

repository.ub.ac.id

**RANCANG BANGUN *DIGITAL BANDPASS FILTER* SEBAGAI
APLIKASI *SOUND ACTIVATION* PADA KONTES ROBOT
PEMADAM API INDONESIA**

SKRIPSI

KONSENTRASI TEKNIK TELEKOMUNIKASI

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh:

GUNTUR EKA PUTRO
NIM. 115060300111015

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2015



LEMBAR PERSETUJUAN

**RANCANG BANGUN *DIGITAL BANDPASS FILTER* SEBAGAI
APLIKASI *SOUND ACTIVATION* PADA KONTES ROBOT
PEMADAM API INDONESIA**

SKRIPSI

KONSENTRASI TEKNIK TELEKOMUNIKASI

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh:

GUNTUR EKA PUTRO
NIM. 115060300111015

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Ir. Wahyu Adi Privono, M.T.
NIP. 19600518 198802 1 001

Ali Mustofa, S.T., M.T.
NIP. 19710601 200003 1 001

LEMBAR PENGESAHAN

**RANCANG BANGUN *DIGITAL BANDPASS FILTER* SEBAGAI
APLIKASI *SOUND ACTIVATION* PADA KONTES ROBOT
PEMADAM API INDONESIA**

SKRIPSI

KONSENTRASI TEKNIK TELEKOMUNIKASI

Disusun oleh:

GUNTUR EKA PUTRO
NIM. 115060300111015

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 6 Oktober 2015

MAJELIS PENGUJI

Dr. Ir. Sholeh Hadi Pramono, M.S.
NIP. 19580728 198701 1 001

Rudy Yuwono, S.T., M.Sc.
NIP. 19710615 199802 1 003

Goegoes Dwi Nusantoro, S.T.,M.T.
NIP. 19711013 200604 1 001

Mengetahui,
Ketua Jurusan Teknik Elektro

M. Aziz Muslim, S.T., M.T., Ph.D.
NIP. 19741203 200012 1 001

PENGANTAR

Alhamdulillah, segala puji hanya bagi Allah Subhanahu Wa Taala. Dialah Allah, Tuhan Yang Maha Mulia, Yang Maha Pengasih lagi Maha Penyayang. Dialah Sebaik baik Penolong dan Sebaik baik Pelindung. Shalawat dan salam kepada Nabi Muhammand Rasulullah Shallallahu Alaihi Wa Salam, Sang pembawa kabar gembira dan sebaik suri tauladan bagi yang mengharap Rahmat dan Hidayah-Nya

Sungguh hanya melalui Pertolongan dan Perlindungan Allah SWT semata sehingga skripsi ini dapat diselesaikan. Dengan seizin Allah SWT, dikesempatan yang baik ini saya menghaturkan rasa terima kasih dan penghargaan yang sebesar besarnya atas bantuan sehingga terselesainya skripsi ini kepada:

- Ayah dan Ibu atas segala kesabaran, perhatian, dan kasih sayang dalam membesarkan dan mendidik penulis, serta mendoakan kemudahan bagi penulis hingga terselesainya skripsi ini.
- Bapak M. Aziz Muslim, ST., MT., Ph.D selaku Ketua Jurusan Teknik Elektro Universitas Brawijaya, Bapak Hadi Suyono, ST., MT., Ph.D selaku Sekretaris Jurusan Teknik Elektro Universitas Brawijaya, Bapak Ali Mustofa, ST., MT. selaku Ketua Prodi Strata Satu Jurusan Teknik Elektro Universitas Brawijaya, dan Ibu Rusmi Ambarwati, S.T.,M.T. selaku Ketua Kelompok Dosen Keahlian Telekomunikasi Jurusan Teknik Elektro Universitas Brawijaya.
- Bapak Dwi Fadillah K, ST., MT. selaku dosen Penasehat Akademik,
- Bapak Ir. Wahyu Adi Priyono, M.T. sebagai Dosen Pembimbing I, dan Bapak Ali Mustofa, S.T., M.T. sebagai Dosen Pembimbing II atas segala bimbingan, pengarahan, bantuan saran, dan kritik yang telah diberikan.
- Seluruh dosen pengajar dan staff Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
- Rani Maulida atas dukungan, perhatian, pengertian, dan kesabaran yang telah diberikan.
- Penghuni kontrakan Sigura-gura V No. 25 H : Keynan, Feby ‘yanto’, Basith, dan Wichak atas canda tawa dan persahabatan yang terjalin selama 3 tahun.
- Teman-teman Brawijaya Smansasi 2011 atas kebersamaan dan perjuangan dalam perantauan

- Rekan-rekan Tim Robot UB 2011 : Liza, Amanda, Desta, Mirza, Ainun, Irham ‘darjo’, Jaka, Bustanul, Bayu ‘emon’, Tegar ‘bowo’, Yanto, atas perjuangan dan kontribusi selama 3 tahun.
- Rekan-rekan Subdivisi KRPAI : Amanda, Desta, Yanto, Ricky, Reza, Sofyan, Wiwin, Hesti, Doni, Ryan, Oky, Alec, Eky, Hanif, atas kerja keras dan kebersamaan dalam tim.
- Seluruh Keluarga Besar Tim Robot Universitas Brawijaya.
- Seluruh Keluarga Besar Laboratorium Sistem Digital.
- Teman-teman seperjuangan 42 : Rozi, Azri, Samid, Mirza, Kacong, Bayu ‘emon’ yang saling berbagi suka duka dalam perjuangan mencapai titik akhir.
- Teman-teman Inverter ’11 Teknik Elektro Universitas Brawijaya.
- Seluruh pihak yang telah memberikan bantuan serta dukungan baik secara langsung maupun tidak langsung atas penyusunan skripsi ini.

Sekiranya Allah SWT mencatat amalan ikhlas kami dan semua pihak yang turut membantu sehingga skripsi ini terselesaikan. Akhirnya, penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun, serta semoga skripsi ini dapat bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi untuk masyarakat.

Malang, Oktober 2015

Penulis

RINGKASAN

GUNTUR EKA PUTRO, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, September 2015, *Rancang Bangun Digital Bandpass Filter sebagai Aplikasi Sound Activation pada Kontes Robot Pemadam Api Indonesia*, Dosen Pembimbing: Ir. Wahyu Adi Priyono, M.T., dan Ali Mustofa, S.T.,M.T.

Kontes Robot Pemadam Api Indonesia (KRPAI) adalah salah satu kategori yang dilombakan dalam Kontes Robot Indonesia (KRI). Sistem robot KRPAI merupakan sistem robot cerdas, dimana robot diberikan kecerdasan buatan untuk melaksanakan tugasnya, yaitu menemukan api didalam sebuah labirin, kemudian memadamkannya. Untuk mengaktifkan robot dibutuhkan sinyal suara yang memiliki frekuensi 3,8 kHz. Nilai frekuensi suara ini sudah ditentukan oleh panitia dan tercantum pada *rule* perlombaan. Sinyal suara ini kemudian disebut sebagai suara aktivasi.

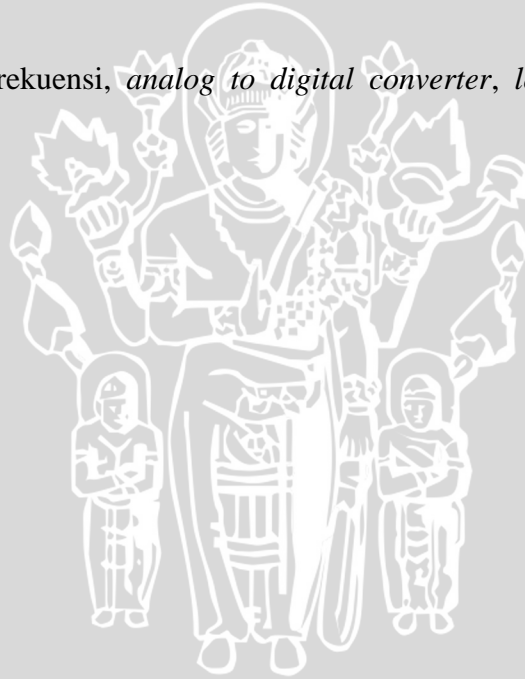
Sound activation adalah sistem pengolahan sinyal pada robot KRPAI yang berfungsi untuk mengenali suara aktivasi. Dalam perlombaan yang sesungguhnya, terdapat sumber audio lain disekitar robot yang mampu mengganggu sistem, antara lain musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara. Oleh karena itu, sistem ini membutuhkan metode pengolahan sinyal yang tahan terhadap *noise* agar tidak mudah terganggu oleh sumber audio lainnya. Untuk itu digunakan pengolahan sinyal secara digital karena kelebihanannya yang lebih tahan terhadap *noise* dibandingkan sistem pengolahan sinyal secara analog.

Sumber audio berupa suara aktivasi diberikan kepada mikrofon yang berfungsi untuk mengubah sinyal suara menjadi sinyal elektrik. Kemudian sinyal elektrik menjadi masukan *lowpass filter* dengan frekuensi *cutoff* 5 kHz yang berfungsi meredam sinyal dengan frekuensi diatas 5 kHz. Sinyal elektrik keluaran *lowpass filter* akan melalui proses *analog to digital converter* (ADC) yang terdiri dari proses *sampling* dengan frekuensi *sampling* 10 kHz, proses kuantisasi dengan 8 bit kuantisasi, dan proses *coding* untuk menghasilkan sinyal diskrit yang kemudian akan dilakukan proses komputasi menggunakan algoritma *digital bandpass filter*. Filter digital ini akan meloloskan sinyal dengan frekuensi 3,8 kHz, dan meredam sinyal dengan frekuensi dibawah frekuensi *cutoff* bawah yaitu 3,7 kHz dan diatas frekuensi *cutoff* atas yaitu 3,9 kHz.

Berdasarkan hasil pengujian, *lowpass filter* mampu meredam sinyal dengan frekuensi diatas 5 kHz, dengan nilai $|H(j\omega)|$ sebesar 0,72 (-2,85 dB) dan *error* $|H(j\omega)|$ pada frekuensi *cutoff* sebesar 1,805 %, sedangkan *digital bandpass filter* mampu meredam sinyal dibawah frekuensi *cutoff* bawah dengan nilai $|H(\omega)|$ sebesar 0,733 (-2,69 dB) dan nilai *error* $|H(\omega)|$ sebesar 3,67 % dan diatas frekuensi *cutoff* atas dengan nilai $|H(j\omega)|$ sebesar 0,73 (-2,72 dB) dan nilai *error* $|H(\omega)|$ sebesar 3,15 %.

Sistem *sound activation* diuji dengan diberikan sumber audio berupa suara aktivasi, musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara. Ketika diberikan suara aktivasi, indikator LED hijau menyala dan LED merah padam, sedangkan ketika diberikan musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara, indikator LED hijau padam dan LED merah menyala. Hal ini menunjukkan sistem mampu mendeteksi suara aktivasi dengan baik dan mengabaikan sumber audio lainnya.

Kata Kunci: KRPAI, frekuensi, *analog to digital converter*, *lowpass filter*, *digital bandpass filter*



DAFTAR ISI

| | Halaman |
|--|-------------|
| PENGANTAR..... | i |
| RINGKASAN..... | iii |
| DAFTAR ISI..... | v |
| DAFTAR GAMBAR..... | viii |
| DAFTAR TABEL..... | xi |
| 1. BAB I PENDAHULUAN..... | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 2 |
| 1.3 Batasan Masalah..... | 2 |
| 1.4 Tujuan..... | 2 |
| 1.5 Kontribusi Penelitian..... | 3 |
| 1.6 Sistematika Penulisan..... | 3 |
| 2 BAB II TINJAUAN PUSTAKA..... | 4 |
| 2.1 Umum..... | 4 |
| 2.2 Sinyal..... | 4 |
| 2.3 Mikrofon..... | 5 |
| 2.4 Analog to Digital Converter (ADC)..... | 7 |
| 2.5 Filter Analog | 11 |
| 2.6 Filter Digital | 16 |
| 2.7 Mikrokontroler Atmel AVR ATmega8 | 19 |
| 3 BAB III METODE PENELITIAN..... | 26 |
| 3.1 Perancangan Sistem..... | 26 |
| 3.1.1 Perancangan Mikrofon | 26 |
| 3.1.2 Perancangan <i>Lowpass Filter</i> | 27 |
| 3.1.3 Perancangan Mikrokontroler | 28 |
| 3.1.4 Perancangan Proses <i>Sampling</i> | 29 |
| 3.1.5 Perancangan Proses Kuantisasi | 29 |
| 3.1.6 Perancangan Proses <i>Coding</i> | 30 |
| 3.1.7 Perancangan <i>Digital Bandpass Filter</i> | 30 |
| 3.2 Pengujian Sistem | 31 |



| | | |
|----------|--|-----------|
| 3.2.1 | Pengujian Mikrofon | 32 |
| 3.2.2 | Pengujian <i>Lowpass Filter</i> | 32 |
| 3.2.3 | Pengujian Proses <i>Sampling</i> | 34 |
| 3.2.4 | Pengujian Proses Kuantisasi | 34 |
| 3.2.5 | Pengujian Proses <i>Coding</i> | 36 |
| 3.2.6 | Pengujian <i>Digital Bandpass Filter</i> | 37 |
| 3.2.7 | Pengujian Keseluruhan Sistem | 39 |
| 3.3 | Pembahasan | 40 |
| 3.4 | Pengambilan Kesimpulan dan Saran | 40 |
| 4 | BAB IV HASIL DAN PEMBAHASAN..... | 41 |
| 4.1 | Umum..... | 41 |
| 4.2 | Perancangan Sistem..... | 41 |
| 4.2.1 | Perancangan Mikrofon | 42 |
| 4.2.2 | Perancangan <i>Lowpass Filter</i> | 43 |
| 4.2.3 | Perancangan Mikrokontroler ATmega8..... | 44 |
| 4.2.4 | Perancangan Proses <i>Sampling</i> | 45 |
| 4.2.5 | Perancangan Proses Kuantisasi | 45 |
| 4.2.6 | Perancangan Proses <i>Coding</i> | 46 |
| 4.2.7 | Perancangan <i>Digital Bandpass Filter</i> | 47 |
| 4.3 | Impelentasi Perancangan..... | 52 |
| 4.4 | Pengujian Sistem | 53 |
| 4.4.1 | Pengujian Mikrofon | 53 |
| 4.4.2 | Pengujian <i>Lowpass Filter</i> | 56 |
| 4.4.3 | Pengujian Proses <i>Sampling</i> | 59 |
| 4.4.4 | Pengujian Proses Kuantisasi | 61 |
| 4.4.5 | Pengujian Proses <i>Coding</i> | 64 |
| 4.4.6 | Pengujian <i>Digital Bandpass Filter</i> | 66 |
| 4.4.7 | Pengujian Keseluruhan Sistem | 70 |
| 5 | BAB V PENUTUP..... | 73 |
| 5.1 | Kesimpulan..... | 73 |
| 5.2 | Saran..... | 73 |

DAFTAR PUSTAKA.....74
LAMPIRAN..... 75



DAFTAR GAMBAR

| No. | Judul | Halaman |
|-------------|---|---------|
| Gambar 2.1 | Sinyal Suara Manusia..... | 5 |
| Gambar 2.2 | Mikrofon Elektret..... | 6 |
| Gambar 2.3 | Rangkaian Elektronik Mikrofon dan Penguat Op-amp..... | 7 |
| Gambar 2.4 | Proses <i>Analog to Digital Converter</i> (ADC)..... | 8 |
| Gambar 2.5 | Sinyal Hasil <i>Sampling</i> | 8 |
| Gambar 2.6 | Fenomena <i>Aliasing</i> | 9 |
| Gambar 2.7 | Sinyal Hasil Kuantisasi..... | 10 |
| Gambar 2.8 | Sinyal Hasil <i>Coding</i> | 11 |
| Gambar 2.9 | Karakteristik Frekuensi LPF Ideal..... | 11 |
| Gambar 2.10 | Karakteristik Frekuensi BPF Ideal..... | 11 |
| Gambar 2.11 | Respon Frekuensi Filter <i>Butterworth</i> , <i>Chebyshev</i> , dan <i>Elliptic</i> | 12 |
| Gambar 2.12 | Rangkaian <i>Lowpass Filter</i> | 13 |
| Gambar 2.13 | Respon Frekuensi <i>Lowpass Filter</i> | 14 |
| Gambar 2.14 | Spesifikasi <i>Lowpass Filter Butterworth</i> | 15 |
| Gambar 2.15 | Respon Frekuensi <i>Lowpass Filter</i> dengan Orde Berbeda..... | 16 |
| Gambar 2.16 | Diagram Blok Filter IIR..... | 17 |
| Gambar 2.17 | Diagram Blok Filter FIR..... | 17 |
| Gambar 2.18 | Prosedur untuk Desain Filter Digital..... | 18 |
| Gambar 2.19 | Konfigurasi Pin ATmega8 Model PDIP..... | 20 |
| Gambar 2.20 | Diagram Blok ADC ATmega8..... | 22 |
| Gambar 2.21 | Diagram Blok <i>Timer2</i> ATmega8..... | 23 |
| Gambar 2.22 | Jendela <i>Code Wizard AVR</i> | 24 |
| Gambar 2.23 | Pemilihan Jenis <i>Chip</i> | 24 |
| Gambar 2.24 | Konfigurasi Port C Sebagai <i>Input</i> | 25 |
| Gambar 2.25 | Mengaktifkan ADC dari Mikrokontroler..... | 25 |
| Gambar 3.1 | Diagram Blok Sistem yang Dirancang..... | 26 |
| Gambar 3.2 | Diagram Alir Perancangan Mikrofon..... | 27 |
| Gambar 3.3 | Diagram Alir Perancangan <i>Lowpass Filter</i> | 28 |
| Gambar 3.4 | Diagram Alir Perancangan Mikrokontroler..... | 28 |
| Gambar 3.5 | Diagram Alir Perancangan Proses <i>Sampling</i> | 29 |
| Gambar 3.6 | Diagram Alir Perancangan Proses Kuantisasi..... | 29 |

| | |
|--|----|
| Gambar 3.7 Diagram Alir Perancangan Proses <i>Coding</i> | 30 |
| Gambar 3.8 Diagram Alir Perancangan <i>Digital Bandpass Filter</i> | 31 |
| Gambar 3.9 Diagram Alir Pengujian Mikrofon..... | 32 |
| Gambar 3.10 Diagram Alir Pengujian <i>Lowpass Filter</i> | 33 |
| Gambar 3.11 Diagram Alir Pengujian Proses <i>Sampling</i> | 34 |
| Gambar 3.12 Diagram Alir Pengujian Proses Kuantisasi..... | 35 |
| Gambar 3.13 Diagram Alir Pengujian Proses <i>Coding</i> | 36 |
| Gambar 3.14 Diagram Alir Pengujian <i>Digital Bandpass Filter</i> | 38 |
| Gambar 3.15 Diagram Alir Pengujian Keseluruhan Sistem..... | 39 |
| Gambar 4.1 Diagram Blok Sistem yang Dirancang..... | 41 |
| Gambar 4.2 Rangkaian Elektronik dari Mikrofon..... | 42 |
| Gambar 4.3 Rangkaian <i>Lowpass Filter</i> | 43 |
| Gambar 4.4 Rangkaian Elektronik Mikrokontroler ATmega8..... | 44 |
| Gambar 4.5 Diagram Alir Proses <i>Sampling</i> | 45 |
| Gambar 4.6 Diagram Alir Proses Kuantisasi..... | 46 |
| Gambar 4.7 Diagram Alir Proses <i>Coding</i> | 46 |
| Gambar 4.8 Spesifikasi <i>Digital Bandpass Filter</i> | 48 |
| Gambar 4.9 Struktur Persamaan Beda..... | 51 |
| Gambar 4.10 Algoritma <i>Digital Bandpass Filter</i> | 51 |
| Gambar 4.11 Diagram Alir Pengujian Sistem..... | 53 |
| Gambar 4.12 Diagram Blok Pengujian Mikrofon..... | 54 |
| Gambar 4.13 Sinyal Keluaran Mikrofon Tanpa Sumber Audio..... | 55 |
| Gambar 4.14 Sinyal Keluaran Mikrofon dengan Sumber Audio 3,8 kHz..... | 55 |
| Gambar 4.15 Diagram Blok Pengujian <i>Lowpass Filter</i> | 56 |
| Gambar 4.16 Grafik Respon Frekuensi <i>Lowpass Filter</i> | 58 |
| Gambar 4.17 Diagram Blok Pengujian Proses <i>Sampling</i> | 59 |
| Gambar 4.18 Hasil Sinyal Proses <i>Sampling</i> | 60 |
| Gambar 4.19 Perbandingan Sinyal Uji dengan Sinyal <i>Sampling</i> | 60 |
| Gambar 4.20 Diagram Blok Pengujian Proses Kuantisasi..... | 61 |
| Gambar 4.21 Hasil Pengujian Proses Kuantisasi..... | 62 |
| Gambar 4.22 Diagram Blok Pengujian Proses <i>Coding</i> | 64 |
| Gambar 4.23 Hasil Proses Kuantisasi pada Sinyal Suara dengan Frekuensi 3,8 kHz..... | 65 |
| Gambar 4.24 Hasil Proses <i>Coding</i> pada Sinyal Suara dengan Frekuensi 3,8 kHz..... | 66 |
| Gambar 4.25 Diagram Blok Pengujian <i>Digital Bandpass Filter</i> | 67 |

