menghitung dan menentukan sudut azimuth dan sudut elevasi dari posisi drone.

4.2.1 Pengujian Neural Network

Pada pengujian proses *Neural Network* dapat dilihat kelompok beberapa sudut dan jarak uji yang digunakan. suara uji drone yang digunakan sudut azimuth yang mengalami kenaikan sebesar 60° dengan sudut elevasi 0° . Untuk mendapatkan nilai dari inputan data *Neural Network* dilakukan perhitungan untuk mendapatkan nilai dari amplitude tegangan masing-masing sensor.

a. Pengujian sudut azimuth 0°

Tabel 4.3 Nilai tegangan dengan input suara drone dari sudut 0°

Pengujian	mic 1 (Volt)	mic 2 (Volt)	mic 3 (Volt)	mic 4 (Volt)
1	1.865234	1.245117	1.28418	1.186523
2	1.99707	1.25	1.308594	1.035156
3	2.006836	1.279297	1.220703	1.162109
4	1.855469	1.157227	1.279297	1.171875
5	1.99707	1.318359	1.337891	1.152344

b. Pengujian sudut azimuth 60°

Tabel 4.4 Nilai tegangan dengan input suara drone dari sudut 60°

Pengujian	mic 1 (Volt)	mic 2 (Volt)	mic 3 (Volt)	mic 4 (Volt)
1	1.899414	1.665039	1.064453	1.044922
2	1.943359	1.625977	1.054688	1.210938
3	2.021484	1.665039	1.098633	1.098633
4	1.977539	1.635742	1.069336	1.083984
5	1.894531	1.655273	1.362305	0.90332

c. Pengujian sudut azimuth 120°

Tabel 4.5 Nilai tegangan dengan input suara drone dari sudut 120°

Pengujian	mic 1 (Volt)	mic 2 (Volt)	mic 3 (Volt)	mic 4 (Volt)
1	1.289063	1.972656	1.381836	0.585938
2	1.303711	1.870117	1.337891	1.083984
3	1.333008	1.943359	1.386719	1.176758
4	1.367188	1.958008	1.264648	1.142578
5	1.206055	1.894531	1.132813	1.162109

d. Pengujian sudut azimuth 180°

Tabel 4.6 Nilai tegangan dengan input suara drone dari sudut 180°

Pengujian	mic 1 (Volt)	mic 2 (Volt)	mic 3 (Volt)	mic 4 (Volt)
1	0.878906	1.948242	1.147461	1.005859
2	1.010742	1.875	1.577148	1.176758
3	1.220703	1.577148	1.879883	0.864258
4	1.069336	1.210938	1.938477	0.546875
5	1.020508	1.469727	1.865234	0.986328

e. Pengujian sudut azimuth 240⁰

Tabel 4.7 Nilai tegangan dengan input suara drone dari sudut 240⁰

Pengujian	mic 1 (Volt)	mic 2 (Volt)	mic 3 (Volt)	mic 4 (Volt)
1	1.09375	1.176758	1.860352	1.206055
2	0.991211	1.303711	1.860352	1.220703
3	1.235352	1.21582	2.021484	1.201172
4	1.152344	1.313477	1.972656	1.171875
5	1.274414	1.201172	1.938477	1.069336

f. Pengujian sudut azimuth 300⁰

Tabel 4.8 Nilai tegangan dengan input suara drone dari sudut 300⁰

Pengujian	mic 1 (Volt)	mic 2 (Volt)	mic 3 (Volt)	mic 4 (Volt)
1	1.577148	1.152344	1.933594	1.074219
2	1.577148	1.152344	1.933594	1.074219
3	2.011719	0.908203	1.987305	0.78125
4	1.274414	1.201172	1.938477	1.069336
5	1.708984	1.367188	1.855469	1.049805

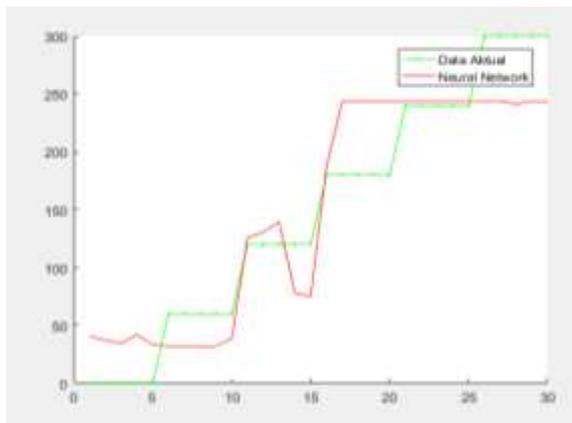
Dari data sudut-sudut nilai tegangan input masing-masing mikrofon tersebut akan diharapkan untuk menjadi nilai target. Nilai target sendiri nanti yang akan menjadi nilai dari perhitungan *neural network*

Dari nilai inputan tersebut akan digunakan sebagai inputan *Neural Network* pada proses pembelajaran. Proses pembelajaran dilakukan yaitu di Komputer Dekstop. Hal ini bertujuan untuk mempercepat proses untuk mendapatkan *weight* dan *bias* yang dibutuhkan untuk klasifikasi. untuk mengetahui nilai dari output yang diinginkan gambar 4.4 dan gambar 4.5 berikut adalah tampilan dari pembelajaran dan error proses pembelajaran *Neural Network*.



Gambar 4.4 Tampilan learning neural network

Dari tampilan learning tersebut dapat dilihat bahwa dengan iterasi sebesar 100.000 performance dari data learning suara drone masih memiliki error yang cukup tinggi yaitu sebesar $1,58 \times 10^3$. Ini menunjukkan mungkin data yang diambil masing kurang banyak untuk dijadikan sebagai data learning neural network. Berikut ini adalah grafik error dari learning data pembacaan suara drone.



Gambar 4.5 Grafik error learning neural network

Pengujian sudut azimuth dengan variabel jarak sumber suara drone dari beberapa sudut elevasi ke antenna mikrofon

4.2.2 Jarak suara drone 10 cm dengan sudut elevasi 0°

Pada pengujian ini akan dilakukan dengan menggunakan input berupa suara yang dihasilkan dari drone yang sesungguhnya dengan intensitas sebesar ± 84 dB. Tabel 4.9 dibawah ini menunjukkan hasil dari pengujian dengan jarak sumber suara 10 cm, sinyal suara dari drone baik dideteksi pada sudut azimuth 0°, 120°, dan 240°, karena sudut tersebut adalah letak dari mikrofon 1, 2, dan 3 terlihat bawah seluruh pengujian pada sudut tersebut berhasil mendeteksi sudut yang benar. Pada sudut azimuth 60°, 180°, dan 300° suara drone lebih besar *error* pendeteksiannya karena sudut tersebut terletak diantara mikrofon sehingga mikrofon agak membedakan arah suara drone. Pada sudut tersebut display interface pada komputer dan LCD errornya menunjukkan sudut yang terletak pada posisi mikrofon. Berikut ini merupakan tabel hasil dari pengujiannya:

Tabel 4.9 Hasil pengujian sudut azimuth dengan jarak suara drone 10 cm dan sudut elevasi 0°

Pengujian	Sudut Azimuth yang terukur					
	0°	60°	120°	180°	240°	300°
1	0°	60°	120°	180°	240°	300°
2	0°	60°	120°	180°	240°	240°
3	0°	120°	120°	240°	240°	0°
4	0°	60°	120°	180°	240°	300°
5	0°	60°	120°	120°	240°	300°
6	0°	0°	120°	180°	240°	300°

4.2.3 Jarak suara drone 20 cm dengan sudut elevasi 0°

Tabel 4.10 Hasil pengujian sudut azimuth dengan jarak suara drone 20 cm dan sudut elevasi 0°

Pengujian	Sudut Azimuth yang terukur					
	0°	60°	120°	180°	240°	300°
1	0°	60°	120°	180°	240°	240°
2	0°	0°	120°	180°	240°	0°
3	0°	120°	120°	180°	240°	300°
4	0°	60°	120°	180°	240°	300°
5	0°	60°	120°	240°	240°	300°
6	0°	60°	120°	180°	240°	300°

Pengujian yang kedua dilakukan dengan jarak suara drone sejauh 20 cm, sedangkan sumber suara yang digunakan masih sama. Pada tabel 4.10 dapat dilihat bahwa hasil pengujian sudut azimuth tidak jauh berbeda dengan pengujian yang sebelumnya karena jarak dari posisi drone selesihnya tidak terlalu jauh hanya 10 cm, dapat disimpulkan bahwa dengan bertambahnya jarak 10 cm tidak terlalu berpengaruh terhadap kinerja dari alat.