Gambar 4.26 Grafik Respon Frekuensi *Digital Bandpass Filter*.....69

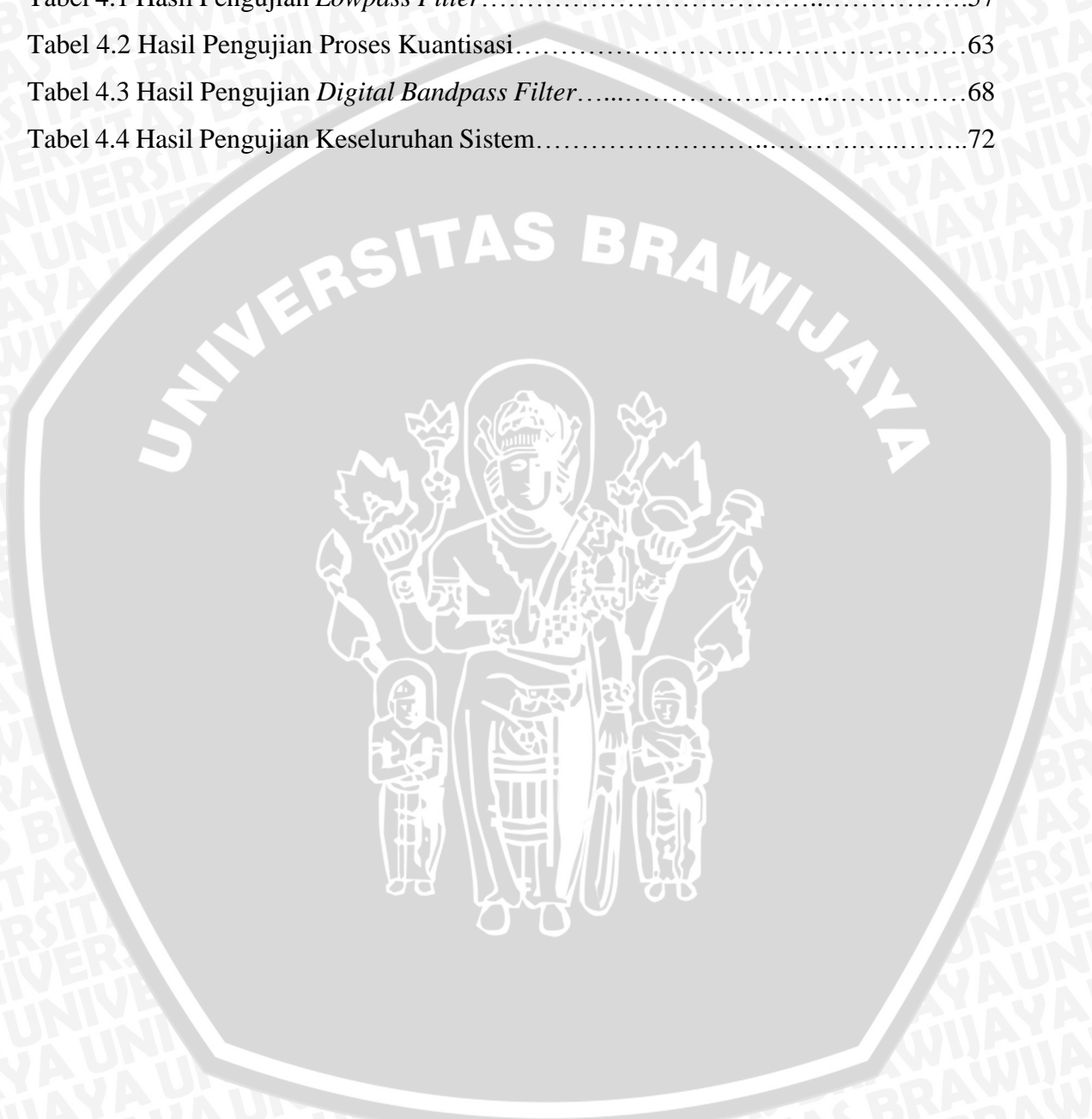
Gambar 4.27 Diagram Blok Pengujian Keseluruhan Sistem.....70

Gambar 4.28 Hasil Pengujian Keseluruhan Sistem.....71



DAFTAR TABEL

| No | Judul | Halaman |
|-----------|---|---------|
| Tabel 2.1 | Transformasi Frekuensi dari <i>LPF Prototype</i> ke Filter Lainnya..... | 15 |
| Tabel 2.2 | Polinomial <i>Butterworth</i> | 16 |
| Tabel 4.1 | Hasil Pengujian <i>Lowpass Filter</i> | 57 |
| Tabel 4.2 | Hasil Pengujian Proses Kuantisasi..... | 63 |
| Tabel 4.3 | Hasil Pengujian <i>Digital Bandpass Filter</i> | 68 |
| Tabel 4.4 | Hasil Pengujian Keseluruhan Sistem..... | 72 |



BAB I PENDAHULUAN

1.1 Latar Belakang

Kontes Robot Pemadam Api Indonesia (KRPAI) adalah salah satu kategori yang dilombakan dalam Kontes Robot Indonesia (KRI). Sistem robot KRPAI merupakan sistem robot cerdas, dimana robot diberikan kecerdasan buatan untuk melaksanakan tugasnya, yaitu menemukan api didalam sebuah labirin, kemudian memadamkannya. Untuk mengaktifkan robot ini dibutuhkan sinyal suara yang memiliki frekuensi 3,8 kHz. Nilai frekuensi suara ini sudah ditentukan oleh panitia dan tercantum pada *rule* perlombaan. Sinyal suara ini kemudian disebut sebagai suara aktivasi.

Sound activation adalah sistem pengolahan sinyal pada robot KRPAI yang berfungsi untuk mengenali suara aktivasi. Dalam perlombaan yang sesungguhnya, terdapat sumber audio lain disekitar robot yang mampu mengganggu sistem, antara lain musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara. Oleh karena itu, sistem ini membutuhkan metode pengolahan sinyal yang tahan terhadap *noise* agar tidak mudah terganggu oleh sumber audio lainnya. Untuk itu digunakan pengolahan sinyal secara digital karena kelebihanannya yang lebih tahan terhadap *noise* dibandingkan sistem pengolahan sinyal secara analog.

Untuk melakukan pengolahan sinyal secara digital dengan masukan berupa sinyal analog, dibutuhkan perangkat yang mampu untuk melakukan proses *Analog to Digital Converter* (ADC). Proses ini terdiri dari beberapa tahap yaitu, *sampling*, kuantisasi dan *coding*. Proses ADC berfungsi untuk mengubah sinyal analog menjadi sinyal diskrit yang akan diproses secara digital.

Filter digital adalah salah satu jenis pengolahan sinyal digital. Berdasarkan fungsinya untuk meloloskan frekuensi sinyal, filter digital terdiri dari *lowpass*, *highpass*, *bandpass*, dan *bandstop*. *Digital bandpass filter* adalah salah satu jenis filter digital yang mampu meloloskan sinyal yang memiliki frekuensi pada *range* diantara frekuensi batas bawah dan frekuensi batas atas. Penggunaan *digital bandpass filter* bertujuan untuk meloloskan sinyal dengan frekuensi 3,8 kHz, dan meredam sinyal dengan frekuensi lebih kecil dan lebih besar dari 3,8 kHz.

Keberadaan suara aktivasi dapat dideteksi dengan mengamati nilai amplitudo sinyal keluaran yang merupakan hasil komputasi menggunakan algoritma *digital bandpass filter*. Untuk menghindari *noise* yang terdeteksi, pembacaan nilai amplitudo sinyal keluaran dilakukan berulang kali sehingga tingkat selektifitas sistem semakin tinggi.

Dengan merancang *digital bandpass filter* untuk diaplikasikan pada *sound activation* diharapkan robot mampu mengenali suara aktivasi dengan baik, walau terdapat sumber audio lainnya disekitar robot.

1.2 Rumusan Masalah

Berdasarkan uraian pada latar belakang diatas maka pembahasan dalam penelitian ini ditekankan pada :

1. Bagaimana merancang diagram blok sistem *sound activation*.
2. Bagaimana merancang *digital bandpass filter* untuk diaplikasikan pada sistem *sound activation*.

1.3 Batasan Masalah

Berdasarkan pembahasan di atas, maka batasan masalah yang ditentukan adalah:

1. Suara aktivasi adalah sinyal suara yang memiliki frekuensi sebesar 3.8 kHz dan diaktifkan dengan durasi selama 5 detik.
2. Proses yang dilakukan adalah *Analog to Digital Converter* yang terdiri dari *sampling*, kuantisasi, dan *coding* serta proses komputasi menggunakan algoritma *digital bandpass filter*.
3. Menggunakan mikrokontroler ATmega8 sebagai prosesor untuk sistem *sound activation*, dengan kapasitas memori sebesar 8 KB.
4. Aplikasi robot yang dibahas adalah pada Kontes Robot Pemadam Api Indonesia (KRPAI).
5. Sumber audio disekitar robot adalah musik bambangan cakil (musik pengiring KRSI), suara tepuk tangan, dan suara orang berbicara.
6. Tidak membahas mengenai catu daya mikrokontroler.

1.4 Tujuan

Tujuan dari penelitian ini adalah merancang *digital bandpass filter* untuk aplikasi *sound activation* pada bidang robotika.

1.5 Kontribusi Penelitian

Penelitian ini dapat menghasilkan kontribusi berupa :

1. Hasil penelitian dapat diaplikasikan sebagai sistem *sound activation* pada robot KRPAI dalam Kontes Robot Indonesia (KRI) 2016.
2. Hasil penelitian dapat dikembangkan kembali untuk aplikasi pada bidang robotika maupun bidang lainnya.
3. Penulis mampu memiliki pemahaman yang lebih mendalam pada bidang pengolahan sinyal.

1.6 Sistematika Penulisan

Sistematika penulisan dan gambaran untuk setiap bab pada laporan penelitian ini adalah sebagai berikut:

BAB I PENDAHULUAN

Memuat latar belakang, rumusan masalah, batasan masalah, tujuan, kontribusi penelitian, dan sistematika penulisan.

BAB II DASAR TEORI

Membahas konsep-konsep dasar yang berhubungan dengan mikrofon, filter analog, filter digital, *analog to digital converter* (ADC), dan mikrokontroler.

BAB III METODOLOGI PENELITIAN

Menjelaskan tentang tahapan penyelesaian penelitian dari metode perancangan sistem, metode pengujian sistem, pembahasan dan pengambilan kesimpulan dan saran.

BAB IV HASIL DAN PEMBAHASAN

Membahas tentang perancangan blok diagram sistem, tahapan-tahapan perancangan sistem, memperoleh hasil melalui pengujian sistem dan, membahas hasil pengujian untuk mengetahui kesesuaian antara hasil pengujian sistem dengan perencanaan sistem.

BAB V PENUTUP

Memuat kesimpulan yang diperoleh terkait hasil pengujian sistem serta memberikan saran untuk memperbaiki dan pengembangan penelitian selanjutnya.

BAB II

TINJAUAN PUSTAKA

2.1 Umum

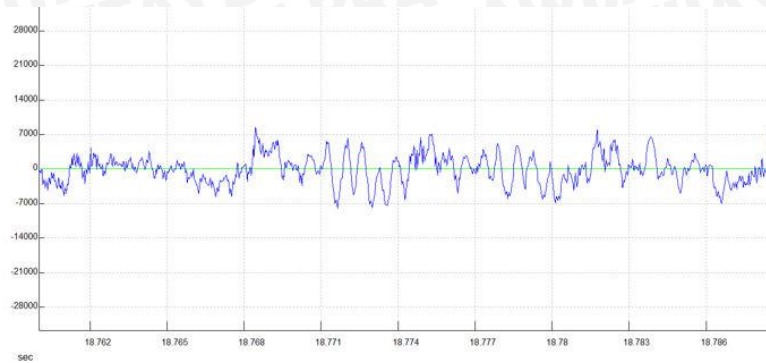
Perkembangan teknologi robotika di Indonesia khususnya pada kategori robot cerdas yang dituntut untuk berfungsi tanpa perlu dikendalikan oleh manusia, semakin cepat. Robot KRPAI membutuhkan sistem pengolahan sinyal yang mampu mendeteksi sumber audio berupa sinyal suara dengan frekuensi 3,8 kHz dan mengabaikan sumber audio lainnya di sekitar robot.

Salah satu metode yang dapat digunakan dalam sistem pengolahan sinyal adalah dengan mengimplementasikan desain *digital bandpass filter* yang berfungsi untuk meloloskan sinyal dengan *range* frekuensi tertentu dan meredam sinyal dengan frekuensi diluar *range* tersebut, dimana sinyal suara yang berupa sinyal analog akan dikonversi menjadi sinyal digital terlebih dahulu menggunakan proses *analog to digital converter* (ADC).

Untuk membuat sistem *sound activation* dibutuhkan beberapa teori, antara lain teori mengenai sinyal, mikrofon, *analog to digital converter* (ADC), filter analog, filter digital, dan mikrokontroler.

2.2 Sinyal

Secara umum sinyal didefinisikan sebagai suatu besaran fisis yang merupakan fungsi waktu, ruangan, atau beberapa variabel. Sinyal yang sering digunakan antara lain sinyal tegangan listrik (v) sebagai fungsi waktu, intensitas sebagai fungsi koordinat x , y , dan waktu. Dalam beberapa hal, sinyal tidak dapat dilihat hubungannya secara nyata dan kompleks. Sinyal pembicaraan (suara manusia) ditunjukkan pada Gambar 2.1.



Gambar 2.1 Sinyal Suara Manusia

Sumber: <http://www.assap.ac.uk>

Suatu sinyal mewakili beberapa jenis informasi yang dapat diamati, antara lain amplitudo, frekuensi, perbedaan fasa, dan gangguan akibat *noise*. Sinyal listrik dapat dianalisis dalam bentuk persamaan tegangan dan arus. Persamaan umum sinyal sinus dapat dinyatakan dalam Persamaan (2-1).

$$v(t) = A \sin(\omega t + \varphi) \quad (2-1)$$

dengan :

$v(t)$ = Nilai sesaat sinyal (volt)

A = Amplitudo sinyal (volt)

φ = Fasa sinyal ($^{\circ}$)

ω = $2\pi f$ (Hz)

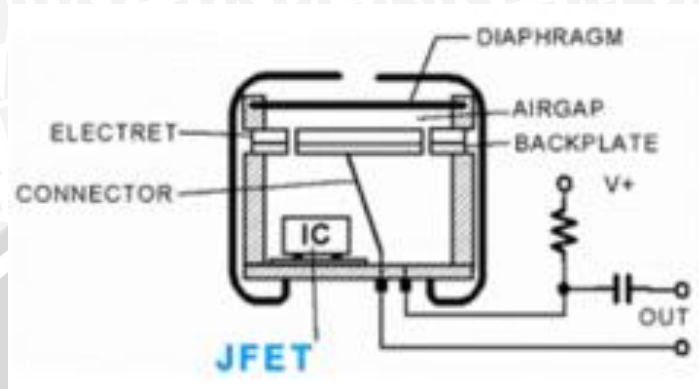
f adalah frekuensi sinyal, yang dapat diartikan sebagai banyaknya gelombang sinyal dalam satu detik, dengan satuan Hertz (Hz).

Sinyal dapat dikelompokkan menjadi 2 jenis yaitu sinyal analog dan sinyal digital. Sinyal analog adalah sinyal yang memiliki variasi keadaan yang sangat banyak, sedangkan sinyal digital adalah suatu sinyal yang dinyatakan dalam 2 keadaan, yaitu 0 dan 1 (biner).

2.3 Mikrofon

Mikrofon berfungsi untuk mengubah sinyal suara menjadi sinyal elektrik. Berdasarkan prinsip kerjanya, macam-macam mikrofon yang dikenal sekarang dapat dikelompokkan menjadi lima jenis, yaitu resistansi variabel, reluktansi variabel, induksi kumparan bergerak (*moving coil induction*), kapasitansi variabel, dan piezoelektrik.

Mikrofon elektret (*electret microphone*) adalah suatu jenis khusus dari mikrofon kapasitor (*condenser microphone*) yang sudah memiliki sumber muatan sendiri yang terpasang didalamnya, sehingga tidak memerlukan pencatu daya luar. Sumber muatan itu sebenarnya adalah sebuah alat penyimpan muatan (*charge storage device*).



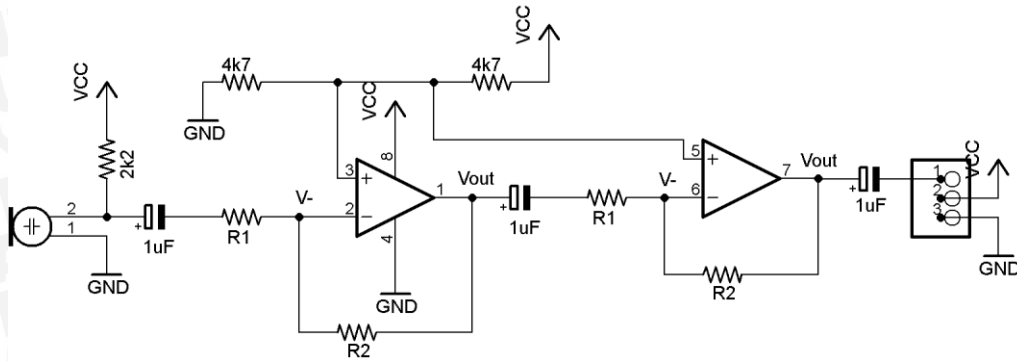
Gambar 2.2 Mikrofon Elektret

Sumber: Roddy dan Coolen (1984:111)

Dalam Gambar 2.2, *backplate* dari mikrofon dilapisi dengan suatu lapisan teflon yang bermuatan, lapisan teflon ini menyediakan sumber muatan tetap yang mengandung sejumlah besar muatan-muatan negatif yang terperangkap (*trapped*), yang mengimbas (*induce*)-kan padanya lewat sebuah resistor beban luar. Muatan yang terperangkap pada satu sisi dan muatan bayangan pada sisi yang lain menimbulkan medan listrik pada celah udara (*airgap*) yang membentuk kapasitor. Bila tekanan suara menyebabkan diafragma bergerak ke muka dan ke belakang, nilai kapasistansi dan tegangan terminal akan berubah.

Impedansi adalah suatu parameter nilai yang mengukur jumlah resistansi sebuah device terhadap arus. Impedansi menggabungkan efek kapasitansi, induktansi dan resistansi dari sebuah sinyal. Impedansi diukur dalam satuan ohm, dan dilambangkan dengan huruf O atau huruf Z. Setiap mikrofon mempunyai spesifikasi tersendiri mengenai impedansinya masing-masing, *electret condenser microphone* memiliki impedansi keluaran sebesar 2,2 k Ω .

Amplitudo sinyal keluaran mikrofon relatif kecil dan bernilai dalam orde milivolt (mV), agar sinyal keluaran mikrofon dapat diproses oleh mikrokontroler, sinyal dikuatkan terlebih dahulu dengan menggunakan penguat op-amp. Gambar rangkaian elektronik mikrofon dimana sudah terdapat penguat op-amp didalam rangkaiannya dapat dilihat pada Gambar 2.3.



Gambar 2.3 Rangkaian Elektronik Mikrofon dan Penguat Op-amp

Sumber: *Datasheet Electret Condenser Mikrofon*

Apabila tegangan pada titik diantara titik R_1 dan R_2 adalah v^- , maka

$$\frac{v_{out} - v^-}{R_2} = \frac{v^- - 0}{R_1}$$

$$\frac{v_{out} - v^-}{R_2} = \frac{v^-}{R_1}$$

$$\frac{v_{out}}{R_2} = \frac{v^-}{R_1} + \frac{v^-}{R_2}$$

$$v_{out} = \frac{R_2}{R_1} v^- + v^-$$

$$v_{out} = \left(\frac{R_2}{R_1} + 1 \right) v^-$$

Karena $v^+ = v^- = v_{in}$, maka persamaan menjadi

$$v_{out} = \left(\frac{R_2}{R_1} + 1 \right) v_{in}$$

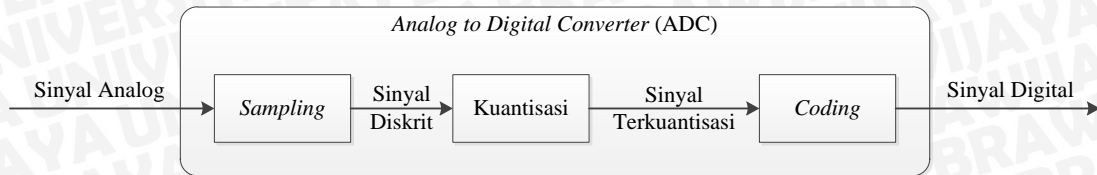
$$v_{out} = G \cdot v_{in}$$

$$G = \frac{R_2}{R_1} + 1$$

(2-2)

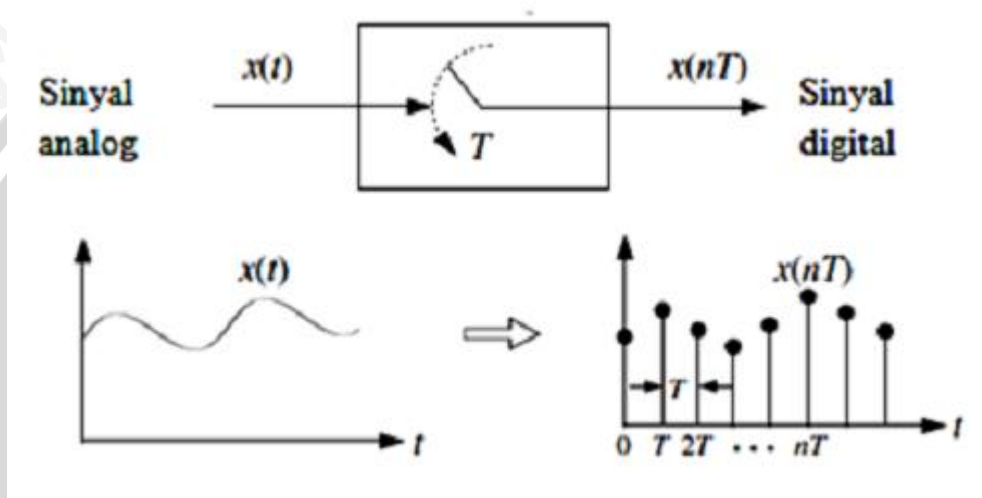
2.4 Analog to Digital Converter (ADC)

Analog to Digital Converter (ADC) adalah suatu proses pengubahan suatu data kontinyu terhadap waktu (analog) menjadi suatu data diskrit terhadap waktu (digital). ADC terbagi menjadi 3 proses antara lain, *sampling*, kuantisasi, dan *coding*. Proses ADC ditunjukkan dalam Gambar 2.4.



Gambar 2.4 Proses Analog to Digital Converter (ADC)

Proses *sampling* adalah suatu proses mengubah sinyal analog menjadi sinyal diskrit yang mewakili informasi dari sinyal analog. Dalam Gambar 2.5 ditunjukkan proses sampling beserta hasil *sampling* yaitu sinyal diskrit.



Gambar 2.5 Sinyal Hasil *Sampling*

Sumber: Sophocles (2010:5)

Dalam pengambilan sampel diperlukan saklar yang dikendalikan oleh pengatur waktu (*timing circuit*) yang terdiri atas komponen digital. Untuk dapat mengamati setiap perubahan yang terjadi pada sebuah sinyal analog (dalam hal ini perubahan level amplitudo), dapat diketahui salah satu caranya dengan mengamati pada periode waktu tertentu.

Untuk mendapatkan sinyal diskrit yang mampu mewakili sifat sinyal aslinya, proses sampling harus memenuhi syarat *Nyquist*, yaitu:

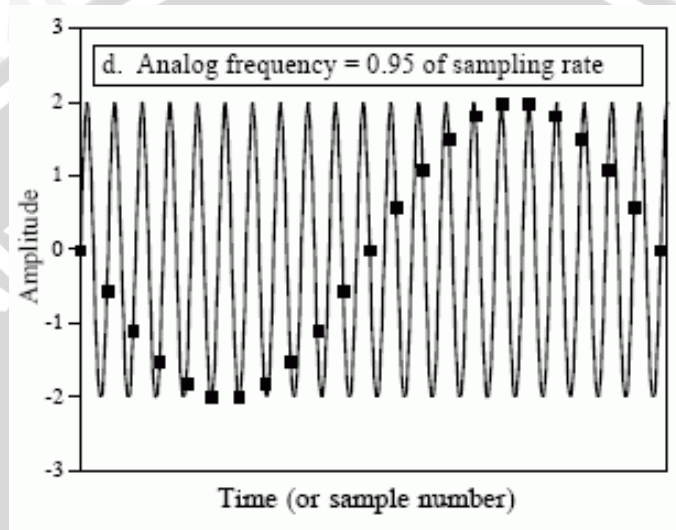
$$f_s \geq 2f_{in} \quad (2-3)$$

dengan:

f_s = frekuensi sinyal *sampling* (Hz)

f_{in} = frekuensi sinyal informasi (Hz)

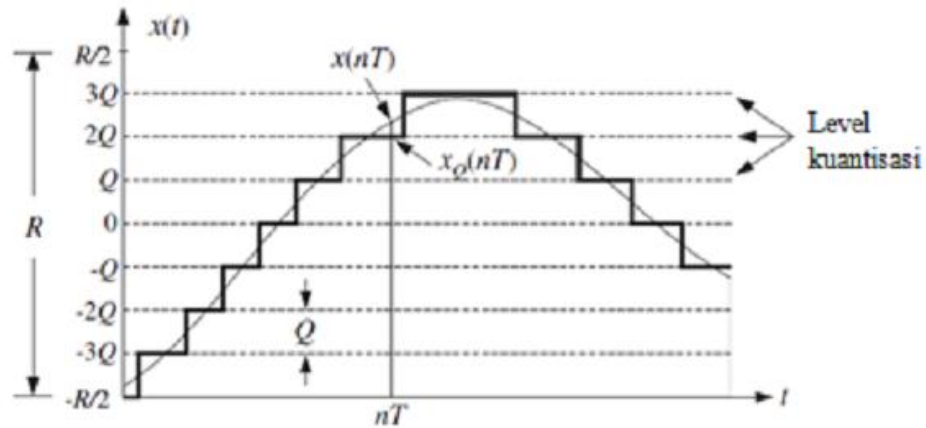
Fenomena *aliasing* pada proses *sampling* akan muncul pada sinyal diskrit hasil *sampling* apabila frekuensi *sampling* tidak memenuhi syarat *Nyquist*. Disebut *aliasing* karena frekuensi tertentu terlihat sebagai frekuensi yang lain (menjadi alias dari frekuensi lain). Semakin tinggi frekuensi sinyal *sampling*, maka sinyal diskrit yang dihasilkan semakin mendekati sinyal asli, dan informasi yang hilang semakin sedikit. Gambar 2.6 menunjukkan *aliasing* yang terjadi ketika besar frekuensi sinyal informasi mendekati besar frekuensi *sampling*.



Gambar 2.6 Fenomena *Aliasing*

Sumber: <http://www.dspguide.com>

Proses kuantisasi adalah proses pemetaan dari sinyal diskrit menjadi sinyal dengan level tertentu berdasarkan bit ADC yang digunakan. Setiap pemetaan akan menghasilkan 2^n rentang (R), dan $2^n - 1$ level. Dalam Gambar 2.7 ditunjukkan proses kuantisasi dengan menggunakan 3 bit ADC yang akan menghasilkan $2^3 = 8$ rentang dan $2^3 - 1 = 7$ level.



Gambar 2.7 Sinyal Hasil Kuantisasi

Sumber: Sophocles (2010:62)

Hasil proses kuantisasi ditunjukkan dalam Persamaan (2-3)

$$\text{Kuantisasi} = \frac{v_{in}}{v_{ref}} \times (2^n - 1) \quad (2-3)$$

dengan:

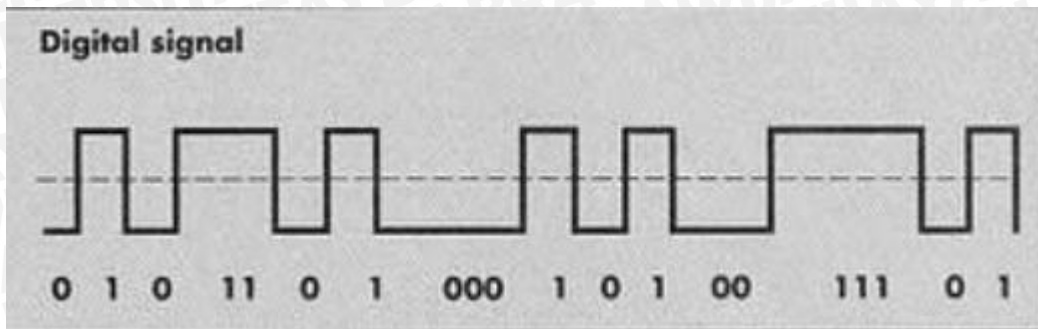
v_{in} = tegangan masukan proses kuantisasi (volt)

v_{ref} = tegangan referensi proses kuantisasi (volt)

n = besar bit kuantisasi

Semakin tinggi besar bit kuantisasi yang digunakan, proses kuantisasi akan semakin teliti memproses data analog menjadi data digital.

Proses *coding* merupakan proses perubahan suatu besaran tertentu ke dalam bentuk lain yang dikenali berdasarkan ketentuan yang teratur. Untuk mengubah sinyal terkuantisasi menjadi sinyal digital diberikan batas kuantisasi. Apabila suatu nilai memenuhi batas tersebut, akan direpresentasikan menjadi nilai biner 1 (logika tinggi) sedangkan untuk keadaan sebaliknya direpresentasikan menjadi nilai biner 0 (logika rendah). Sinyal hasil proses *coding* ditunjukkan dalam Gambar 2.8.

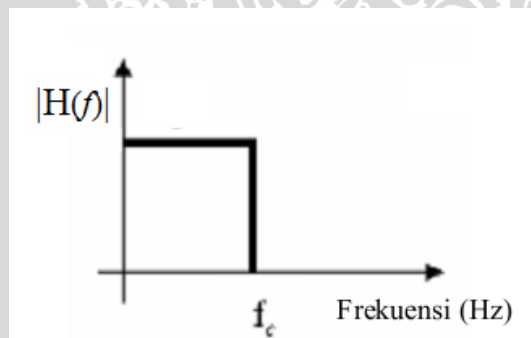


Gambar 2.8 Sinyal Hasil Coding

Sumber: Sophocles (2010:76)

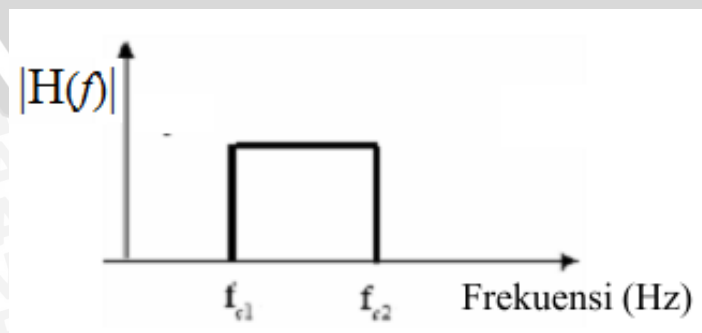
2.5 Filter Analog

Filter adalah suatu rangkaian yang berfungsi untuk meloloskan dan meredam frekuensi dari suatu sinyal. Berdasarkan sifat ini, filter dibagi menjadi 4, antara lain *lowpass filter* (LPF), *highpass filter* (HPF), *bandpass filter* (BPF), *bandreject filter* (BRF). Grafik karakteristik frekuensi LPF ideal dapat dilihat pada Gambar 2.9 dan grafik karakteristik frekuensi BPF ideal dapat dilihat pada Gambar 2.10.



Gambar 2.9 Karakteristik Frekuensi LPF Ideal

Sumber: Tanudjaja (2007:69)



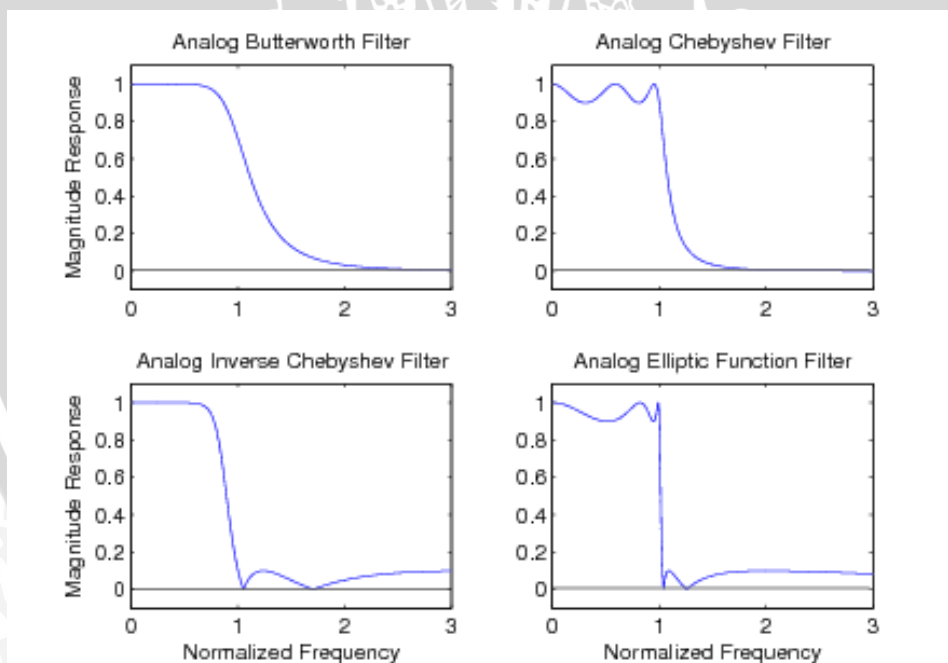
Gambar 2.10 Karakteristik Frekuensi BPF Ideal

Sumber: Tanudjaja (2007:70)

Berdasarkan pemrosesnya filter dapat dibedakan atas 2 bagian, yaitu filter analog dan filter digital. Filter analog dibagi menjadi 2, yaitu filter pasif dan filter aktif. Sedangkan filter digital dapat dibedakan menjadi 2 jenis, yaitu filter FIR (*Finite Impulse Response*) yang disebut filter nonrekursif dan IIR (*Infinite Impulse Response*) atau yang disebut filter rekursif.

Filter analog bentuk fisisnya direalisasikan dengan komponen – komponen pendukung : R, L, C, dan komponen aktif, sedangkan filter digital direalisasikan dalam bentuk algoritma (program) yang diturunkan dari model matematikanya dalam bentuk persamaan beda.

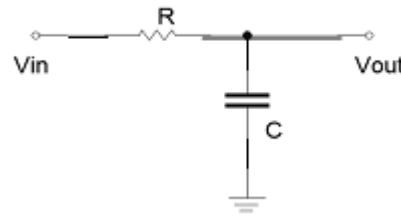
Secara umum, filter analog yang banyak digunakan sebagai dasar adalah filter *butterworth*, filter *chebyshev*, dan filter *elliptic*. Dalam penelitian ini digunakan filter analog *butterworth* sebagai filter dasar karena respon yang dihasilkan tidak memiliki *ripple* baik dalam daerah *passband* maupun pada daerah *stopband*. Dalam Gambar 2.11 ditunjukkan respon frekuensi dari filter *butterworth*, *chebyshev 1*, *chebyshev 2*, dan *elliptic*.



Gambar 2.11 Respon Frekuensi Filter *Butterworth* , *Chebyshev*, dan *Elliptic*

Lowpass filter adalah suatu jenis filter pasif yang memiliki karakteristik meloloskan sinyal dengan frekuensi yang lebih rendah dari frekuensi *cutoff* dan meredam sinyal dengan frekuensi yang lebih tinggi dari frekuensi *cutoff*.

Dalam Gambar 2.12 ditunjukkan rangkaian LPF analog orde 1. Fungsi alih dari rangkaian analog tersebut dinyatakan dalam Persamaan (2-4).



Gambar 2.12 Rangkaian *Lowpass Filter*

$$V_{in} = V_R + V_C$$

$$V_{in} = \frac{V_{in}}{R + \frac{1}{j\omega C}} R + V_C$$

$$V_C = V_{in} - \frac{V_{in}}{R + \frac{1}{j\omega C}} R$$

$$V_C = V_{in} \left(1 - \frac{1}{R + \frac{1}{j\omega C}} R \right)$$

$$V_C = V_{out} = V_{in} \left(1 - \frac{1}{R + \frac{1}{j\omega C}} R \right)$$

$$\frac{V_{out}}{V_{in}} = \frac{\frac{1}{j\omega C}}{R + \frac{1}{j\omega C}}$$

$$H(j\omega) = \frac{\frac{1}{j\omega C}}{R + \frac{1}{j\omega C}}$$

$$H(j\omega) = \frac{1}{1 + j\omega RC}$$

$$H(s) = \frac{1}{1 + sRC}$$

(2-4)

Respon frekuensi *lowpass filter* dapat ditentukan dengan Persamaan (2-5)

$$|H(j\omega)| = \frac{1}{\sqrt{1+(\omega RC)^2}} \quad (2-5)$$

$$\text{Saat } \omega = 0 \rightarrow |H(j\omega)| = 1$$

$$\omega = \infty \rightarrow |H(j\omega)| = 0$$

$$\omega = \frac{1}{RC} \rightarrow |H(j\omega)| = \frac{1}{\sqrt{2}} = 0,707$$

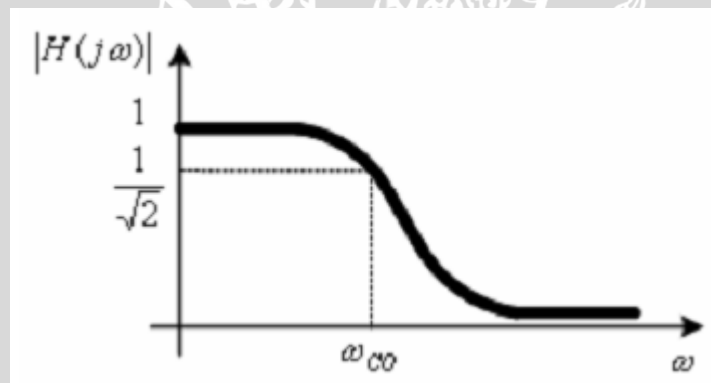
Frekuensi *cutoff lowpass filter* adalah nilai frekuensi ketika nilai $\omega = \frac{1}{RC}$ yang menyebabkan nilai $|H(j\omega)|$ turun menjadi $\frac{1}{\sqrt{2}}$ atau 0,707. Nilai frekuensi *cutoff lowpass filter* (f_c) ditentukan dengan Persamaan (2-6).

$$\omega = \frac{1}{RC}$$

$$2\pi f_c = \frac{1}{RC}$$

$$f_c = \frac{1}{2\pi RC}$$

(2-6)



Gambar 2.13 Respon Frekuensi *Lowpass Filter*

Sumber: Ramdhani (2005:222)

Suatu filter LPF dengan $\omega_c = 1$ rad/s dikenal sebagai filter ternormalisasi (*prototype*). Bila fungsi alih LPF ternormalisasi telah ditentukan, maka fungsi alih filter LPF, HPF, BPF, atau BRF dengan frekuensi $\omega_c > 1$ rad/s dapat ditentukan dengan melalui suatu proses transformasi frekuensi analog ke analog seperti terlihat pada Tabel 2.1.

Tabel 2.1 Transformasi Frekuensi dari LPF *prototype* ke filter lainnya

| No | Type Konversi | Transformasi Frekuensi |
|----|---------------|--|
| 1. | LPF ke LPF | $s \rightarrow \frac{s}{\omega_c}$ |
| 2. | LPF ke HPF | $s \rightarrow \frac{\omega_c}{s}$ |
| 3. | LPF ke BPF | $s \rightarrow \frac{s^2 + \omega_U \omega_L}{(\omega_U - \omega_L)s}$ |
| 4. | LPF ke BRF | $s \rightarrow \frac{(\omega_U - \omega_L)s}{s^2 + \omega_U \omega_L}$ |

Sumber: Tanudjaja (2007:77)

Besarnya orde filter n dapat dihitung dengan menggunakan Persamaan (2-7).

$$n = \frac{\log\left(\frac{10^{-k_1/10_{-1}}}{10^{-k_2/10_{-1}}}\right)}{2 \log \frac{\Omega_1}{\Omega_2}} \quad (2-7)$$

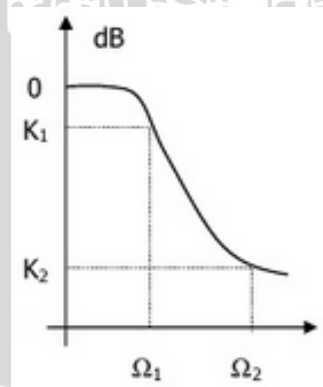
dengan :

Ω_1 = Frekuensi *Passband*

Ω_2 = Frekuensi *Stopband*

K_1 = Besar Penguatan *Passband*

K_2 = Besar Penguatan *Stopband*

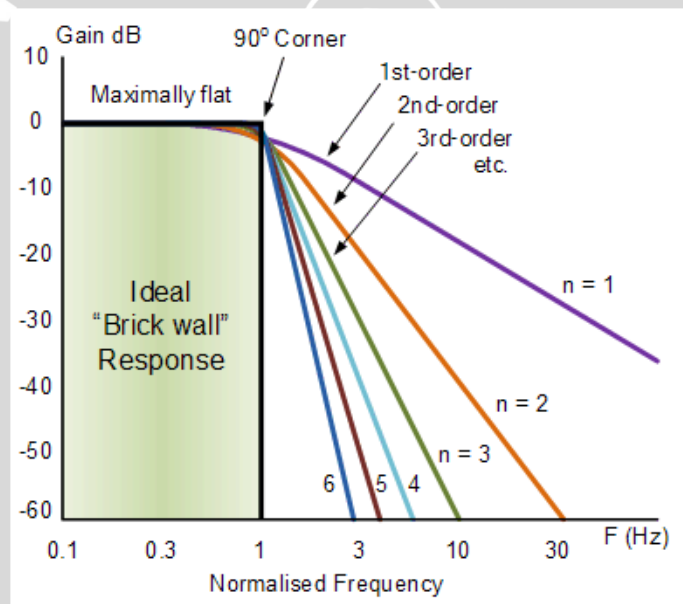
Gambar 2.14 Spesifikasi *Lowpass Filter Butterworth*

Jika orde filter sudah diketahui, fungsi alih filter analog $H(s)$ dapat ditentukan dengan menggunakan polinomial *butterworth*. Polinomial *butterworth* dapat dilihat secara lengkap pada Tabel 2.2.

Tabel 2.2 Polinomial *Butterworth*

| N | a ₀ | a ₁ | a ₂ | a ₃ | a ₄ | a ₅ | a ₆ | a ₇ | a ₈ | a ₉ |
|---|----------------|----------------|---------------------|---------------------|---------------------|---------------------|---------------------|--------------------|---------------------|-----------------|
| 1 | 1 | | | | | | | | | |
| 2 | 1 | 1.414s | 1s ² | | | | | | | |
| 3 | 1 | 2s | 2s ² | 1s ³ | | | | | | |
| 4 | 1 | 2.61s | 3.41s ² | 2.61s ³ | 1s ⁴ | | | | | |
| 5 | 1 | 3.24s | 5.24s ² | 5.24s ³ | 3.24s ⁴ | 1s ⁵ | | | | |
| 6 | 1 | 3.86s | 7.46s ² | 9.14s ³ | 7.46s ⁴ | 3.86s ⁵ | 1s ⁶ | | | |
| 7 | 1 | 4.49s | 10.10s ² | 14.61s ³ | 14.61s ⁴ | 10.10s ⁵ | 4.49s ⁶ | 1s ⁷ | | |
| 8 | 1 | 5.13s | 13.14s ² | 21.85s ³ | 25.69s ⁴ | 21.85s ⁵ | 13.14s ⁶ | 5.13s ⁷ | 1s ⁸ | |
| 9 | 1 | 5.76s | 16,58s ² | 31,16s ³ | 41,98s ⁴ | 41,98s ⁵ | 31,16s ⁶ | 5,76s ⁷ | 16,58s ⁸ | 1s ⁹ |

Sumber : <http://www.electronicshub.org>

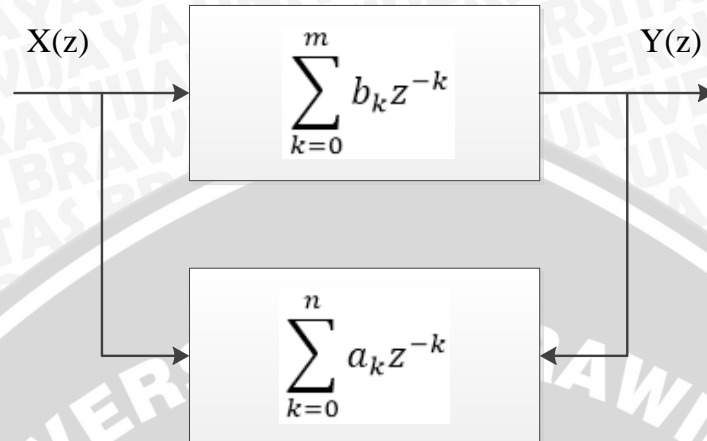
Gambar 2.15 Respon Frekuensi *Lowpass Filter* dengan Orde Berbeda

Sumber : <http://www.electronics-tutorials.ws>

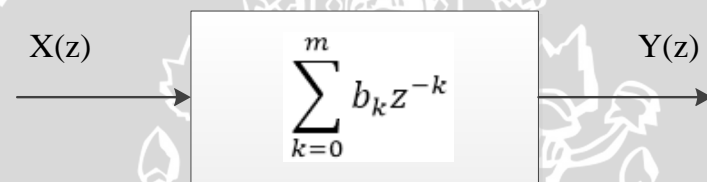
2.6 Filter Digital

Filter digital dibagi menjadi 2 jenis berdasarkan tanggapannya, yaitu *Infinite Impulse Response* (IIR) dan *Finite Impulse Response* (FIR). *Infinite Impulse Response* disebut response impulsnya tak terbatas (*infinite*) karena adanya *feedback* didalam filter, jika masukan filter IIR diberikan *impulse* (yaitu sebuah sinyal '1' diikuti dengan banyak sinyal '0'), maka *output* sistem filter IIR akan terus beresilasi karena adanya umpan balik (*feedback*), walaupun dalam kondisi praktis akan hilang pada suatu waktu.

Kelebihan dari filter IIR antara lain adalah membutuhkan koefisien yang lebih sedikit untuk respon frekuensi yang curam sehingga dapat mengurangi jumlah waktu perhitungan.



Gambar 2.16 Diagram Blok Filter IIR (*closed loop*)



Gambar 2.17 Diagram Blok Filter FIR (*open loop*)

Untuk merancang filter digital IIR diperlukan fungsi alih filter analog $H(s)$. dari fungsi alih analog ini akan diturunkan fungsi alih filter digital $H(z)$. Fungsi alih filter IIR dapat ditentukan dengan Persamaan (2-9).