4.2.4 Jarak suara drone 10 cm dengan sudut elevasi 30°

Pada pengujian yang ketiga yaitu akan dilakukan dengan jarak sumber suara drone sejauh 10 cm dan sudut elevasi 30°. Tidak jauh berbeda pada pengujian pertama hanya berubah sudut elevasinya meningkat sebesar 30° dan hasil pendeteksian masih bisa disimpulkan baik. Masih terjadi error di beberapa pendeteksian sudut yang terletak diantara sensor mikrofon, hal ini dikarenakan kualitas mikrofon yang berbeda-beda sehingga menyebabkan sinyal yang terbaca memiliki sedikit perbedaan meskipun suara yang dibaca sama. Akan tetapi pembacaan sudut yang dihasilkan tetap berubah-ubah mengikuti gerakan dari suara drone. Berikut ini adalah tabel hasil dari pengujiannya:

Tabel 4.11 Hasil pengujian sudut azimuth dengan jarak suara drone 10 cm dan sudut elevasi 30^0

Pengujian	Sudut Azimuth yang terukur					
	0^0	60^0	120^0	180^0	240^0	300^0
1	0^0	120^0	120^0	180^0	240^0	300^0
2	0^0	60^0	120^0	180^0	240^0	300^0
3	0^0	60^0	120^0	120^0	240^0	300^0
4	0^0	60^0	120^0	240^0	240^0	0^0
5	0^0	60^0	120^0	120^0	240^0	300^0
6	0^0	120^0	120^0	180^0	240^0	240^0

4.2.5 Jarak suara drone 20 cm dengan sudut elevasi 30^0

Pengujian berikutnya dengan jarak 20 cm dan sudut elevasi 30^0 . Pada pengujian ini nilai *error* penentuan sudutnya relatif sama dengan sebelumnya. Pada sudut yang terletak pada sensor yaitu sudut 0^0 , 120^0 , dan 240^0 pendeteksiannya masih baik seluruh pengujiannya masih menunjukkan pendeteksian sudut yang benar sehingga dapat disimpulkan bahwa jarak 20 cm kinerja alat masih baik. Berikut ini adalah tabel hasil dari pengujiannya:

Tabel 4.12 Hasil pengujian sudut azimuth dengan jarak suara drone 20 cm dan sudut elevasi 30^0

Pengujian	Sudut Azimuth yang terukur					
	0^0	60^0	120^0	180^0	240^0	300^0
1	0^0	60^0	120^0	300^0	240^0	300^0
2	0^0	60^0	120^0	180^0	240^0	300^0
3	0^0	120^0	120^0	180^0	240^0	300^0
4	0^0	60^0	120^0	120^0	240^0	300^0
5	0^0	180	120^0	180^0	240^0	0^0
6	0^0	60^0	120^0	120^0	240^0	240^0

4.2.6 Jarak suara drone 10 cm dengan sudut elevasi 60⁰

Pada pengujian yang kelima yaitu akan dilakukan dengan jarak posisi suara drone sejauh 10 cm dan sudut elevasi meningkat menjadi 60⁰. Tidak jauh berbeda pada pengujian pertama dan pengujian ketiga hasil pendeteksian masih bisa disimpulkan baik. Namun pada sudut 300⁰ mengalami peningkatan error sebanyak 3 kali kesalahan dalam pendeteksian posisi dari drone, karena pada sudut tersebut selain terletak diantara mikrofon sudut elevasinya juga naik sehingga agak kesulitan mendeteksinya sehingga beberapa kali pengujian sering menunjukkan sudut 0⁰ dan sudut 240⁰. Berikut ini adalah tabel hasil pengujiannya:

Tabel 4.13 Hasil pengujian sudut azimuth dengan jarak suara drone 10 cm dan sudut elevasi 60⁰

Pengujian	Sudut Azimuth yang terukur					
	0 ⁰	60 ⁰	120 ⁰	180 ⁰	240 ⁰	300 ⁰
1	0 ⁰	60 ⁰	120 ⁰	300 ⁰	240 ⁰	300 ⁰
2	0 ⁰	60 ⁰	120 ⁰	180 ⁰	240 ⁰	300 ⁰
3	0 ⁰	120 ⁰	120 ⁰	180 ⁰	240 ⁰	240 ⁰
4	0 ⁰	60 ⁰	120 ⁰	120 ⁰	240 ⁰	0 ⁰
5	0 ⁰	60 ⁰	120 ⁰	180 ⁰	240 ⁰	300 ⁰
6	0 ⁰	180 ⁰	120 ⁰	240 ⁰	240 ⁰	240 ⁰

4.2.7 Jarak suara drone 20 cm dengan sudut elevasi 60⁰

Pada pengujian yang keenam yaitu pengujian terakhir akan dilakukan dengan jarak posisi suara drone sejauh 20 cm dan sudut elevasi yang sama yaitu 60⁰. Pada Tabel 4.14 di bawah ini dapat kita lihat bahwa tingkat error masih relatif sama dan tidak mengamali perubahan yang signifikan tetapi pada sudut azimuth 60⁰ mengalami peningkatan error yaitu menjadi hanya 3 kali mendeteksi sudut yang benar dari 6 kali pengujian, dan sudut 180⁰ dan 300⁰ sama dengan pengujian kelima hanya 3 kali mendeteksi posisi drone yang sesuai. Di bawah ini adalah tabel hasil pengujiannya dengan 6 kali pengujian:

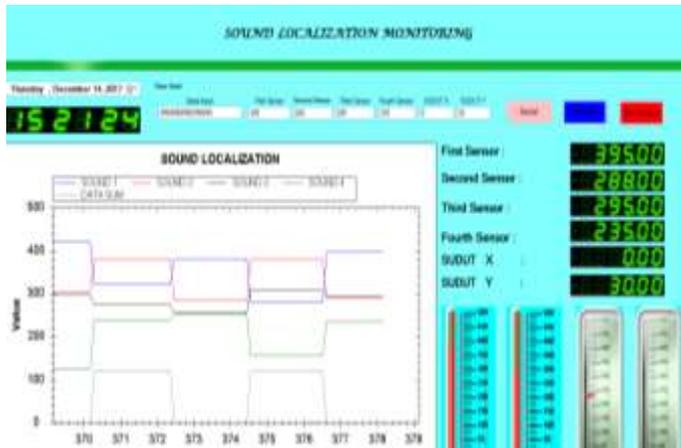
Tabel 4.14 Hasil pengujian sudut azimuth dengan jarak suara drone 20 cm dan sudut elevasi 60⁰

Pengujian	Sudut Azimuth yang terukur					
	0 ⁰	60 ⁰	120 ⁰	180 ⁰	240 ⁰	300 ⁰
1	0 ⁰	0 ⁰	120 ⁰	120 ⁰	240 ⁰	240 ⁰
2	0 ⁰	60 ⁰	120 ⁰	180 ⁰	240 ⁰	300 ⁰
3	0 ⁰	60 ⁰	120 ⁰	180 ⁰	240 ⁰	240 ⁰
4	0 ⁰	0 ⁰	120 ⁰	240 ⁰	240 ⁰	300 ⁰
5	0 ⁰	60 ⁰	120 ⁰	180 ⁰	240 ⁰	240 ⁰
6	0 ⁰	120 ⁰	120 ⁰	120 ⁰	240 ⁰	300 ⁰