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{\sum_{k=0}^N a_k z^{-k}} \quad (2-9)$$

Fungsi alih filter digital ini diperoleh dengan menggunakan metode transformasi *bilinear* yang berfungsi untuk menurunkan fungsi alih filter analog $H(s)$ menjadi fungsi alih filter digital $H(z)$. Perubahan dari frekuensi filter analog ke frekuensi filter digital bersifat nonlinier yaitu $s = j\omega$ adalah bidang datar sedangkan bidang $z = e^{j\omega}$ adalah bidang lingkaran. Untuk mengubah frekuensi analog menjadi frekuensi digital menggunakan Persamaan (2-10).

$$\omega_i = \Omega_i T_s = 2\pi \frac{f_i}{f_s}, i = 1, 2, \dots, N \quad (2-10)$$

dengan :

ω_i = frekuensi digital (rad)

Ω_i = frekuensi analog (rad/s)

T_s = periode *sampling* (s)

f_i = nilai frekuensi (Hz)

f_s = frekuensi *sampling* (Hz)

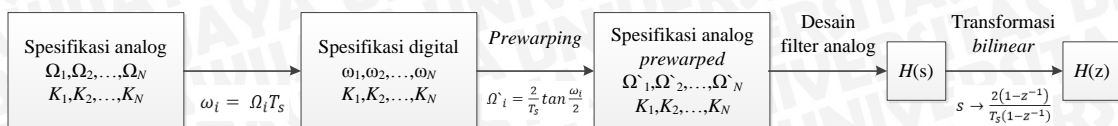
Karena hubungan antara frekuensi analog dengan frekuensi digital merupakan hubungan yang tidak linier, maka frekuensi yang digunakan dalam desain filter analog adalah frekuensi kritis analog yang telah terkoreksi Ω'_i : $i = 1, 2, \dots, N$ yang nilainya dapat diperoleh dengan menggunakan Persamaan (2-11).

$$\Omega'_i = \frac{2}{T_s} \tan \frac{\omega_i}{2}, i = 1, 2, \dots, N \quad (2-11)$$

Operasi ini dikenal sebagai *prewarping*. Persyaratan amplitudo untuk analog tidak diubah dan tetap sama dengan persyaratan pada filter digital. Sebuah filter analog $H(s)$ dirancang untuk memenuhi spesifikasi *prewarped* kemudian dilakukan transformasi *bilinear* pada $H(s)$, sehingga diperoleh filter digital $H(z)$ yang fungsinya dinyatakan dengan Persamaan (2-12).

$$H(z) = H(s) \Big|_s = \frac{2(1-z^{-1})}{T_s(1-z^{-1})} \quad (2-12)$$

Prosedur untuk desain filter digital dapat dilihat pada Gambar 2.18 yang terdiri dari empat langkah yaitu pengubahan spesifikasi filter analog menjadi spesifikasi filter digital, *prewarping* spesifikasi filter digital, merancang filter analog untuk mendapatkan fungsi alih filter analog $H(s)$, dan menerapkan transformasi *bilinear* untuk memperoleh fungsi alih filter digital $H(z)$.



Gambar 2.18 Prosedur untuk Desain Filter Digital

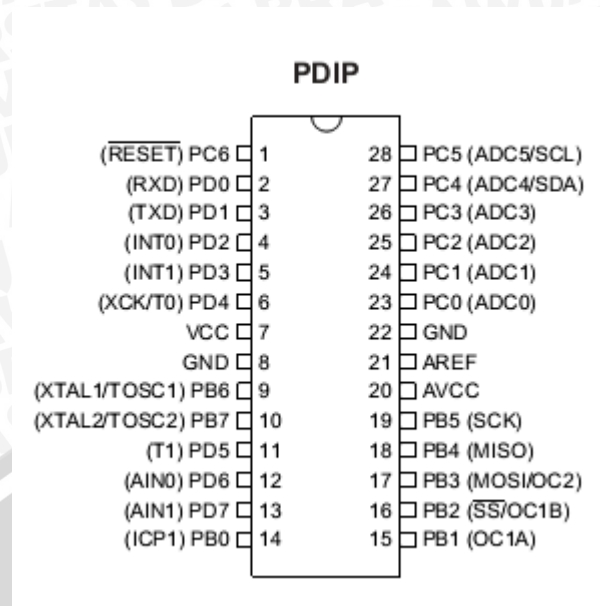
2.7 Mikrokontroler Atmel AVR ATmega8

Mikrokontroler adalah suatu IC (*integrated circuit*) yang memiliki kemampuan untuk membuat keputusan berdasarkan sinyal dari luar (*input*), dan berdasarkan algoritma yang dibentuk menjadi suatu program tertentu. Pada dasarnya mikrokontroler memiliki mikroprosesor, *timer*, *counter*, perangkat I/O (input/output) dan memori internal.

ATmega8 menurut *datasheet* adalah mikrokontroler CMOS 8-bit berarsitektur AVR RISC yang memiliki 8 KB *ISP Flash*. Dengan mengeksekusi instruksi dalam *clock* yang sangat cepat, ATmega 8 memiliki 15 *throughput* mendekati 1 MIPS per MHz, yang memungkinkan perancang sistem mengoptimalkan konsumsi daya terhadap kecepatan pemrosesan. Beberapa konfigurasi yang dimiliki mikrokontroler ATmega8 adalah sebagai berikut :

- Sebuah CPU 8 bit.
- Jalur dua arah (*bidirectional*) yang digunakan sebagai saluran masukan atau keluaran yang dikontrol oleh *register* DDR.
- Dua *timer/counter* 8 bit dan sebuah *timer/counter* 16 bit.
- 23 jalur *programable I/O (input/output)*.
- Analog to Digital Converter (ADC) 10-bit dan *analog comparator* di dalam *chip*.
- Osilator internal dan rangkaian pewaktu.
- Sebuah komunikasi serial USART yang dapat diprogram.
- Sebuah *master/slave* serial SPI yang dapat diprogram.
- Sebuah *Two Wire Serial Interface*.

Mikrokontroler ATmega8 mempunyai 28 pin yang memiliki fungsi tersendiri. susunan masing-masing pin ditunjukkan dalam Gambar 2.19.



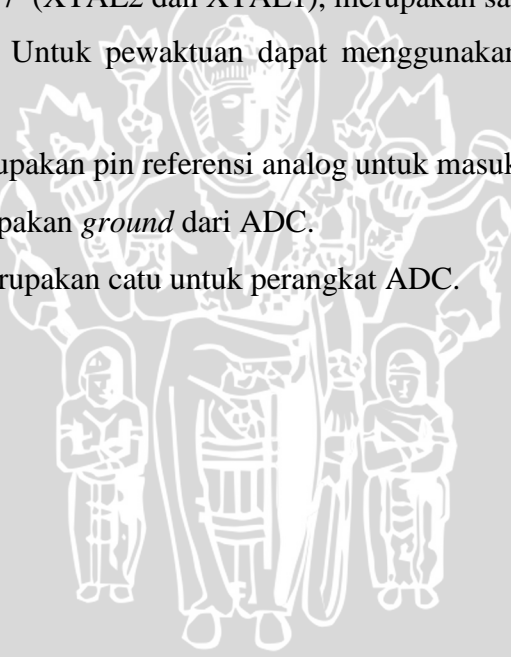
Gambar 2.19 Konfigurasi Pin ATmega8 Model PDIP

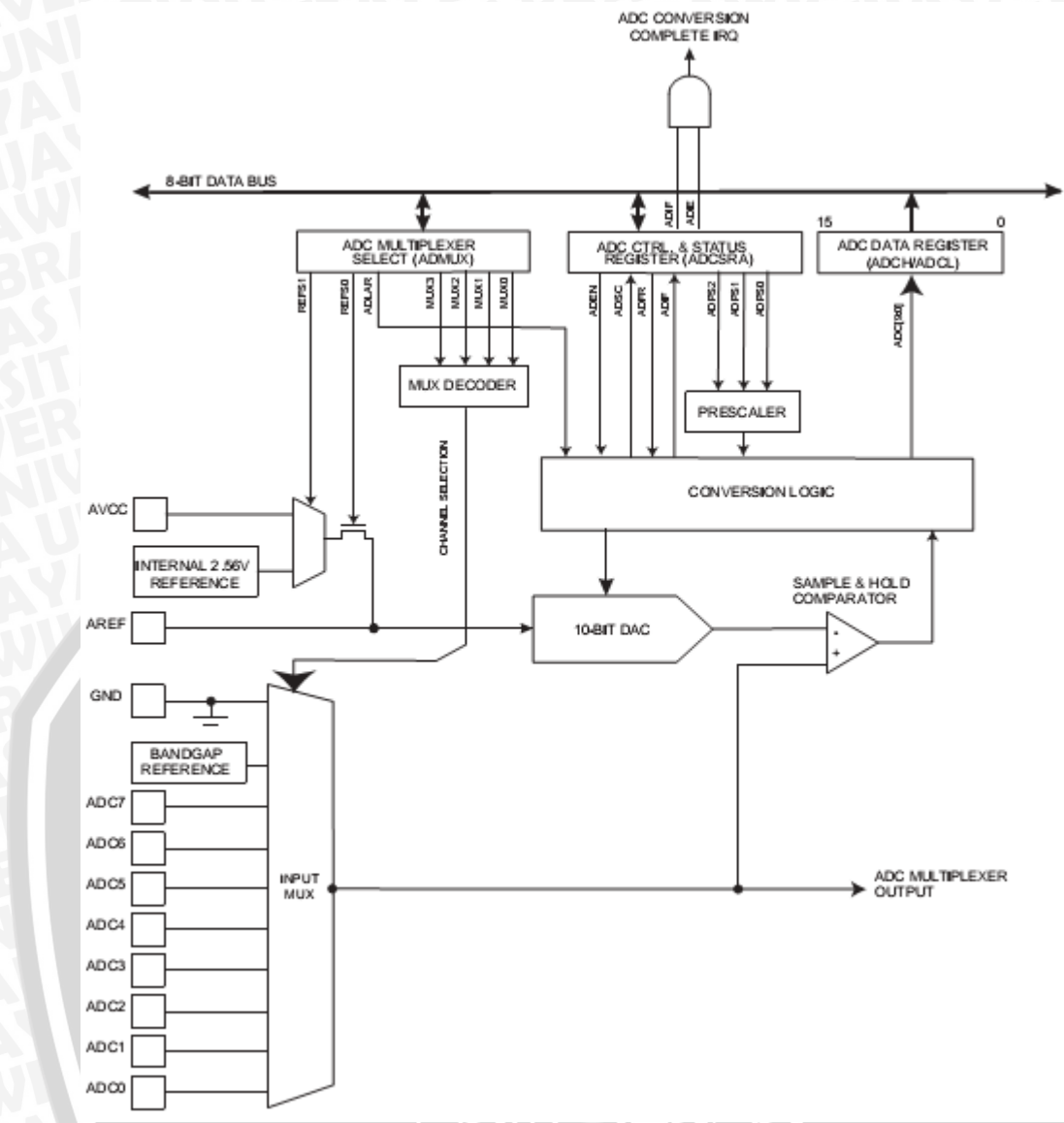
Sumber : *Datasheet ATmega8*

Fungsi kaki-kaki PIN dalam ATmega8 sesuai dengan *datasheet* ATmega8 antara lain sebagai berikut:

- Port B (Pin B0..7), merupakan saluran masukan atau keluaran dua arah yang juga mempunyai fungsi khusus. Fungsi khusus Port B diantaranya adalah Port B0 {ICP (*Timer/counter input capture pin*)}, Port B1 {OC1A (*Timer/counter 1 output compare A match output*)}, Port B2 yang bisa digunakan sebagai SS (*SPI slave select input*) atau OC1B (*Timer/counter 1 output compare B match output*), Port B3 yang bisa digunakan sebagai MOSI (*SPI bus master output/slave input*) atau OC2 (*timer/counter 2 compare match output*), Port B4 {MISO (*SPI bus master input/slave output*)}, Port B7 {SCK (*SPI bus serial clock*)}, Port B5 {SCK (*SPI bus serial clock*)}, Port B6 (XTAL1 & TOSC1 (*Timer Oscilator pin1*)), dan Port B7 (XTAL2 & TOSC2 (*Timer Oscilator pin2*)).
- Port C (Pin C0..5), merupakan saluran masukan/keluaran dua arah yang juga mempunyai fungsi khusus. Fungsi khusus Port C adalah sebagai ADC (*input ADC channel 0..5*), selain itu pada beberapa Port memiliki fungsi lain yaitu Port C4 (SDA (*Two-Wire serial bus data input/output line*)), dan Port C5 (SCL (*Two-Wire serial bus clock line*)).

- Pin C6 memiliki fungsi *reset*, merupakan saluran dua masukan untuk mereset mikrokontroler dengan memberi masukan pin *reset* logika rendah.
- Port D (Pin D0..7), merupakan saluran masukan/keluaran dua arah yang juga mempunyai fungsi khusus. Fungsi khusus dari Port D diantaranya adalah : Port D0 {RXD (*USART input pin*)}, Port D1 {TXD (*USART output pin*)}, Port D2 {INT0 (*Eksternal interupt 0 input*)}, Port D3 {INT1 (*Eksternal interupt 1 input*)}, Port D4 (T0 (*timer/counter0 eksternalcounter input*) & XCK (*USART eksternal clock input/output*), Port D5 (T1 (*timer/counter eksternal counter input*)), Port D6 (AIN0 (*Analog comparator positive input*)) dan Port D7 (AIN1 (*Analog comparator negative input*)).
- Pin 7 VCC, merupakan masukan untuk catu daya positif DC sebesar 5 volt.
- Pin 8 GND, merupakan *ground* dari seluruh rangkaian.
- Pin B6 dan Pin B7 (XTAL2 dan XTAL1), merupakan saluran untuk mengatur pewaktuan sistem. Untuk pewaktuan dapat menggunakan pewaktuan internal maupun eksternal.
- Pin 21 AREF, merupakan pin referensi analog untuk masukan ADC.
- Pin 22 GND, merupakan *ground* dari ADC.
- Pin 20 AVCC, merupakan catu untuk perangkat ADC.

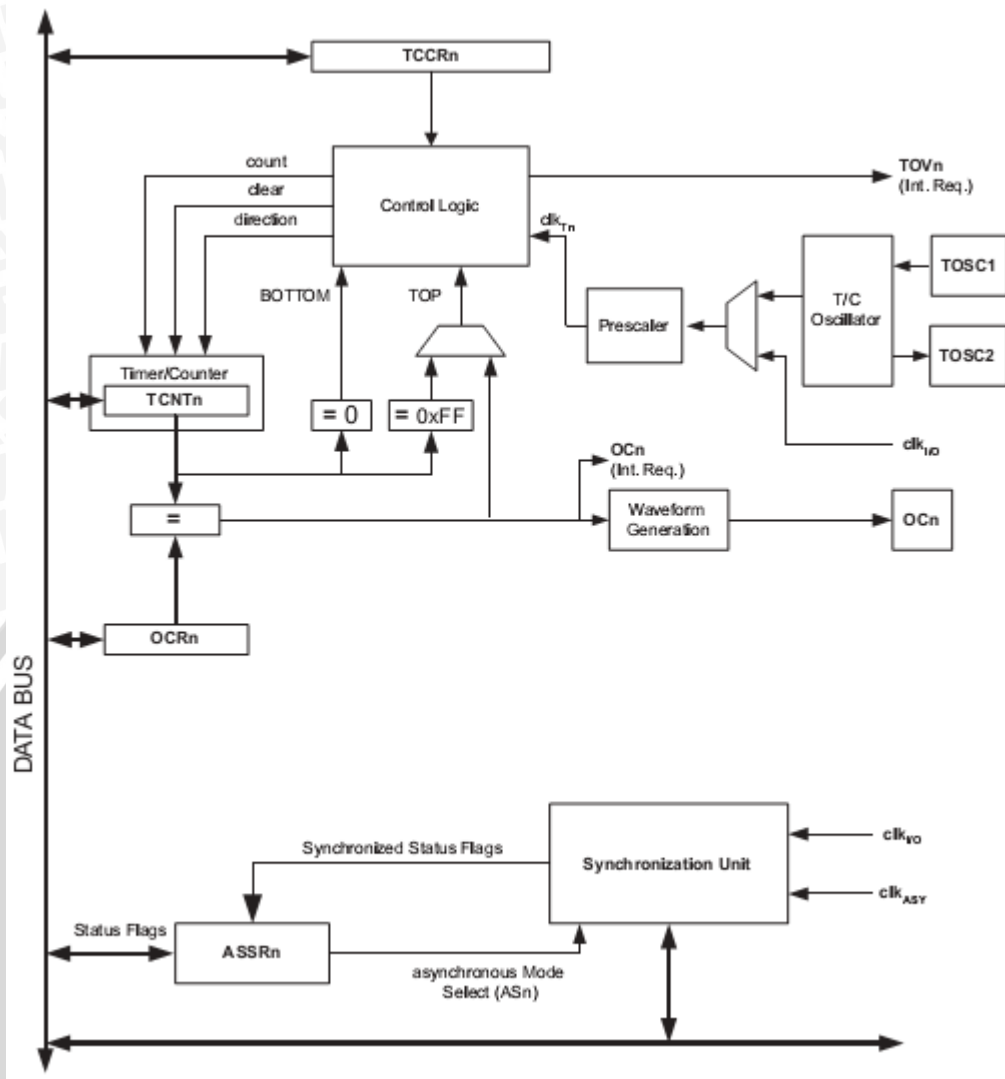




Gambar 2.20 Diagram Blok ADC ATmega8

Dalam Gambar 2.20 ditunjukkan diagram blok proses *Analog to Digital Converter* (ADC) pada mikrokontroler ATmega8. Kanal masukan analog dipilih dengan mengatur *MUX bits* dalam register ADMUX. ADC diaktifkan dengan mengatur *ADC Enable bit*, ADEN dalam register ADCSRA. Hasil konversi ditampilkan pada *ADC Data Registers*, ADCH dan ADCL. ADC memiliki fasilitas *interrupt* yang dapat dipicu ketika proses konversi telah selesai.





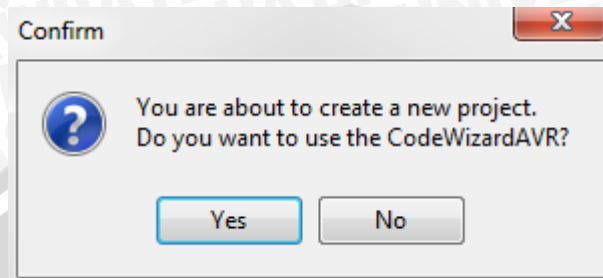
Gambar 2.21 Diagram Blok *Timer2* ATmega 8

Dalam Gambar 2.21 ditunjukkan diagram blok *timer2* pada mikrokontroler ATmega 8. *Timer2* memiliki ketelitian sebesar 8 bit yang melakukan pencacahan dimulai dari 0 hingga 255. Namun pencacahan dapat dimulai dari angka berapa pun selama tidak melebihi angka maksimum yang bisa dihitung pada register TCNT2. Ketika nilai pencacahan sudah mencapai maksimum, maka pencacahan akan dimulai dari awal kembali. Proses ini akan terus berulang sehingga memungkinkan mikrokontroler melakukan kegiatan yang bersifat periodik dengan waktu yang dapat diatur. Fungsi *timer2* ini dimanfaatkan untuk mengatur nilai frekuensi *sampling* pada proses *sampling*.