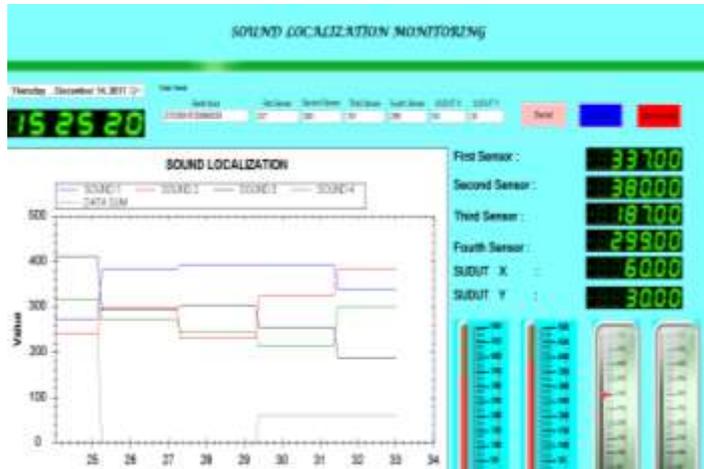
Perhitungan nilai error & Tampilan pengujian di komputer

- a. Sudut 0^0
Dari 36x pengujian dapat dilihat bahwa seluruhnya pengujian sudut azimuth 0^0 dinyatakan berhasil, sehingga error dari alat ini adalah 0%.
- b. Sudut 60^0
Dari 36x pengujian terdapat 21x berhasil dan 15x gagal dalam pendeteksian sudut azimuth 60^0 , sehingga dapat dihitung nilai error alat ini adalah:
 $13/36 \times 100\% = 36,1\%$

Gambar 4.6 dan gambar 4.7 berikut ini adalah tampilan interface pada komputer yang menunjukkan pendeteksian posisi drone dengan sudut azimuth 0^0 dan sudut 60^0 :



Gambar 4.6 Tampilan pengujian sudut azimuth 0^0



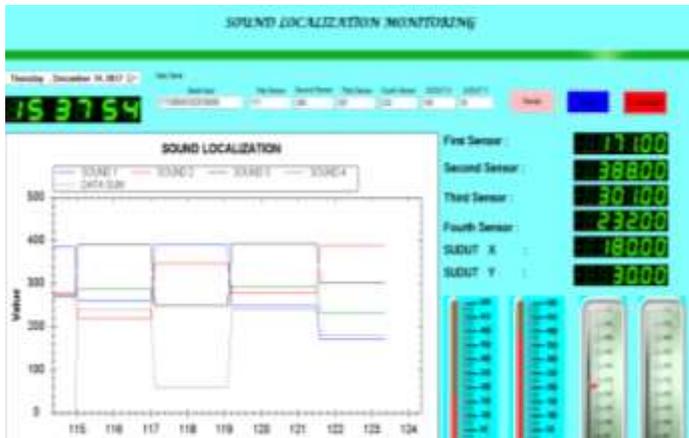
Gambar 4.7 Tampilan pengujian sudut azimuth 60°

- c. Sudut 120°
 Dari 36x pengujian sudut azimuth 120° seluruhnya berhasil menunjukkan sudut yang semestinya sehingga error dari alat ini adalah 0%.
- d. Sudut 180°
 Dari 36x pengujian terdapat 20x berhasil dan 15x gagal dalam pendeteksian sudut azimuth 180° , sehingga dapat dihitung nilai error alat ini adalah:
 $15/36 \times 100\% = 41,6\%$

Gambar 4.8 dan 4.9 berikut ini adalah tampilan interface pada komputer yang menunjukkan pendeteksian posisi drone dengan sudut azimuth 120° dan sudut 180° :



Gambar 4.8 Tampilan pengujian sudut azimuth 120°



Gambar 4.9 Tampilan pengujian sudut azimuth 180°

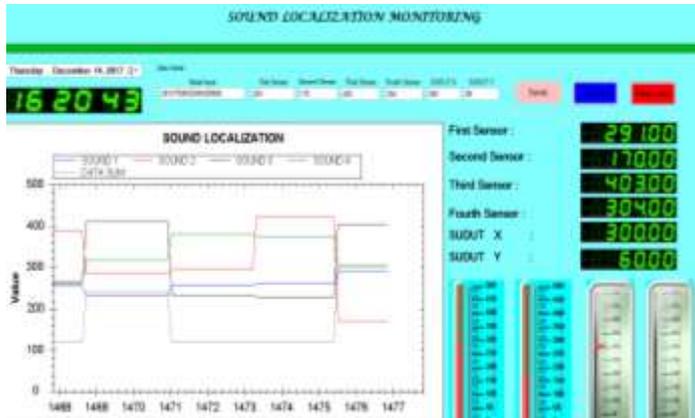
- e. Sudut 240⁰
 Dari 36x pengujian terhadap sudut azimuth 240⁰ seluruh pengujiannya berhasil menunjukkan sudut yang benar sehingga error pada alat ini yaitu 0%.