Untuk menulis program pada IC ATmega8 *software* yang dapat digunakan adalah CV AVR (*Code Vision AVR*). CV AVR adalah suatu *software compiler* berbahasa C yang dapat digunakan dalam beberapa IC ATmega, ATTiny, dan beberapa

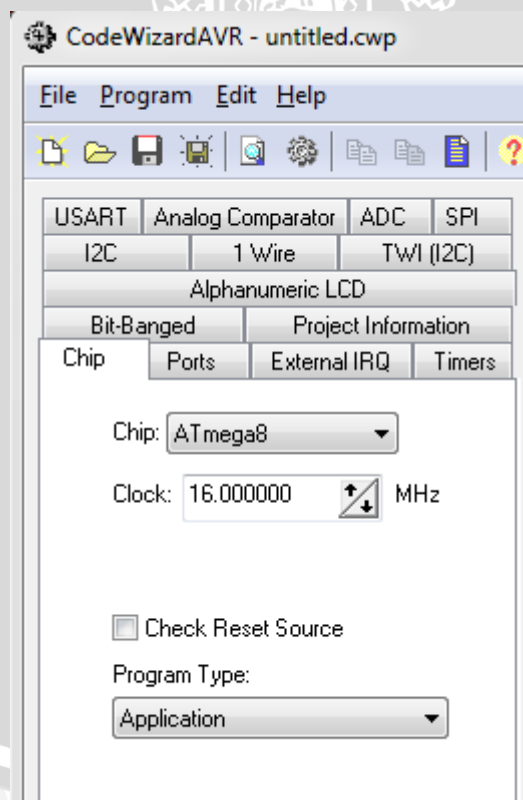
IC lainnya. Misalnya untuk mengaktifkan fitur ADC (*Analog to Digital Converter*) pada mikrokontroler ATmega8, langkah yang dilakukan adalah sebagai berikut :

1. Buka *software* CV AVR.
2. Buka *CodeWizardAVR* untuk memudahkan pemrograman.



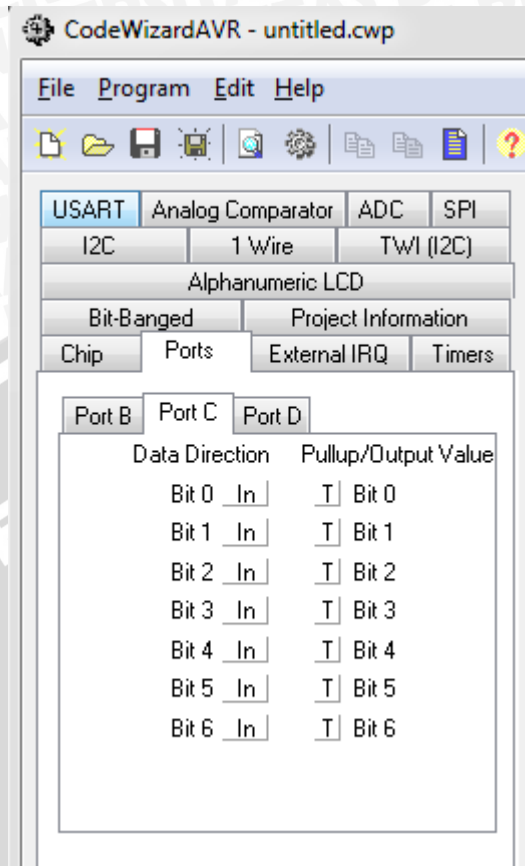
Gambar 2.22 Jendela *CodeWizardAVR*

3. Pilih jenis *chip* mikrokontroler yang digunakan



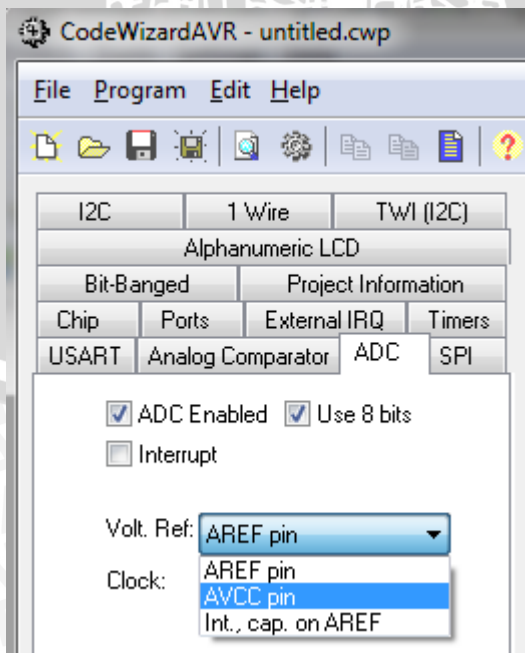
Gambar 2.23 Pemilihan Jenis *Chip*

4. Aktifkan Port C sebagai *input*



Gambar 2.24 Konfigurasi Port C Sebagai *Input*

5. Pilih tab ADC dan beri tanda \checkmark pada "*ADC enable*"



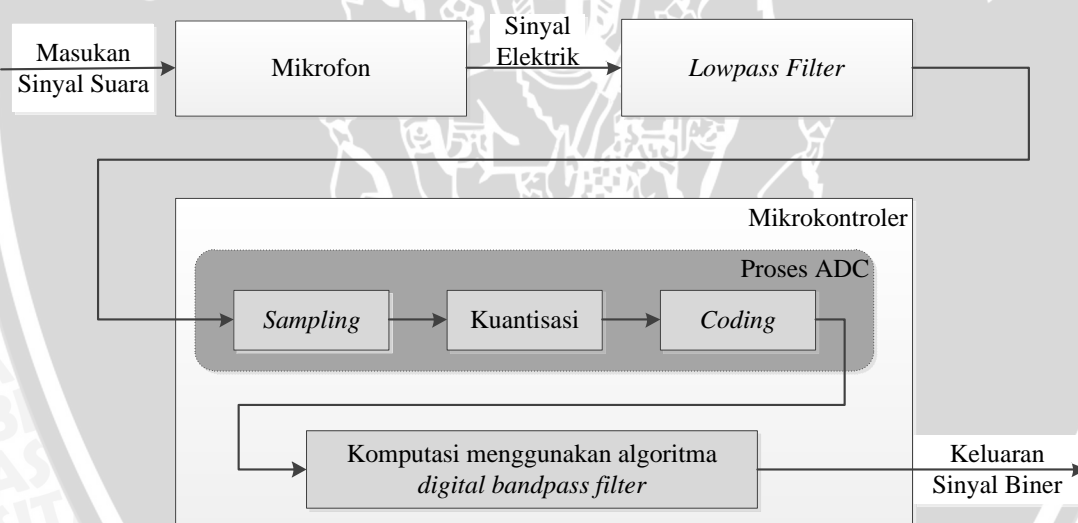
Gambar 2.25 Mengaktifkan ADC dari mikrokontroler

BAB III METODE PENELITIAN

Metode kajian yang digunakan dalam penelitian ini bersifat aplikatif mengenai bagaimana rancang bangun *digital bandpass filter* untuk diaplikasikan pada sistem *sound activation*. Langkah-langkah yang perlu dilakukan untuk merealisasikan sistem yang dirancang adalah perancangan sistem, pengujian sistem, pembahasan, dan pengambilan kesimpulan dan saran.

3.1 Perancangan Sistem

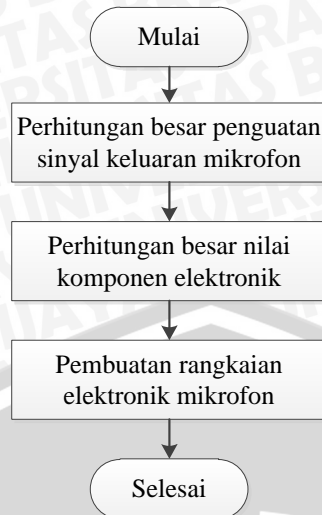
Perancangan sistem merupakan perancangan alat secara umum yang diperlukan dalam sistem *sound activation*. Diagram blok sistem yang dirancang ditunjukkan dalam Gambar 3.1.



Gambar 3.1 Diagram Blok Sistem yang Dirancang

3.1.1 Perancangan Mikrofon

Mikrofon berfungsi untuk mengubah sinyal suara menjadi sinyal elektrik. Amplitudo sinyal keluaran mikrofon relatif kecil dan bernilai dalam orde milivolt, untuk itu dibutuhkan rangkaian penguat untuk memperbesar amplitudo sinyal keluaran mikrofon. Dalam penelitian ini, rangkaian penguat sudah termasuk dalam rangkaian elektronik pada mikrofon. Diagram alir perancangan mikrofon ditunjukkan dalam Gambar 3.2.



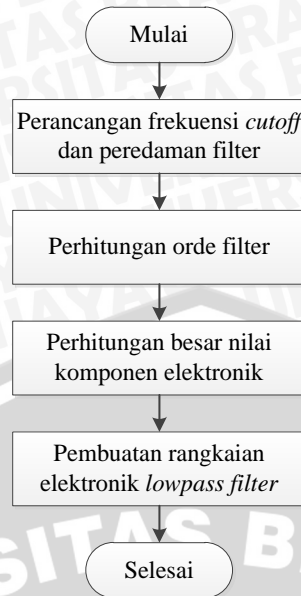
Gambar 3.2 Diagram Alir Perancangan Mikrofon

Tahapan yang dilakukan dalam perancangan mikrofon antara lain :

1. Perhitungan besar penguatan sinyal keluaran mikrofon.
2. Perhitungan besar nilai komponen elektronik.
3. Pembuatan rangkaian elektronik mikrofon.

3.1.2 Perancangan *Lowpass Filter*

Lowpass filter berfungsi untuk meloloskan sinyal yang memiliki frekuensi lebih kecil dari frekuensi *cutoff* dan meredam sinyal yang memiliki frekuensi lebih besar dari frekuensi *cutoff*. Dalam penelitian ini *lowpass filter* digunakan untuk meloloskan sinyal yang memiliki frekuensi lebih kecil dari setengah frekuensi *sampling* ($f \leq \frac{1}{2}f_s$) yang bertujuan agar tidak mengganggu proses *sampling*. Diagram alir perancangan *lowpass filter* ditunjukkan dalam Gambar 3.3.



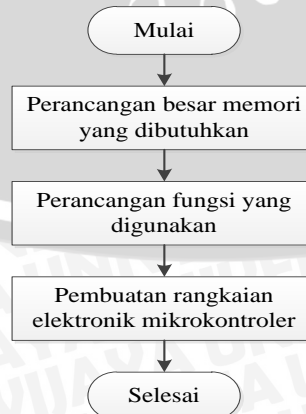
Gambar 3.3 Diagram Alir Perancangan *Lowpass Filter*

Tahapan yang dilakukan dalam perancangan *lowpass filter* antara lain :

1. Perancangan frekuensi *cutoff* dan peredaman filter.
2. Perhitungan orde filter.
3. Perhitungan besar nilai komponen elektronik yang digunakan.
4. Pembuatan rangkaian elektronik *lowpass filter*.

3.1.3 Perancangan Mikrokontroler

Dalam penelitian ini, mikrokontroler berfungsi untuk melakukan proses ADC yang terdiri dari proses *sampling*, proses kuantisasi, proses *coding* kemudian melakukan proses komputasi menggunakan algoritma *digital bandpass filter*. Diagram alir perancangan mikrokontroler ditunjukkan dalam Gambar 3.4.



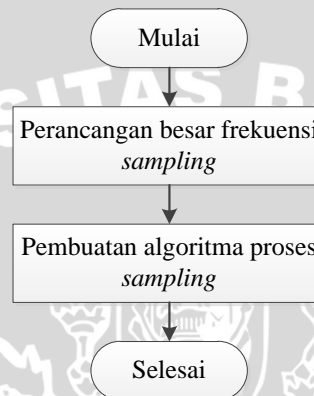
Gambar 3.4 Diagram Alir Perancangan Mikrokontroler

Tahapan yang dilakukan dalam perancangan mikrokontroler antara lain :

1. Perancangan besar memori yang dibutuhkan.
2. Perancangan fungsi yang digunakan pada mikrokontroler.
3. Pembuatan rangkaian elektronik mikrokontroler.

3.1.4 Perancangan Proses *Sampling*

Proses *sampling* bertujuan untuk mengubah sinyal kontinyu menjadi sinyal diskrit. Diagram alir perancangan proses *sampling* ditunjukkan dalam Gambar 3.5.



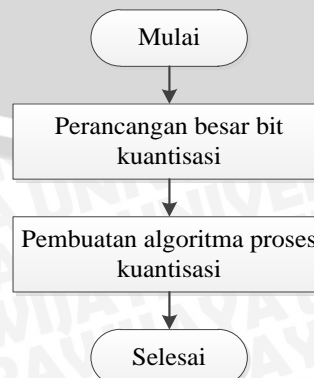
Gambar 3.5 Diagram Alir Perancangan Proses *Sampling*

Tahapan yang dilakukan dalam perancangan proses *sampling* antara lain :

1. Perancangan besar frekuensi *sampling* yang digunakan.
2. Pembuatan algoritma proses *sampling* untuk diterapkan pada mikrokontroler.

3.1.5 Perancangan Proses Kuantisasi

Proses kuantisasi bertujuan untuk mengubah level tegangan / amplitudo sinyal kontinyu menjadi besaran diskrit. Diagram alir perancangan proses kuantisasi ditunjukkan dalam Gambar 3.6.



Gambar 3.6 Diagram Alir Perancangan Proses Kuantisasi

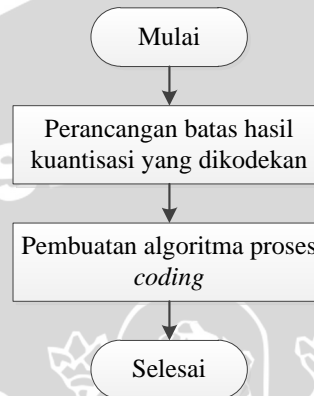
Tahapan yang dilakukan dalam perancangan proses kuantisasi antara lain :

1. Perancangan besar bit kuantisasi yang digunakan.
2. Pembuatan algoritma proses kuantisasi untuk diterapkan pada mikrokontroler.

3.1.6 Perancangan Proses Coding

Proses *coding* bertujuan untuk mengkodekan besaran diskrit dari hasil kuantisasi.

Diagram alir perancangan proses *coding* ditunjukkan dalam Gambar 3.7.



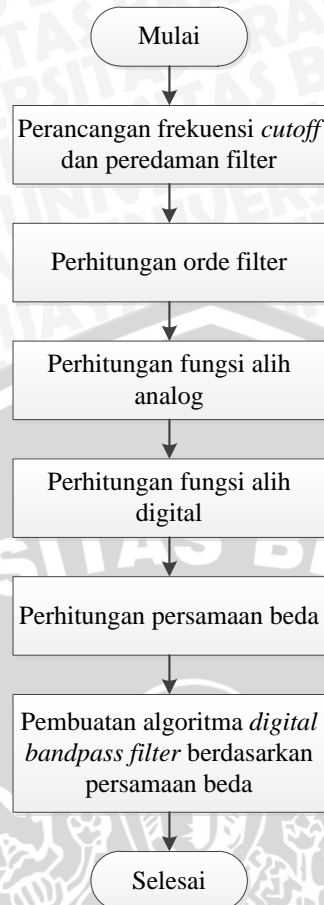
Gambar 3.7 Diagram Alir Perancangan Proses Coding

Tahapan yang dilakukan dalam perancangan proses *coding* antara lain :

1. Perancangan batas hasil kuantisasi yang dikodekan.
2. Pembuatan algoritma proses *coding* untuk diterapkan pada mikrokontroler.

3.1.7 Perancangan Digital Bandpass Filter

Dalam penelitian ini, *digital bandpass filter* berfungsi untuk meloloskan sinyal diskrit dengan frekuensi 3,8 kHz dan meredam sinyal diskrit dengan frekuensi lebih rendah dan lebih besar dari 3,8 kHz. Sinyal diskrit tersebut akan melalui proses komputasi menggunakan algoritma *digital bandpass filter*. Diagram alir perancangan *digital bandpass filter* ditunjukkan dalam Gambar 3.8.



Gambar 3.8 Diagram Alir Perancangan *Digital Bandpass Filter*

Tahapan yang dilakukan dalam perancangan *digital bandpass filter* antara lain :

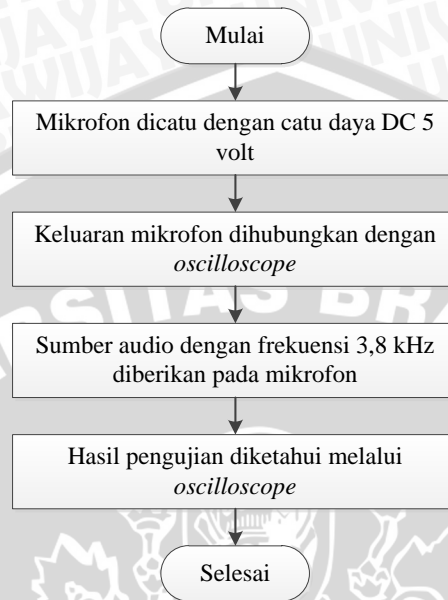
1. Perancangan frekuensi *cutoff* dan besar peredaman filter.
2. Perhitungan orde filter.
3. Perhitungan fungsi alih analog.
4. Perhitungan fungsi alih digital.
5. Perhitungan persamaan beda.
6. Pembuatan algoritma *digital bandpass filter* berdasarkan persamaan beda untuk diterapkan pada mikrokontroler.

3.2 Pengujian Sistem

Untuk mengetahui kinerja sistem dan kesesuaian antara sistem yang telah dibuat dengan sistem yang dirancang maka dilakukan pengujian sistem. Pengujian dilakukan pada masing-masing subsistem kemudian dilakukan pengujian keseluruhan sistem.

3.2.1 Pengujian Mikrofon

Pengujian mikrofon dilakukan untuk mengetahui kinerja mikrofon dalam menghasilkan sinyal elektrik dari sinyal suara yang diberikan. Diagram alir pengujian mikrofon ditunjukkan dalam Gambar 3.9.



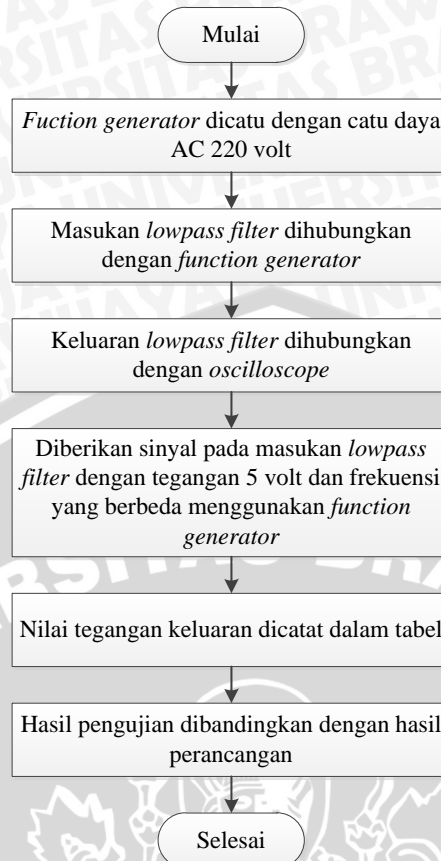
Gambar 3.9 Diagram Alir Pengujian Mikrofon

Langkah-langkah yang dilakukan dalam melakukan pengujian mikrofon antara lain :

1. Mikrofon dicatu dengan catu daya DC 5 volt.
2. Keluaran mikrofon dihubungkan dengan *oscilloscope*.
3. Sumber audio dengan frekuensi 3,8 kHz diberikan pada mikrofon.
4. Hasil pengujian diketahui melalui *oscilloscope* pada keluaran mikrofon.

3.2.2 Pengujian *Lowpass Filter*

Pengujian *lowpass filter* dilakukan untuk mengetahui kemampuan *lowpass filter* dalam meredam sinyal dengan nilai frekuensi yang lebih besar dari frekuensi *cutoff*. Diagram alir pengujian *lowpass filter* ditunjukkan dalam Gambar 3.10.



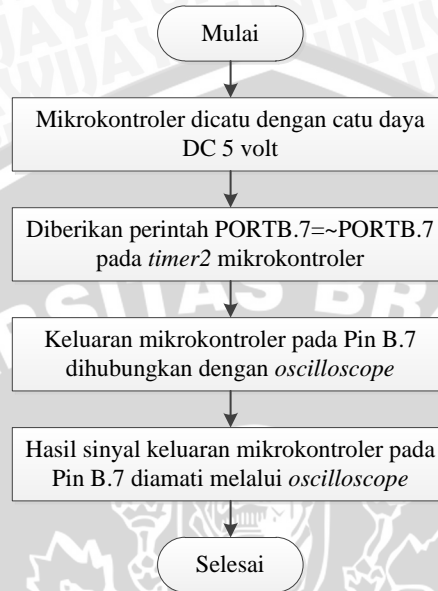
Gambar 3.10 Diagram Alir Pengujian *Lowpass Filter*

Langkah-langkah yang dilakukan dalam melakukan pengujian *lowpass filter* antara lain :

1. *Fuction generator* dicatu dengan catu daya AC 220 volt.
2. Masukan *lowpass filter* dihubungkan dengan *fuction generator*.
3. Keluaran *lowpass filter* dihubungkan dengan *oscilloscope*.
4. Diberikan sinyal pada masukan *lowpass filter* dengan tegangan 5 volt dan frekuensi yang berbeda menggunakan *fuction generator* yaitu 100 Hz, 500 Hz, 1000 Hz, 1500 Hz, 2000 Hz, 2500 Hz, 3000 Hz, 3500 Hz, 4000 Hz, 4500 Hz, 5000 Hz, 5500 Hz, 6000 Hz, 6500 Hz, 7000 Hz, 7500 Hz, 8000 Hz, 8500 Hz, 9000 Hz, 9500 Hz, dan 10000 Hz.
5. Nilai tegangan keluaran dicatat dalam tabel.
6. Hasil pengujian dibandingkan dengan hasil perancangan secara teori.

3.2.3 Pengujian Proses *Sampling*

Pengujian proses *sampling* dilakukan untuk mengetahui kesesuaian frekuensi *sampling* yang dilakukan oleh mikrokontroler dengan perancangan. Diagram alir pengujian proses *sampling* ditunjukkan dalam Gambar 3.11.



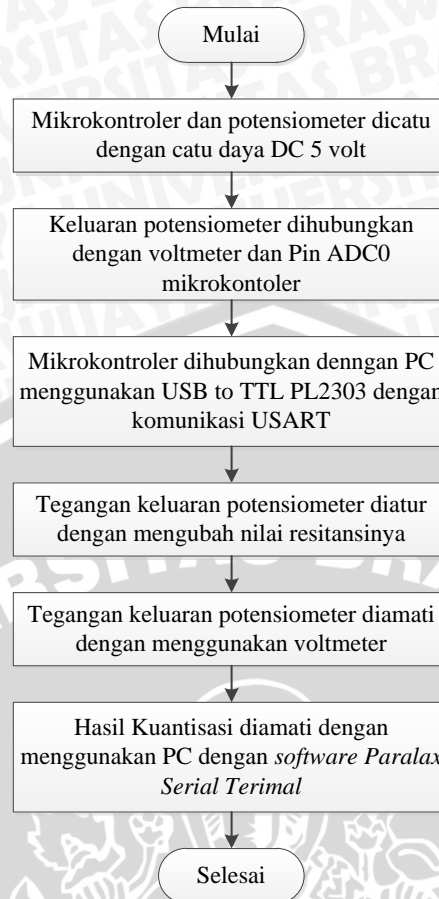
Gambar 3.11 Diagram Alir Pengujian Proses *Sampling*

Langkah-langkah yang dilakukan dalam melakukan pengujian proses *sampling* antara lain :

1. Mikrokontroler dicatu dengan catu daya DC 5 volt.
2. Diberikan perintah PORTB.7~PORTB.7 pada *timer2* mikrokontroler yang berfungsi untuk mengatur frekuensi *sampling*.
3. Keluaran mikrokontroler pada Pin B.7 dihubungkan dengan *oscilloscope*.
4. Hasil sinyal keluaran mikrokontroler pada Pin B.7 diamati melalui *oscilloscope*.

3.2.4 Pengujian Proses Kuantisasi

Pengujian proses kuantisasi dilakukan untuk mengetahui kesesuaian proses kuantisasi yang dilakukan oleh mikrokontroler dengan perancangan. Diagram alir pengujian proses kuantisasi ditunjukkan dalam Gambar 3.12.



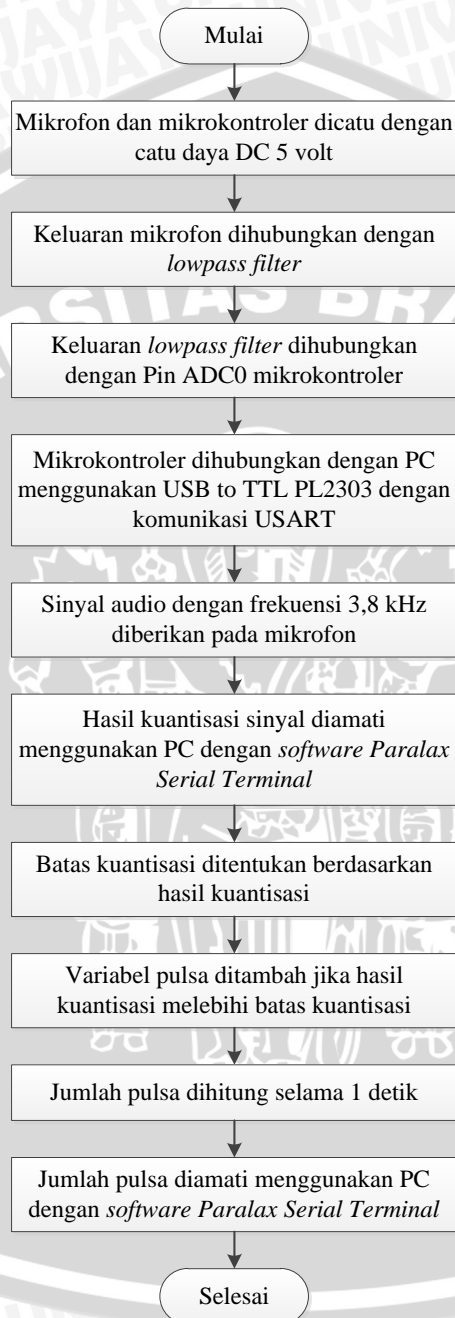
Gambar 3.12 Diagram Alir Pengujian Proses Kuantisasi

Langkah-langkah yang dilakukan dalam melakukan pengujian proses kuantisasi antara lain :

1. Mikrokontroler dan potensiometer dicatu dengan catu daya DC 5 volt.
2. Keluaran potensiometer dihubungkan dengan voltmeter dan Pin ADC0 mikrokontroler.
3. Mikrokontroler dihubungkan dengan PC menggunakan USB to TTL PL2303 dengan komunikasi USART.
4. Tegangan keluaran potensiometer diatur dengan mengubah nilai resistansinya sehingga memberikan tegangan keluaran berbeda-beda dengan besar 1 volt, 2 volt, 3 volt, 4 volt, dan 5 volt.
5. Tegangan keluaran potensiometer diamati dengan menggunakan voltmeter.
6. Hasil kuantisasi diamati dengan menggunakan PC dengan *software Parallax Serial Terminal*.