- f. Sudut 300⁰
 Dari 36x pengujian terdapat 22x berhasil dan 14x gagal dalam pendeteksian sudut azimuth 300⁰, sehingga dapat dihitung nilai error alat ini adalah:
 $14/36 \times 100\% = 38,8\%$

Gambar 4.8 dan 4.9 berikut ini adalah tampilan interface pada komputer yang menunjukkan pendeteksian posisi drone dengan sudut azimuth 120⁰ dan sudut 180⁰:



Gambar 4.10 Tampilan pengujian sudut azimuth 240⁰



Gambar 4.11 Output display pengujian sudut 300^o

Dari nilai error yang didapat maka dapat dihitung rata-rata error dari alat ini yaitu:

$$(36,1 \% + 41,6 \% + 38,8 \%) : 6 = 19,41 \%$$

BAB 5 PENUTUP

5.1 Kesimpulan

Pada penelitian ini telah dirancang dan dibuat suatu alat pendeteksi posisi suara drone berdasarkan sinyal suara. Sistem ini terdiri dari 4 buah mikrofon sebagai sensor suara. Tegangan keluaran dari sensor suara dilewatkan pada rangkaian filter dan dikonversi oleh analog to digital converter pada mikrokontroler Arduino Uno. Metode Decision tree digunakan untuk menghasilkan sudut azimuth dan elevasi dari posisi drone yang kemudian ditampilkan pada LCD atau komputer. Hasil pengujian menunjukkan bahwa sistem ini mempunyai kemampuan untuk mendeteksi posisi suara drone dengan kenaikan sudut azimuth sebesar 60^0 dan sudut elevasi sebesar 30^0 dengan jarak sumber suara di bawah 100cm dan intensitas suara sebesar 84 dB. Pada pengujian posisi suara drone dengan berbagai sudut azimuth dan elevasi menghasilkan rerata kesalahan sebesar 19,41%. Sistem ini diharapkan dapat digunakan sebagai pengarah pancaran gelombang radio pada peralatan drone jammer sehingga dapat meningkatkan efisiensi daya listrik.