3.2.5 Pengujian Proses Coding

Pengujian proses *coding* dilakukan untuk mengetahui kesesuaian proses *coding* yang dilakukan oleh mikrokontroler dengan perancangan. Diagram alir pengujian proses *coding* ditunjukkan dalam Gambar 3.13.



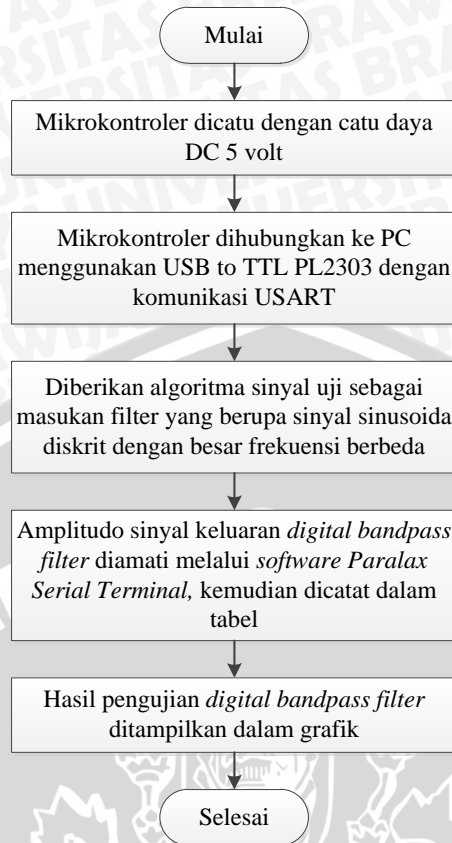
Gambar 3.13 Diagram Alir Pengujian Proses Coding

Langkah-langkah yang dilakukan dalam melakukan pengujian proses *coding* antara lain :

1. Mikrofon dan mikrokontroler dicatu dengan catu daya DC 5 volt.
2. Keluaran mikrofon dihubungkan dengan *lowpass filter*.
3. Keluaran *lowpass filter* dihubungkan dengan Pin ADC0 mikrokontroler.
4. Mikrokontroler dihubungkan dengan PC menggunakan USB to TTL PL2303 dengan komunikasi USART.
5. Sinyal audio dengan frekuensi 3,8 kHz diberikan pada mikrofon.
6. Hasil kuantisasi sinyal diamati menggunakan menggunakan PC dengan *software Parallax Serial Terminal*.
7. Batas kuantisasi ditentukan berdasarkan hasil kuantisasi.
8. Variabel pulsa ditambah jika hasil kuantisasi melebihi batas kuantisasi
9. Jumlah pulsa dihitung selama 1 detik.
10. Jumlah pulsa diamati menggunakan PC dengan *software Parallax Serial Terminal*.

3.2.6 Pengujian *Digital Bandpass Filter*

Pengujian ini dilakukan dengan cara melakukan proses komputasi algoritma *digital bandpass filter*. Hal ini bertujuan untuk mengetahui apakah filter mampu meloloskan sinyal yang memiliki besar frekuensi diantara frekuensi *cutoff* dan meredam sinyal yang memiliki besar frekuensi bukan diantara frekuensi *cutoff*. Diagram alir pengujian *digital bandpass filter* ditunjukkan dalam Gambar 3.14.



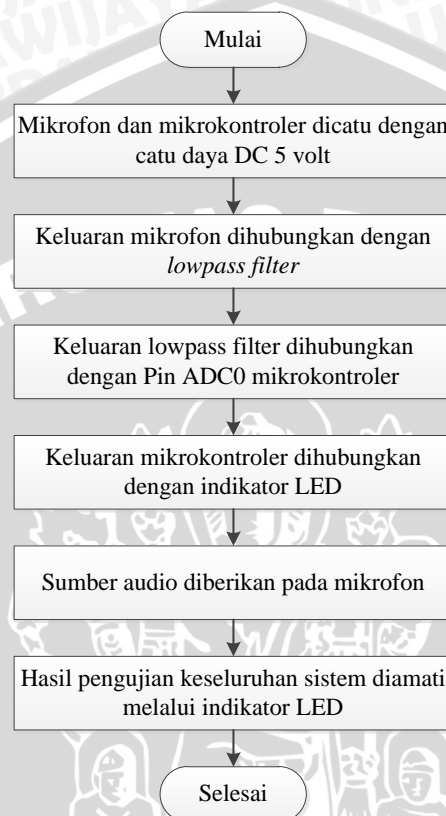
Gambar 3.14 Diagram Alir Pengujian *Digital Bandpass Filter*

Langkah-langkah yang dilakukan dalam melakukan pengujian *digital bandpass filter* antara lain :

1. Mikrokontroler dicatu dengan catu daya DC 5 volt.
2. Mikrokontroler dihubungkan dengan PC menggunakan USB to TTL PL2303 dengan komunikasi USART.
3. Diberikan algoritma sinyal uji sebagai masukan filter yang berupa sinyal sinusoida diskrit dengan besar frekuensi berbeda yaitu sebesar 500 Hz, 1000 Hz, 2000 Hz, 3000 Hz, 3500 Hz, 3600 Hz, 3700 Hz, 3800 Hz, 3900 Hz, 4000 Hz, 4100 Hz, 4500 Hz, 4200 Hz, 4400 Hz, 4600 Hz, 4800 Hz, dan 5000 Hz.
4. Amplitudo sinyal keluaran *digital bandpass filter* diamati melalui *software Parallax Serial Terminal*, kemudian dicatat dalam tabel.
5. Hasil pengujian *digital bandpass filter* ditampilkan dalam grafik.

3.2.7 Pengujian Keseluruhan Sistem

Pengujian keseluruhan sistem bertujuan untuk mengetahui kinerja sistem dalam mendeteksi sumber audio berupa sinyal suara dengan frekuensi 3,8 kHz dan mengabaikan sumber audio lainnya. Diagram alir pengujian keseluruhan sistem ditunjukkan dalam Gambar 3.15.



Gambar 3.15 Diagram Alir Pengujian Keseluruhan Sistem

Langkah-langkah yang dilakukan dalam melakukan pengujian keseluruhan sistem antara lain :

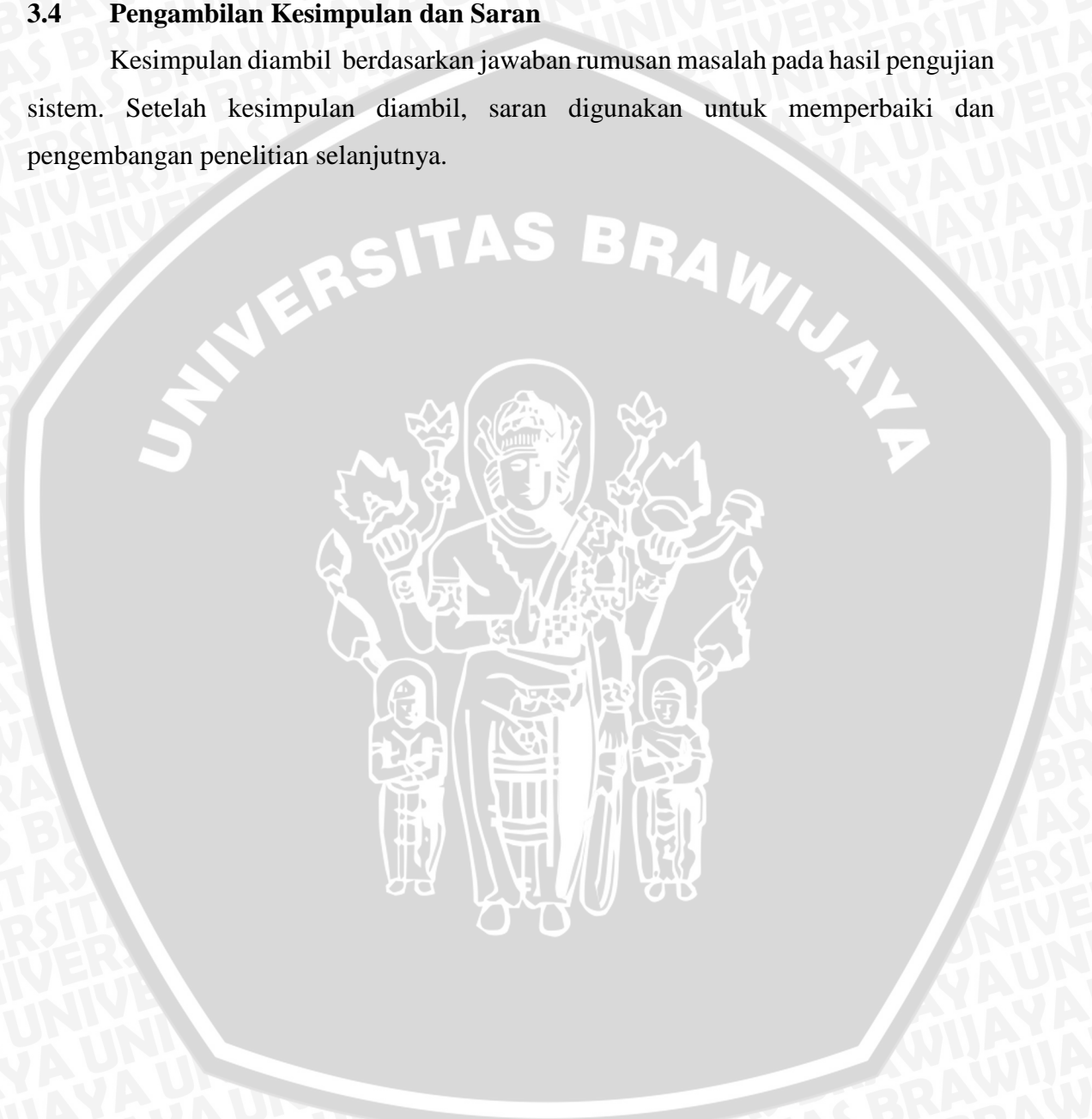
1. Mikrofon dan mikrokontroler dicatu dengan catu daya DC 5 volt.
2. Keluaran mikrofon dihubungkan dengan *lowpass filter*.
3. Keluaran *lowpass filter* dihubungkan dengan Pin ADC0 mikrokontroler.
4. Keluaran mikrokontroler dihubungkan dengan indikator LED.
5. Sumber audio yang terdiri dari sinyal suara dengan frekuensi 3,8 kHz, musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara diberikan pada mikrofon.
6. Hasil pengujian keseluruhan sistem diamati melalui indikator LED.

3.3 Pembahasan

Pembahasan bertujuan untuk melakukan analisis terkait kesesuaian sistem yang telah dibuat dengan sistem yang dirancang berdasarkan hasil dari pengujian sistem yang dibandingkan dengan perancangan sistem.

3.4 Pengambilan Kesimpulan dan Saran

Kesimpulan diambil berdasarkan jawaban rumusan masalah pada hasil pengujian sistem. Setelah kesimpulan diambil, saran digunakan untuk memperbaiki dan pengembangan penelitian selanjutnya.



BAB IV

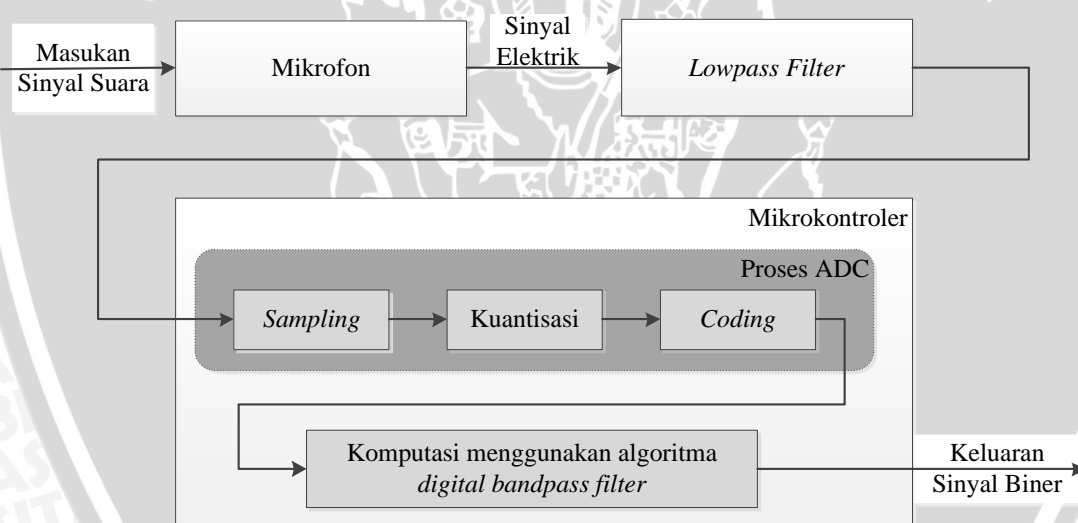
HASIL DAN PEMBAHASAN

4.1 Umum

Pembahasan yang dikaji yaitu mengenai perancangan sistem, implementasi perancangan, dan pengujian sistem. Sistem yang dibuat dalam penelitian ini adalah sistem *sound activation* yang mengaplikasikan desain *digital bandpass filter* yang akan diterapkan pada robot KRPAI.

4.2 Perancangan Sistem

Perancangan sistem merupakan perancangan alat secara umum yang diperlukan dalam sistem *sound activation*. Diagram blok sistem yang dirancang ditunjukkan dalam Gambar 4.1.



Gambar 4.1 Diagram Blok Sistem yang Dirancang

Sumber: Perancangan

4.2.1 Perancangan Mikrofon

Mikrofon yang digunakan adalah *electret condenser microphone* dan dihubungkan ke masukan sistem. Berdasarkan *datasheet electret condenser microphone*, besar tegangan keluaran mikrofon adalah 8 mV, agar tegangan keluaran mikrofon dapat mencapai 1 V dibutuhkan rangkaian penguat yang mampu menguatkan tegangan keluaran dengan besar penguatan 125 kali. Dalam rangkaian ini telah ditambahkan rangkaian penguat yang berupa rangkaian penguat *non-inverting* dengan nilai penguatan mendekati 125 yaitu sebesar 121 kali.

Untuk mendapatkan rangkaian penguat dengan nilai penguatan (*Gain*) sebesar 121 kali, dilakukan dua kali penguatan dengan masing-masing 11 kali penguatan yang dirangkai secara seri. Oleh karena itu, dibutuhkan nilai resistansi untuk nilai penguatan (*G*) tersebut, yaitu dengan menggunakan Persamaan (2-2).

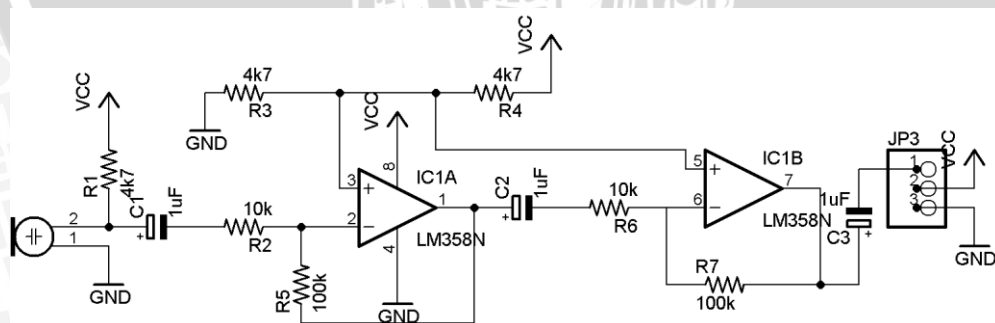
$$G = \left(1 + \frac{R_2}{R_1}\right)$$

$$11 = \left(1 + \frac{R_2}{R_1}\right)$$

$$10 = \frac{R_2}{R_1}$$

$$R_2 = 10R_1$$

Apabila digunakan nilai R_1 sebesar 10 k Ω , maka nilai R_2 sebesar 100 k Ω . Untuk rangkaian elektronik dari mikrofon ditunjukkan dalam Gambar 4.2.



Gambar 4.2 Rangkaian Elektronik dari Mikrofon

Sumber: Perancangan

4.2.2 Perancangan *Lowpass Filter*

Lowpass filter yang dirancang adalah sebuah filter yang dapat meredam sinyal dengan frekuensi diatas 5 kHz. Filter ini berfungsi sebagai filter *anti-aliasing* yang bertujuan agar tidak terjadi fenomena *aliasing* pada proses *sampling* yang menyebabkan informasi yang hilang semakin bnyak, karena frekuensi *sampling* yang digunakan adalah sebesar 10 kHz. Keluaran filter ini akan menjadi masukan proses *analog to digital converter*, untuk itu dibutuhkan nilai *magnitude* maksimal yang rata pada daerah *passband* untuk meningkatkan akurasi pembacaan. Jenis filter *butterworth* digunakan karena aplikasinya sebagai filter *anti-aliasing* dan respon frekuensinya yang memiliki nilai *magnitude* maksimal yang rata pada daerah *passband*. *Lowpass filter* yang dirancang merupakan filter pasif karena pada sinyal elektrik keluaran mikrofon sudah dilakukan penguatan sebelum menjadi masukan filter.

Untuk membuat *lowpass filter* dengan *frekuensi cutoff* (f_c) sebesar 5 kHz, nilai R dan C dapat ditentukan dengan Persamaan (2-6) , yaitu :

$$f_c = \frac{1}{2\pi RC}$$

$$RC = \frac{1}{2 \cdot 3,14 \cdot 5000}$$

$$RC = \frac{1}{31400}$$

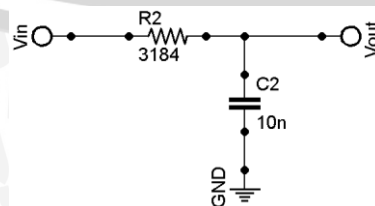
$$RC = 3,18471 \cdot 10^{-5}$$

Jika C = 10 nF, maka

$$R = \frac{3,18471 \cdot 10^{-5}}{10^{-8}}$$

$$R = 3184,71 \Omega$$

Sehingga rangkaian *lowpass filter* seperti pada Gambar 4.3.

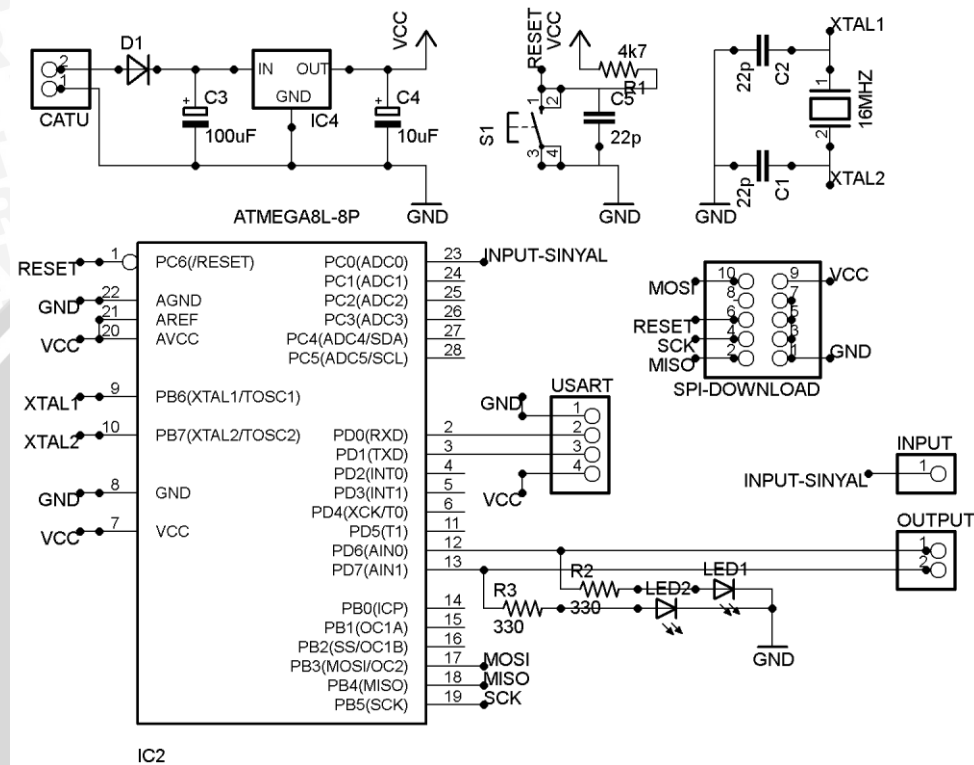


Gambar 4.3 Rangkaian *Lowpass Filter*

Sumber: Perancangan

4.2.3 Perancangan Mikrokontroler ATmega8

Sebuah mikrokontroler memerlukan rangkaian elektronik untuk mendukung kinerja dari mikrokontroler tersebut. Rangkaian elektronik dari mikrokontroler ditunjukkan pada Gambar 4.4.



Gambar 4.4 Rangkaian Elektronik Mikrokontroler ATmega8

Sumber: Perancangan

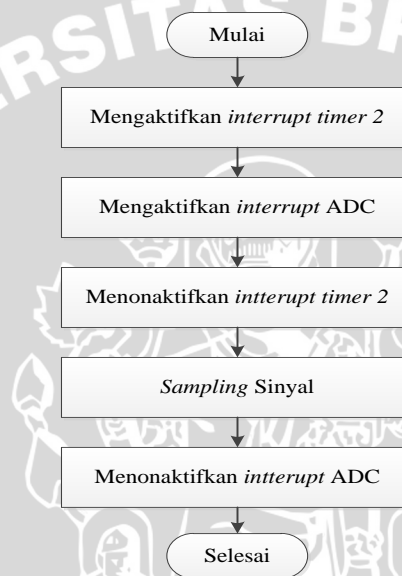
Rangkaian elektronik dari mikrokontroler dirancang sesuai dengan kebutuhan dari fungsi yang akan digunakan pada mikrokontroler. Dalam penelitian ini fungsi mikrokontroler yang digunakan mencakup *timer*, ADC, USART, I/O.

Beberapa Pin yang digunakan pada mikrokontroler antara lain :

1. Pin untuk sumber daya mikrokontroler yaitu Pin VCC dan Pin *ground* (GND).
2. Pin masukan mikrokontroler yaitu Pin C.0 (ADC 0) yang digunakan untuk *input* sistem mikrokontroler yang menerima sinyal keluaran dari *low pass filter*.
3. Pin AVCC sebagai referensi ADC.
4. Pin keluaran sistem yaitu Pin D.6 dan Pin D.7 sebagai penghasil sinyal biner sekaligus indikator LED hijau pada Pin D.6 dan LED merah pada Pin D.7.
5. Selain itu beberapa pin lain yang digunakan untuk kelengkapan pengujian yaitu Pin Tx dan Rx serta Pin antarmuka *downloader* dan pin RESET.

4.2.4 Perancangan Proses *Sampling*

Nilai frekuensi *sampling* yang digunakan dalam proses *sampling* ini adalah sebesar 10 kHz ($f_s = 10 \text{ kHz}$). Agar mikrokontroler mampu melakukan proses ini, periode *sampling* ADC mikrokontroler harus diatur dengan menggunakan *timer2* yang akan mengaktifkan *sampling* ADC dengan periode 0.1 ms ($T = 1/f_s$). Pengaktifan *interrupt routine sampling* ADC ATmega8 dilakukan dengan memberikan nilai 1 pada bit ADCSRA.7 kemudian untuk memulai proses *sampling* dengan ADC yaitu dengan memberikan nilai 1 pada bit ADCSRA.6. Algoritma proses *sampling* dapat dilihat pada Gambar 4.5.

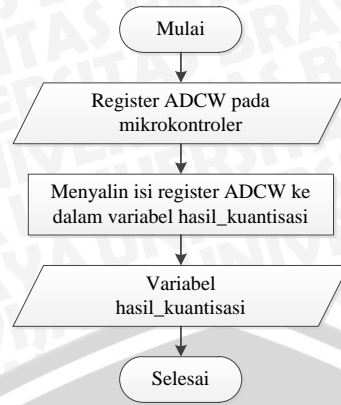


Gambar 4.5 Diagram Alir Proses *Sampling*

Sumber: Perancangan

4.2.5 Perancangan Proses Kuantisasi

Pada penelitian ini jumlah bit yang digunakan untuk proses kuantisasi adalah 8 bit. Kuantisasi akan dimulai dengan memberikan nilai 0 pada bit ke-5 dari register ADMUX. Hasil keluaran dari kuantisasi ini akan sesuai dengan Persamaan (2-3) dan disimpan pada register ADCW. Untuk menggunakan hasil kuantisasi cukup salin nilai dari register ADCW ke dalam variabel yang diinginkan. Algoritma proses kuantisasi ditunjukkan dalam Gambar 4.6.

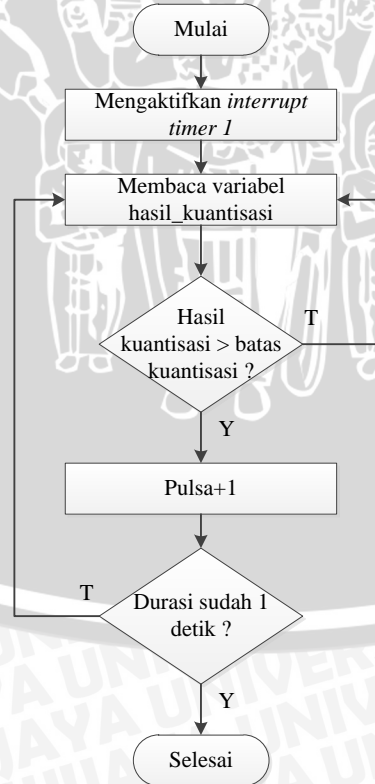


Gambar 4.6 Diagram Alir Proses Kuantisasi

Sumber: Perancangan

4.2.6 Perancangan Proses *Coding*

Batas nilai kuantisasi yang digunakan dalam proses *coding* ditentukan berdasarkan hasil kuantisasi dari sinyal suara yang diberikan pada mikrofon. Nilai pulsa akan meningkat jika hasil kuantisasi melebihi batas kuantisasi. Dalam proses *coding*, dilakukan perhitungan jumlah pulsa selama satu detik. Algoritma proses *coding* ditunjukkan dalam Gambar 4.7.

Gambar 4.7 Diagram Alir Proses *Coding*

Sumber: Perancangan

4.2.7 Perancangan *Digital Bandpass Filter*

Dalam proses ini dilakukan proses komputasi sinyal diskrit menggunakan algoritma *digital bandpass filter*. Filter dirancang untuk mampu meloloskan sinyal dengan frekuensi 3,8 kHz dan meredam sinyal pada daerah *stopband* dimana besar amplitudo keluaran sinyal akan diredam hingga mencapai 1/10 dari besar amplitudo masukan sinyal, sehingga nilai redaman filter dalam satuan dB adalah sebesar 20 dB.

Spesifikasi *digital bandpass filter* yang dirancang yaitu :

1. frekuensi *passband* bawah (f_l) sebesar 3,7 kHz,
2. frekuensi *passband* atas (f_u) sebesar 3,9 kHz,
3. frekuensi *stopband* bawah (f_1) sebesar 3,6 kHz,
4. frekuensi *stopband* atas (f_2) sebesar 4 kHz,
5. *passband ripple* (K_1) sebesar 3 dB,
6. *stopband attenuation* (K_2) sebesar 20 dB.

Dengan spesifikasi diatas selanjutnya dilakukan langkah menghitung orde filter kemudian menghitung fungsi alih analognya kemudian dilakukan transformasi bilinear untuk menghitung fungsi alih digital. Setelah didapat fungsi alih digital, selanjutnya menghitung persamaan beda sebagai dasar untuk membuat algoritma *digital bandpass filter* untuk diterapkan pada mikrokontroler. Untuk mendapatkan orde filter dapat menggunakan Persamaan (2-7), yaitu :

$$K_1 = -3 \text{ dB}$$

$$f_l = 3700 \text{ Hz}$$

$$f_1 = 3600 \text{ Hz}$$

$$K_2 = -20 \text{ dB}$$

$$f_u = 3900 \text{ Hz}$$

$$f_2 = 4000 \text{ Hz}$$

Passband

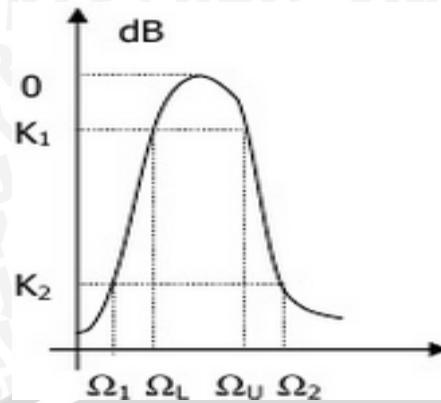
$$\Omega_l = 2\pi f_l = 2 * \pi * 3700 = 7400\pi = 23.247,785 \text{ rad/s}$$

$$\Omega_u = 2\pi f_u = 2 * \pi * 3900 = 7800\pi = 24.504,422 \text{ rad/s}$$

Stopband

$$\Omega_1 = 2\pi f_1 = 2 * \pi * 3600 = 7200\pi = 22.619,467 \text{ rad/s}$$

$$\Omega_2 = 2\pi f_2 = 2 * \pi * 4000 = 8000\pi = 25.132,741 \text{ rad/s}$$



Gambar 4.8 Spesifikasi *Digital Bandpass Filter*

Sumber: Perancangan

Dalam metode transformasi bilinear, ditentukan frekuensi *prewarping* terlebih dahulu menggunakan Persamaan (2-11).

Passband

$$W_l = \Omega_l T = 23.247,785 / 10000 = 2,3247 \text{ rad}$$

$$W_u = \Omega_u T = 24.504,422 / 10000 = 2,4504 \text{ rad}$$

$$\Omega'_l = 2/T * \tan(W_l/2) = 20000 \tan(2,3247/2) = 405,7923 \text{ rad/s}$$

$$\Omega'_u = 2/T * \tan(W_u/2) = 20000 \tan(2,4504/2) = 427,7406 \text{ rad/s}$$

Stopband

$$W_1 = \Omega_1 T = 22.619,467 / 10000 = 2,2619 \text{ rad}$$

$$W_2 = \Omega_2 T = 25.132,741 / 10000 = 2,5132 \text{ rad}$$

$$\Omega'_1 = 2/T * \tan(W_1/2) = 20000 \tan(2,2619/2) = 394,8273 \text{ rad/s}$$

$$\Omega'_2 = 2/T * \tan(W_2/2) = 20000 \tan(2,5132/2) = 438,7064 \text{ rad/s}$$

Lowpass Prototype :

$$A = \frac{-(\Omega'_1)^2 + \Omega'_l \Omega'_u}{\Omega'_1 (\Omega'_u - \Omega'_l)}$$

$$A = \frac{-(394,8273)^2 + (405,7923) \cdot (427,7406)}{394,8273 (405,7923 - 427,7406)}$$

$$A = 2.040812$$

$$B = \frac{(\Omega_u)^2 - \Omega_l \Omega_u}{\Omega_u (\Omega_u - \Omega_l)}$$

$$B = \frac{(438,7064)^2 - (405,7923) \cdot (427,7406)}{438,7064 (405,7923 - 427,7406)}$$

$$B = 1.961755$$

$$\Omega_r = \min(A, B)$$

$$\Omega_r = 1.961755$$

$$n = \frac{\log \left(\frac{10^{-K_1/10} - 1}{10^{-K_2/10} - 1} \right)}{2 \log \frac{1}{\Omega_r}}$$

$$n = \frac{\log \left(\frac{10^{0.3} - 1}{10^2 - 1} \right)}{2 \log \frac{1}{1,961755}}$$

$$n = 3.4111$$

$$n \cong 4$$

Dari Tabel 2.2 polinomial *butterworth* untuk $n = 4$

$$H_{LPF4}(s) = \frac{1}{s^4 + 2,61s^3 + 3,41s^2 + 2,61s + 1}$$

Transformasi *analog transfer LPF to BPF* :

$$s \rightarrow \frac{s^2 + \Omega_l \Omega_u}{s(\Omega_u - \Omega_l)}$$

$$\text{maka : } s \rightarrow \frac{s^2 + (405,7923) \cdot (427,7406)}{s (405,7923 - 427,7406)}$$

$$H_{BPF}(s) = \frac{1}{\left[\frac{s^2 + (405,7923) \cdot (427,7406)}{s (405,7923 - 427,7406)} \right]^4 + 2,61 \left[\frac{s^2 + (405,7923) \cdot (427,7406)}{s (405,7923 - 427,7406)} \right]^3 + 3,41 \left[\frac{s^2 + (405,7923) \cdot (427,7406)}{s (405,7923 - 427,7406)} \right]^2 + 2,61 \left[\frac{s^2 + (405,7923) \cdot (427,7406)}{s (405,7923 - 427,7406)} \right] + 1}$$

Transformasi Bilinear :

$$H(z) = H(s) \Big|_s = \frac{2(1-z^{-1})}{T_s(1-z^{-1})} = 20000 \frac{(1-z^{-1})}{(1-z^{-1})}$$

$$H_{BPF}(z) = \frac{1}{\left[\frac{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right)^2 + (405,7923) \cdot (427,7406)}{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right) (405,7923 - 427,7406)} \right]^4 + \left[\frac{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right)^2 + (405,7923) \cdot (427,7406)}{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right) (405,7923 - 427,7406)} \right]^3 + \left[\frac{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right)^2 + (405,7923) \cdot (427,7406)}{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right) (405,7923 - 427,7406)} \right]^2 + \left[\frac{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right)^2 + (405,7923) \cdot (427,7406)}{\left(20000 \frac{(1-z^{-1})}{(1-z^{-1})}\right) (405,7923 - 427,7406)} \right] + 1}$$

$$H_{BPF}(z) = \frac{0,0041 - 0,0083z^{-2} + 0,0041z^{-4}}{1 + 2,7818z^{-1} + 3,7473z^{-2} + 2,5267z^{-3} + 0,8253z^{-4}}$$

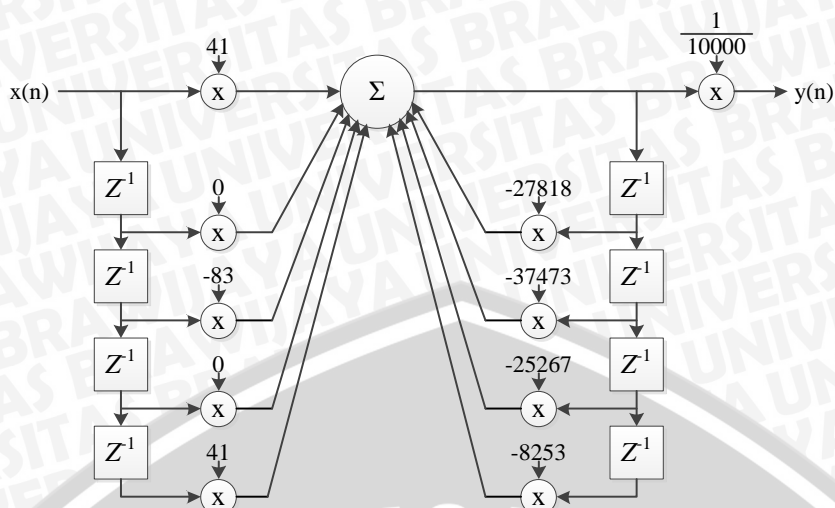
$$\frac{Y(z)}{X(z)} = \frac{0,0041 - 0,0083z^{-2} + 0,0041z^{-4}}{1 + 2,7818z^{-1} + 3,7473z^{-2} + 2,5267z^{-3} + 0,8253z^{-4}}$$

$$Y(z) = 0,0041X(z) - 0,0083z^{-2}X(z) + 0,0041z^{-4}X(z) - 2,7818z^{-1}Y(z) - 3,7473z^{-2}Y(z) - 2,5267z^{-3}Y(z) - 0,8253z^{-4}Y(z)$$

Dilakukan invers transformasi Z untuk memperoleh persamaan beda $y(n)$:

$$y(n) = 0,0041x(n) - 0,0083x(n-2) + 0,0041x(n-4) - 2,7818y(n-1) - 3,7473y(n-2) - 2,5267y(n-3) - 0,8253y(n-4)$$

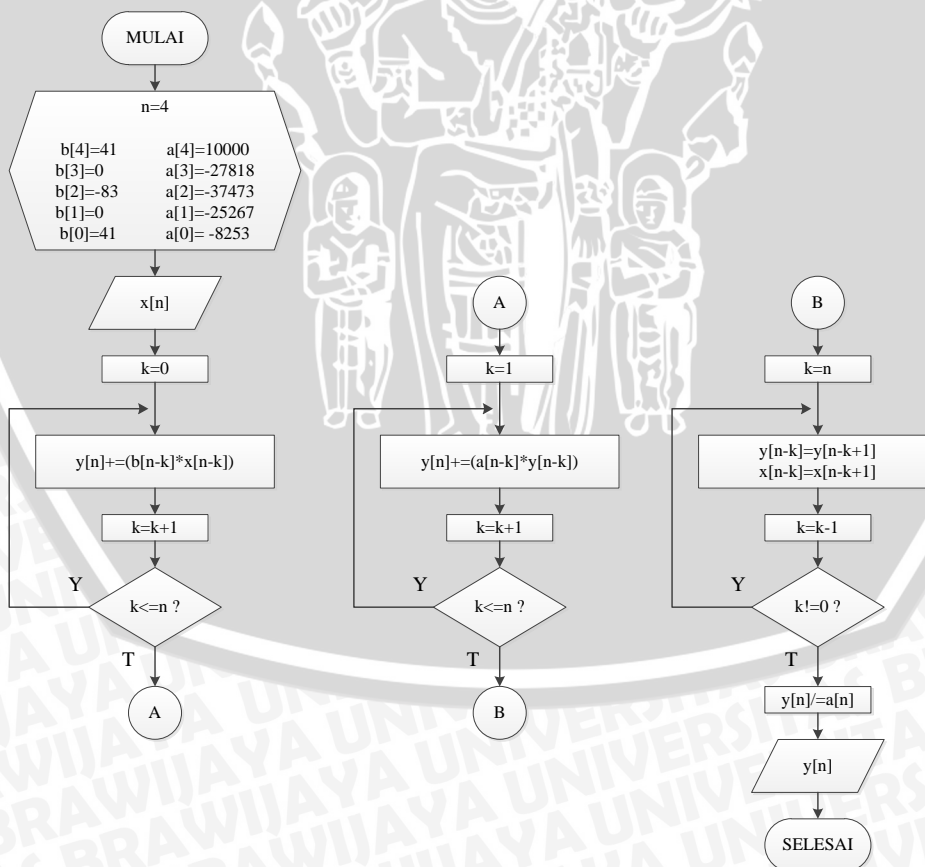
$$y(n) = \frac{1}{10000} [41x(n) - 83x(n-2) + 41x(n-4) - 27818y(n-1) - 37473y(n-2) - 25267y(n-3) - 8253y(n-4)]$$



Gambar 4.9 Struktur Persamaan Beda

Sumber: Perancangan

Persamaan beda diubah menjadi algoritma *digital bandpass filter* untuk diterapkan pada mikrokontroler. Diagram alir algoritma *digital bandpass filter* ditunjukkan dalam Gambar 4.10.



Gambar 4.10 Algoritma *Digital Bandpass Filter*

Sumber: Perancangan

Langkah-langkah yang dilakukan dalam algoritma tersebut adalah sebagai berikut :

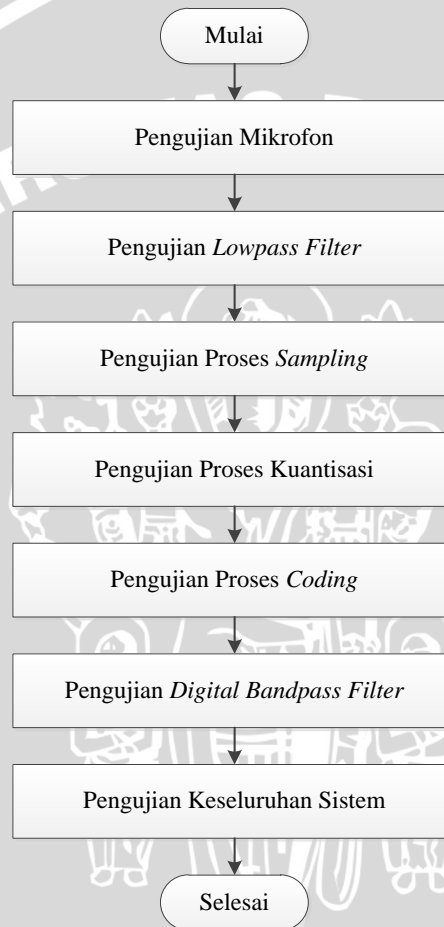
1. Nilai variabel n diisi dengan nilai orde filter yaitu 4.
2. Nilai koefisien filter diisi dalam variabel *array* $b[4]$, $b[3]$, $b[2]$, $b[1]$, $b[0]$ untuk koefisien x , dan $a[4]$, $a[3]$, $a[2]$, $a[1]$, $a[0]$ untuk koefisien y .
3. diberikan masukan variabel $x[n]$.
4. Variabel k diisi dengan nilai 0.
5. Variabel $y[n]$ ditambah dengan hasil perkalian variabel $b[n-k]$ dengan $x[n-k]$.
6. Variabel k ditambah dengan 1
7. Variabel k dibandingkan dengan variabel n , apabila nilai k kurang dari sama dengan n , diulangi langkah 5, apabila tidak, dilakukan langkah selanjutnya.
8. Variabel k diisi dengan nilai 1.
9. Variabel $y[n]$ ditambah dengan hasil perkalian variabel $b[n-k]$ dengan $y[n-k]$.
10. Variabel k ditambah dengan 1
11. Variabel k dibandingkan dengan variabel n , apabila nilai k kurang dari sama dengan n , diulangi langkah 9, apabila tidak, dilakukan langkah selanjutnya.
12. Variabel k diisi dengan nilai variabel n .
13. Variabel $y[n-k]$ diisi dengan nilai variabel $y[n-k+1]$ dan variabel $x[n-k]$ diisi dengan nilai variabel $x[n-k+1]$.
14. Variabel k dikurangi dengan 1
15. Variabel k dibandingkan dengan nilai 0, apabila nilai k tidak sama dengan 0, diulangi langkah 13, apabila tidak, dilakukan langkah selanjutnya.
16. Variabel $y[n]$ dibagi dengan nilai variabel $a[n]$.
17. Variabel $y[n]$ ditampilkan sebagai hasil keluaran *digital bandpass filter*.

4.3 Implementasi Perancangan

Proses implementasi perancangan meliputi pencarian komponen, dan perangkaian komponen sehingga terbentuk sistem perangkat keras, kemudian dilakukan penerapan perangkat lunak pada perangkat keras yang telah dibuat. Alat diimplementasikan menjadi suatu *prototype* berupa rangkaian board sistem *sound activation* pada robot KRPAI.

4.4 Pengujian Sistem

Pengujian sistem dibagi menjadi tujuh pengujian, antara lain pengujian mikrofon, pengujian *lowpass filter*, pengujian proses *sampling*, pengujian proses kuantisasi, pengujian proses *coding*, pengujian *digital bandpass filter*, serta pengujian keseluruhan sistem. Hal ini dilakukan bertujuan untuk mengetahui kesesuaian sistem yang dibuat dengan perancangan, serta mengetahui masing-masing sub-sistem bekerja dengan baik. Diagram alir pengujian sistem ditunjukkan dalam Gambar 4.11.



Gambar 4.11 Diagram Alir Pengujian Sistem

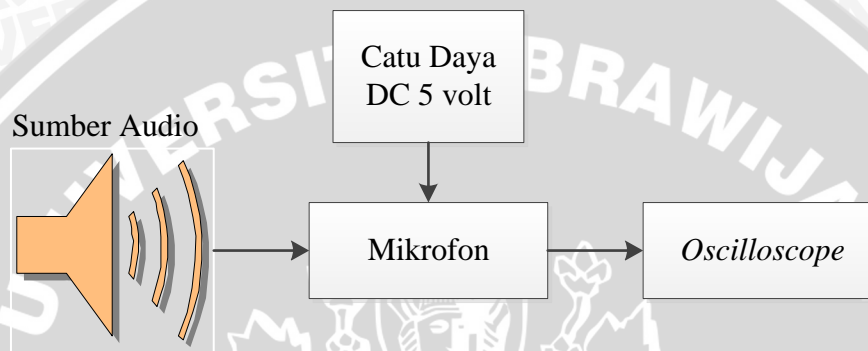
Sumber: Pengujian

4.4.1 Pengujian Mikrofon

- **Tujuan pengujian**

Pengujian mikrofon bertujuan untuk mengetahui kinerja mikrofon dalam mengubah sinyal suara menjadi sinyal elektrik.