5.2 Saran

Penambahan jumlah mikrofon serta tipe mikrofon yang lebih sensitif diperlukan untuk dapat memprediksi posisi suara drone dengan tingkat keakuratan yang lebih tinggi. Pemakaian mikrokontroler yang mempunyai prosesornya dengan kecepatan tinggi diperlukan agar pengolahan data sinyal dapat dilakukan secara terintegrasi didalam modul tersebut sehingga tidak diperlukan *personal computer*. Implementasi pengarah suara secara mekanik dan jarak antar mikrofon dapat dilakukan untuk meningkatkan keakuratan sistem.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1] Zakaria, Neza, Sasmini. *“Legalitas Penggunaan Drone yang Melintasi Batas Negara Berdasarkan Hukum Internasional”*. 2015
- [2] Anta Pradana, M.G, Prasakti, Ridho. Worsito, S.B, Fajaryati, Nuryake. *“Inovasi Rancang Bangun Drone Single Propeller sebagai wahana pemetaan lahan berbasis Unmanned Aerial Vehicle (UAV)”*. Universitas Negeri Yogyakarta, 2016.
- [3] Asfari, Mohamad. *“Penentuan Arah Sumber Suara Dengan Metode Interaural Time Difference Menggunakan Mikrokontroler Stm32f4”*, Institut Teknologi Sepuluh Nopember, Surabaya, 2017..
- [4] Malhotra, A., Zhi Chua., Haroush, G.,Leung, Chi., Olexa, Pavol., Wilson, A., Zhao, Y. *“Detection of Civil Unmanned Aerial Vehicles by Sound Processing”*. Imperial Collage, London. 2016.
- [5] Fausi, Moh., Naba, Agus., Santjojo, Djoko. *“Analisis Jarak Microphone Array dengan Teknik Pemrosesan Sinyal Fast Fourier Transform Beamforming”*. 2015
- [6] Prawira, W.P, Muningsgar, Jodelin., Suci Santi, Made Rai. *“Alat Perekam Aktivitas Jantung dengan Mic Kondensor dan PC-Link USB Smart I/O”*. Universitas Kristen Satya Wacana. 2015.
- [7] <https://www.adafruit.com/product/1713>,access 20 Desember 2017.
- [8] Albet, Muhammad., Ginta, P.W., Sudarsono, Aji. *“Pembuatan Jendela Otomatis menggunakan Sensor Cahaya”*. Universitas Dehasen, Bengkulu. 2014.
- [9] Maemunnur, Abdul Fatah. *“Rancang Bangun Sistem Alat Ukur Turbidity Untuk Analisis Kualitas Air Berbasis Arduino Uno”*. Universitas Pendidikan Indonesia, Bandung, 2016.
- [10] Fitriandi, A., Komalasari, E, Gusmedi, Herri. *“Rancang Bangun Alat Monitoring Arus dan Tegangan Berbasis Mikrokontroler dengan SMS Gateway”*. Universitas Lampung, 2016.
- [11] Muhammad Rivai, Tasripan, Sumandari, *“Pengenalan Pola Sinyal Suara Kerusakan Motor Listrik menggunakan Neural Network”*, Institut Teknologi Sepuluh Nopember, Surabaya, 2007.
- [12] Gumelar, Mido.*“Rancang Bangun Wireless Electronic Nose Berbasis Teknologi Internet of Things”*. Institut Teknologi Sepuluh Nopember, Surabaya, 2017.

- [13] Muhammad Rivai, Suwito, Budiman, Fajar, Nursyca, M.A. “*Pengenalan Suara Burung Menggunakan Mel Frequency Cepstrum Coefficient dan Jaringan Syaraf Tiruan pada sistem Pengusir Hama Burung*”. Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- [14] Rais Ruli, Ahmad. “*Implementasi Aplikasi Pendaftaran dan Pembayaran Kontrakkan Berbasis Desktop VB Net dan Microsoft Access*”. Amik BSI Tangerang, 2017.

LAMPIRAN

1. Program Arduino Uno

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x3F, 16, 2);

int sudut,sudut_y;
float bagi;
String inputString = "";    // Variabel untuk mendeteksi adanya
inputan serial dari vb.
boolean stringComplete = false; //variabel untuk menerima
perintah dari VB
boolean req =false;        //variabel untuk menerima perintah dari
VB DENGAN KONDISI PERINTAH

void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  lcd.begin();
  lcd.backlight();
  inputString.reserve(200);
  lcd.setCursor(6,0);
  lcd.print("SOUND");
  lcd.setCursor(3,1);
  lcd.print("LOCALIZATION");
  delay(500);
  lcd.print(".");
  delay(500);
  lcd.print(".");
  delay(500);
  lcd.print(".");
  delay(500);
  lcd.print(".");
}
```

```

delay(500);
lcd.clear();
lcd.setCursor(6,0);
lcd.print("SOUND");
lcd.setCursor(4,1);
lcd.print("Ready...!");
delay(2000);
lcd.clear();
lcd.setCursor(6,0);
lcd.print("SOUND");
lcd.setCursor(4,1);
lcd.print("SUDUT = ...?");
}

```

// the loop routine runs over and over again forever:

```

void loop() {

  // read the input on analog pin 0:
  if (stringComplete) { //KONDISI MENGETAHUI
ADANYA SERIAL YANG MASUK
    //Serial.println(inputString);
    if (inputString[0]=='y'){ //VARIABEL SELEKSI SERIAL
CONNECT
      req= true; //inputan untuk perintah mengirimkan
hasil baca dari vb terpenuhi
    }else if(inputString[0]=='n'){ //VARIABEL SELEKSI
SERIAL DISCONNECT
      req=false; //inputan untuk perintah menghentikan
pengiriman hasil baca sensor dari vb terpenuhi
    }
    // clear the string:
    inputString = "";
    stringComplete = false;
  }
}

```

```

int sensorValue0 = analogRead(A0);
int sensorValue1 = analogRead(A1);
int sensorValue2 = analogRead(A2);
int sensorValue3 = analogRead(A3);

```

```

if(sensorValue0 >= 380 || sensorValue1 >= 380 || sensorValue2 >=
380 || sensorValue3 >= 380){
    lcd.clear();
    bagi = ((sensorValue0 + sensorValue1 + sensorValue2)/3);

    if(sensorValue0 > bagi && sensorValue1 > bagi &&
sensorValue2 < bagi){
        sudut = 60;
    }else if(sensorValue0 < bagi && sensorValue1 > bagi &&
sensorValue2 > bagi){
        sudut = 180;
    }else if(sensorValue0 > bagi && sensorValue1 < bagi &&
sensorValue2 > bagi){
        sudut = 300;
    }else{
        if(sensorValue0 > sensorValue1 && sensorValue0 >
sensorValue2){
            sudut = 0;
        }else if(sensorValue1 > sensorValue0 && sensorValue1 >
sensorValue2){
            sudut = 120;
        }else if(sensorValue2 > sensorValue0 && sensorValue2 >
sensorValue1){
            sudut = 240;
        }
    }
}

if(sensorValue3 > 400){
    sudut_y = 90;
}else if(sensorValue3 > 300 && sensorValue3 < 400){
    sudut_y = 60;
}else if(sensorValue3 < 300){
    sudut_y = 30;
}

if (req){ //KONDISI JIKA TERPENUHI ADANYA "Y"
mengirim data dari sensor ke vb.
    Serial.print(sensorValue0);

```

```

Serial.print("|");
Serial.print(sensorValue1);
Serial.print("|");
Serial.print(sensorValue2);
Serial.print("|");
Serial.print(sensorValue3);
Serial.print("|");
Serial.print(sudut);
Serial.print("|");
Serial.print(sudut_y);
Serial.println("|");

lcd.setCursor(1,0);
lcd.print("S_LOCALIZATION");
lcd.setCursor(0,1);
lcd.print("< X= ");
lcd.print(sudut);
lcd.print(", Y= ");
lcd.print(sudut_y);
lcd.print("");
delay(2000);
}
}
// print out the value you read:

delay(1);
}

void serialEvent() {
  while (Serial.available()) { //menunggu adanya masukan
serial dari vb.
    // get the new byte:
    char inChar = (char)Serial.read(); //membaca adanya masukan
serial dari vb.
    // add it to the inputString:
    if (inChar == '\n') {
      stringComplete = true;
    }else{

```

```
inputString += inChar;  
// if the incoming character is a newline, set a flag  
// so the main loop can do something about it:  
}  
}  
}
```

Halaman ini sengaja dikosongkan

BIODATA PENULIS



Fardian Abdi Putra. Dilahirkan di Tanjungpandan, pada tanggal 23 Agustus 1990 merupakan putra kedua dari dari tujuh bersaudara pasangan Bapak Usman Hasan dan Ibu Ramayani. Penulis menamatkan sekolah di SDN 45 Tanjungpandan tahun 2002. Kemudian masuk ke SMPN 1 Tanjungpandan, tamat tahun 2005, dan melanjutkan di SMAN 1 Tanjungpandan dan tamat pada tahun 2008. Pada tahun 2008, penulis melanjutkan pendidikan di AAU (Akademi Angkatan Udara) Yogyakarta dan tamat pada tahun 2012 dengan gelar D4. Selanjutnya penulis mendapatkan kesempatan dari TNI AU untuk melaksanakan pendidikan S1 program Lintas Jalur di Jurusan Teknik Elektro, FTI – ITS Surabaya pada pertengahan tahun 2015. Penulis memilih bidang studi Elektronika dan mengambil topik penelitian di Laboratorium Elektronika Industri.

E-mail : fardianabdi23@gmail.com

Halaman Ini Sengaja Dikosongkan