- **Alat-alat yang digunakan**
 1. Catu daya DC 5 volt.
 2. *Microphone electret condenser* ABM-713-RC.
 3. *Oscilloscope* PC LAB 2000 SE.
 4. *Software driver* PC LAB 2000 SE
 5. Sumber audio berupa sinyal suara dengan frekuensi 3,8 kHz.
 6. Kabel konektor
- **Diagram blok pengujian**

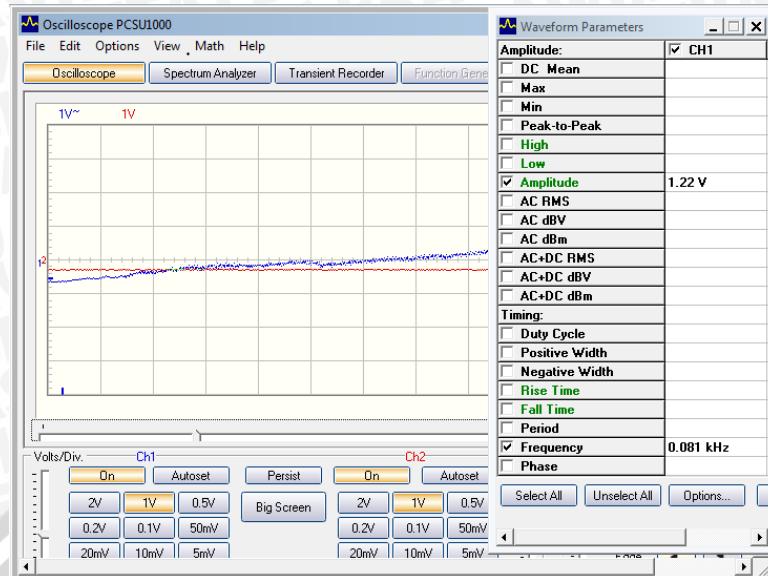


Gambar 4.12 Diagram Blok Pengujian Mikrofon

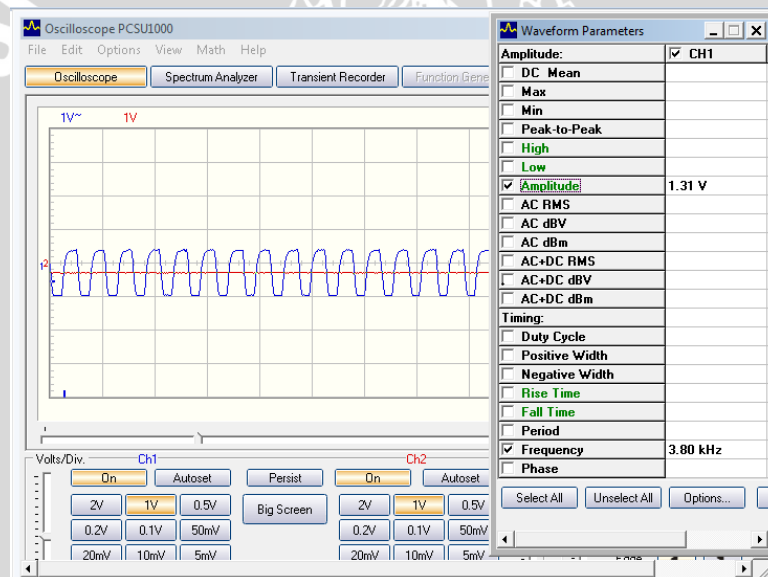
Sumber: Pengujian

- **Langkah-langkah pengujian**
 Pengujian mikrofon dilakukan dengan langkah-langkah sebagai berikut :
 1. Mikrofon dicatu dengan catu daya DC 5 volt.
 2. Keluaran mikrofon dihubungkan dengan *oscilloscope*.
 3. Sumber audio dengan frekuensi 3,8 kHz diberikan pada mikrofon.
 4. Hasil pengujian diketahui melalui *oscilloscope* pada keluaran mikrofon.
- **Hasil Pengujian**

Dalam Gambar 4.13 ditunjukkan sinyal keluaran mikrofon ketika tidak ada sumber audio, sedangkan dalam Gambar 4.14 ditunjukkan sinyal keluaran mikrofon ketika diberikan sumber audio 3,8 kHz.



Gambar 4.13 Sinyal Keluaran Mikrofon Tanpa Sumber Audio
Sumber: Hasil Pengujian



Gambar 4.14 Sinyal Keluaran Mikrofon dengan Sumber Audio 3,8 kHz
Sumber: Hasil Pengujian

- **Analisis hasil pengujian**

Hasil pengujian yang didapatkan mengindikasikan bahwa mikrofon mampu menghasilkan sinyal elektrik dari sinyal suara yang diberikan dan sinyal elektrik yang dihasilkan memiliki nilai frekuensi 3,8 kHz, sebanding dengan nilai frekuensi suara yang diberikan yaitu sebesar 3,8 kHz.

4.4.2 Pengujian *Lowpass Filter*

- **Tujuan pengujian**

Pengujian *lowpass filter* bertujuan untuk mengetahui kinerja *lowpass filter* yang dibuat dalam meredam sinyal dengan frekuensi diatas 5 kHz.

- **Alat-alat yang digunakan**

1. Rangkaian *lowpass filter*.
2. Catu daya AC 220 volt.
3. *Function generator* SO5127-2R
4. *Oscilloscope* Tetronix TDS 1012C-SC (3% error)
5. Kabel konektor

- **Diagram blok pengujian**



Gambar 4.15 Diagram Blok Pengujian *Lowpass Filter*

Sumber: Pengujian

- **Langkah-langkah pengujian**

Pengujian *lowpass filter* dilakukan dengan langkah-langkah sebagai berikut :

1. *Function generator* dicatu dengan catu daya AC 220 volt.
2. Masukan *lowpass filter* dihubungkan dengan *function generator*.
3. Keluaran *lowpass filter* dihubungkan dengan *oscilloscope*.
4. Diberikan sinyal pada masukan *lowpass filter* dengan tegangan 5 volt dan frekuensi yang berbeda menggunakan *function generator* yaitu 100 Hz, 500 Hz, 1000 Hz, 1500 Hz, 2000 Hz, 2500 Hz, 3000 Hz, 3500 Hz, 4000 Hz, 4500 Hz, 5000 Hz, 5500 Hz, 6000 Hz, 6500 Hz, 7000 Hz, 7500 Hz, dan 8000 Hz, 8500 Hz, 9000 Hz, 9500 Hz, dan 10000 Hz.
5. Nilai tegangan keluaran dicatat dalam tabel.
6. Hasil pengujian dibandingkan dengan hasil perancangan secara teori.

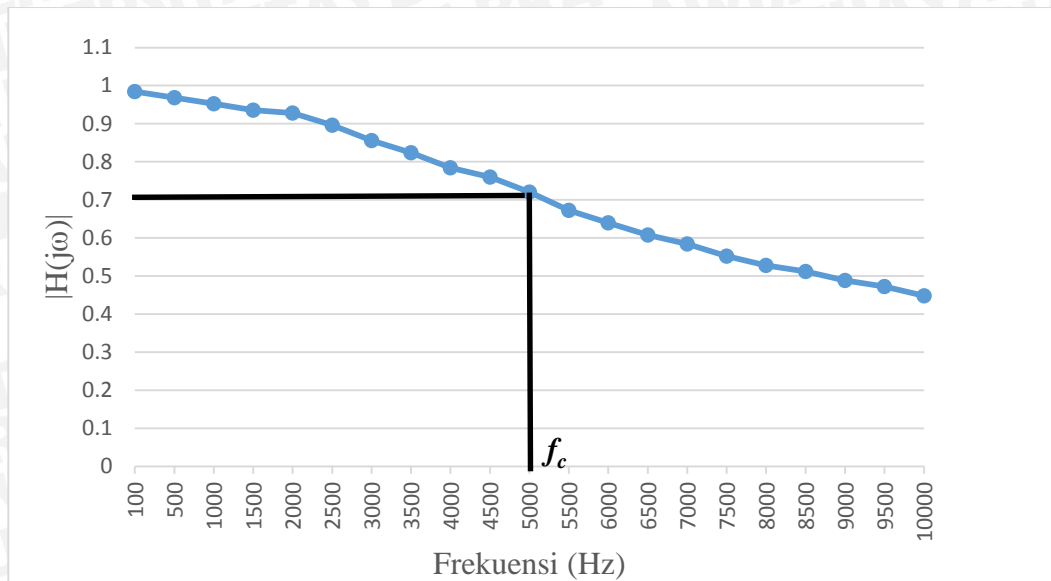
- Hasil pengujian

Tabel hasil pengujian *lowpass filter* ditunjukkan dalam Tabel 4.1. Grafik hasil pengujian *lowpass filter* ditunjukkan dalam Gambar 4.16 dan Gambar 4.17.

Tabel 4.1 Hasil Pengujian *Lowpass Filter*

| Tegangan masukan (volt) | Frekuensi Sinyal Masukan (Hz) | Tegangan Keluaran Perancangan (volt) | $ H(j\omega) $ Perancangan | Tegangan Keluaran Pengujian (volt) | $ H(j\omega) $ Pengujian | Error $ H(j\omega) $ (%) |
|--|-------------------------------|--------------------------------------|----------------------------|------------------------------------|--------------------------|--------------------------|
| 5 | 100 | 4,99 | 0,999 | 4,92 | 0,984 | 1,605 |
| 5 | 500 | 4,97 | 0,995 | 4,84 | 0,968 | 2,792 |
| 5 | 1000 | 4,9 | 0,98 | 4,76 | 0,952 | 3,001 |
| 5 | 1500 | 4,78 | 0,957 | 4,68 | 0,936 | 2,329 |
| 5 | 2000 | 4,64 | 0,928 | 4,64 | 0,928 | 0,047 |
| 5 | 2500 | 4,47 | 0,894 | 4,48 | 0,896 | 0,181 |
| 5 | 3000 | 4,28 | 0,857 | 4,28 | 0,856 | 0,166 |
| 5 | 3500 | 4,09 | 0,819 | 4,12 | 0,824 | 0,587 |
| 5 | 4000 | 3,9 | 0,78 | 3,92 | 0,784 | 0,41 |
| 5 | 4500 | 3,71 | 0,743 | 3,8 | 0,76 | 2,21 |
| 5 | 5000 | 3,53 | 0,707 | 3,60 | 0,72 | 1,805 |
| 5 | 5500 | 3,36 | 0,672 | 3,36 | 0,672 | 0,084 |
| 5 | 6000 | 3,2 | 0,64 | 3,20 | 0,64 | 0,012 |
| 5 | 6500 | 3,04 | 0,609 | 3,04 | 0,608 | 0,263 |
| 5 | 7000 | 2,9 | 0,581 | 2,92 | 0,584 | 0,491 |
| 5 | 7500 | 2,77 | 0,554 | 2,76 | 0,552 | 0,469 |
| 5 | 8000 | 2,64 | 0,529 | 2,64 | 0,528 | 0,358 |
| 5 | 8500 | 2,53 | 0,506 | 2,56 | 0,512 | 0,993 |
| 5 | 9000 | 2,42 | 0,485 | 2,44 | 0,488 | 0,505 |
| 5 | 9500 | 2,32 | 0,465 | 2,36 | 0,472 | 1,347 |
| 5 | 10000 | 2,23 | 0,447 | 2,24 | 0,448 | 0,198 |
| Error $H(j\omega)$ rata-rata (%) | | | | | | 0,946 |

Sumber: Hasil Pengujian



Gambar 4.16 Grafik Respon Frekuensi pada *Lowpass Filter*

Sumber: Hasil Pengujian

- **Analisis hasil pengujian**

Berdasarkan Tabel 4.1, dapat disimpulkan *lowpass filter* mampu meloloskan sinyal dengan frekuensi dibawah 5 kHz dan meredam sinyal dengan frekuensi diatas 5 kHz, ditunjukkan dengan nilai tegangan keluaran sebesar 3,6 volt dan nilai $|H(j\omega)|$ sebesar 0,72 pada frekuensi 5 kHz. Bila dibandingkan dengan tegangan keluaran perancangan sebesar 3,53 volt dan nilai $|H(j\omega)|$ perancangan sebesar 0,707, nilai *error* $|H(j\omega)|$ didapatkan = $\left| \frac{0,72-0,707}{0,72} \right| \times 100\% = 1,805\%$.

Dalam Gambar 4.16 ditunjukkan grafik respon $|H(j\omega)|$ terhadap frekuensi, dimana semakin besar frekuensi, nilai $|H(j\omega)|$ akan semakin kecil, hal ini menunjukkan bahwa filter yang dirancang mampu meredam sinyal elektrik frekuensi tinggi yaitu diatas frekuensi 5 kHz. Penurunan nilai $|H(j\omega)|$ disebabkan karena pada frekuensi tinggi nilai reaktansi kapasitor menjadi rendah yang memberikan efek kondisi hubung singkat pada terminal keluaran hingga menghasilkan keluaran mencapai nol.

4.4.3 Pengujian Proses *Sampling*

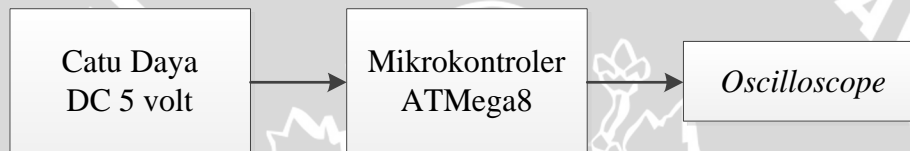
- **Tujuan pengujian**

Pengujian proses *sampling* bertujuan untuk mengetahui kesesuaian frekuensi *sampling* yang diatur pada mikrokontroler dengan perancangan.

- **Alat-alat yang digunakan**

1. Catu daya DC 5 volt
2. Mikrokontroler ATmega8
3. *Oscilloscope* PC LAB 2000 SE
4. *Software* PC LAB 2000 SE
5. Kabel konektor

- **Diagram blok pengujian**



Gambar 4.17 Diagram Blok Pengujian Proses *Sampling*

Sumber: Pengujian

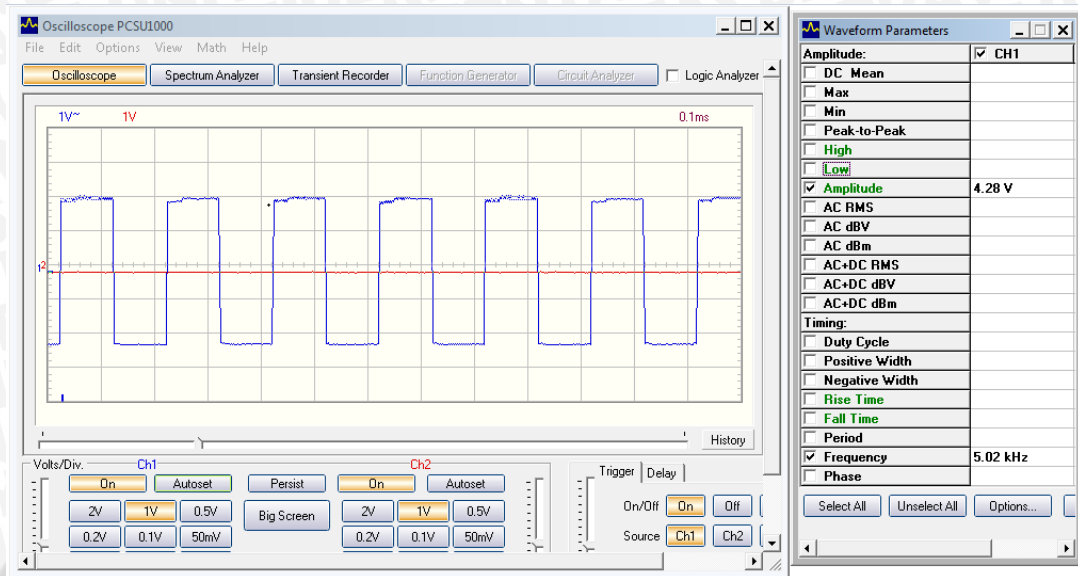
- **Langkah-langkah pengujian**

Pengujian proses *sampling* dilakukan dengan langkah-langkah sebagai berikut :

1. Mikrokontroler dicatu dengan catu daya DC 5 volt.
2. Diberikan perintah PORTB.7~PORTB.7 pada *timer2* mikrokontroler ATmega8 yang berfungsi untuk mengatur frekuensi *sampling*.
3. Keluaran mikrokontroler pada Pin B.7 dihubungkan dengan *oscilloscope*.
4. Hasil sinyal keluaran mikrokontroler ATmega8 pada Pin B.7 diamati melalui *oscilloscope*.

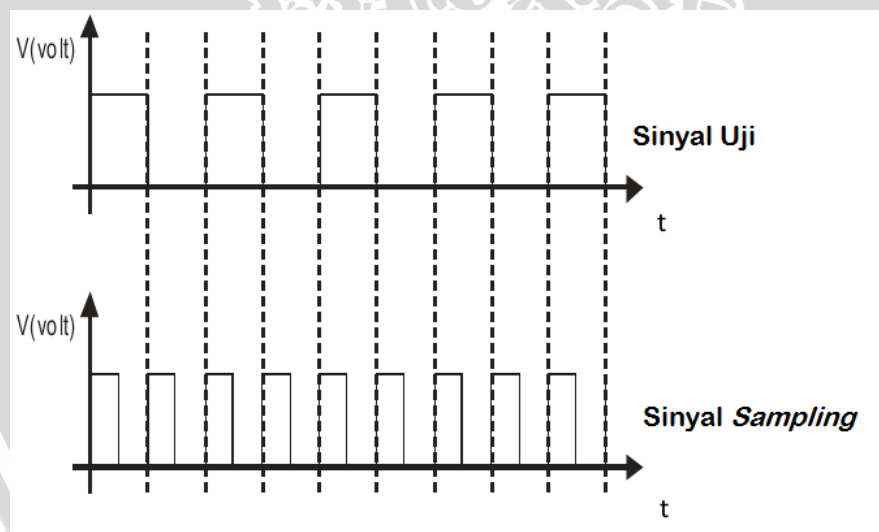
- Hasil pengujian

Hasil pengujian proses *sampling* ditunjukkan pada Gambar 4.18.



Gambar 4.18 Hasil Sinyal Proses *Sampling*

Sumber: Hasil Pengujian



Gambar 4.19 Perbandingan Sinyal Uji dengan Sinyal *Sampling*

Sumber: Hasil Pengujian

- **Analisis hasil pengujian**

Program uji berisi interupsi yang membuat sinyal keluaran pada Pin B.7 berubah setiap kali periode *sampling*. Oleh karena itu periode sinyal keluaran Pin B.7 adalah dua kali periode sinyal *sampling* dan frekuensinya adalah setengah dari frekuensi *sampling* seperti yang ditunjukkan dalam Gambar 4.19.

Berdasarkan hasil pengujian yang didapatkan, frekuensi keluaran Pin B.7 adalah 5,02 kHz sehingga frekuensi *sampling* adalah $5,02 \text{ kHz} \times 2 = 10,04 \text{ kHz}$.

4.4.4 Pengujian Proses Kuantisasi

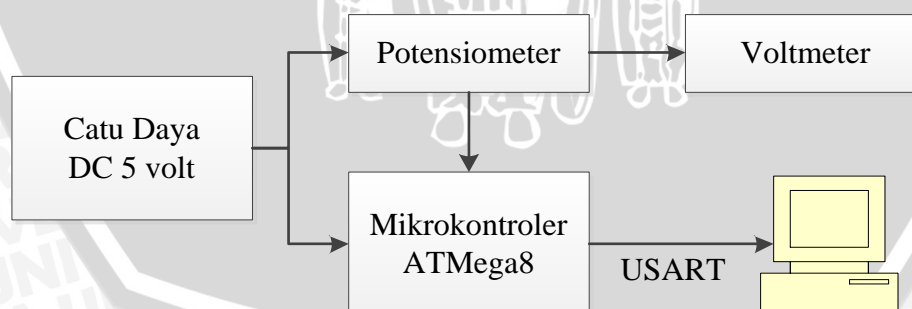
- **Tujuan pengujian**

Pengujian proses kuantisasi bertujuan untuk mengetahui kesesuaian hasil kuantisasi pada mikrokontroler sesuai dengan perancangan.

- **Alat-alat yang digunakan**

1. Mikrokontroler ATmega 8
2. Catu daya DC 5 volt
3. Potensiometer
4. PC
5. *Software Parallax Serial Terminal*
6. USB to TTL PL2303
7. Kabel konektor

- **Diagram blok pengujian**



Gambar 4.20 Diagram Blok Pengujian Proses Kuantisasi

Sumber: Pengujian

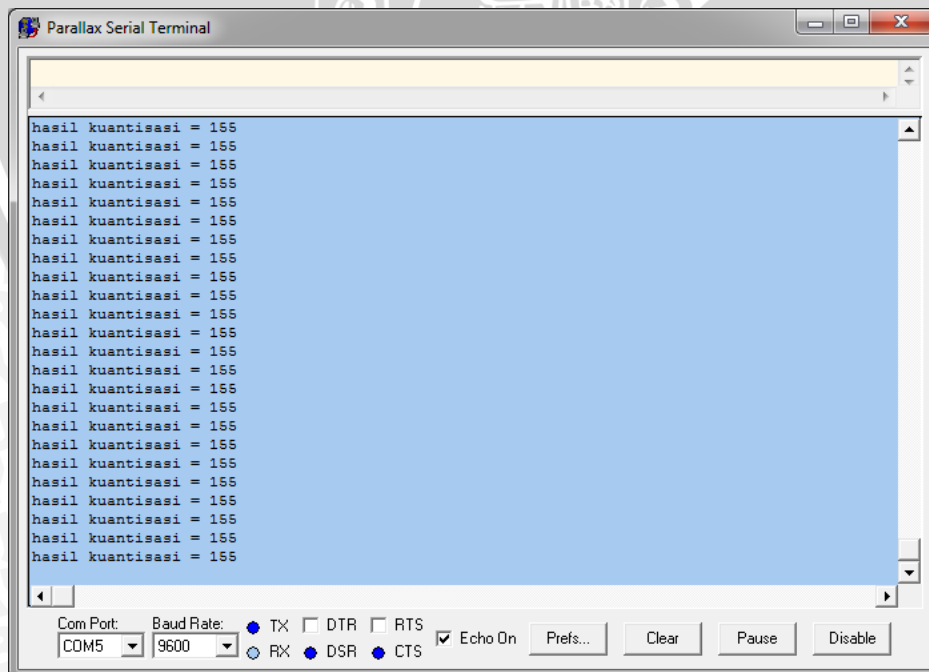
- **Langkah-langkah pengujian**

Pengujian proses kuantisasi dilakukan dalam langkah-langkah sebagai berikut:

1. Mikrokontroler dan potensiometer dicatu dengan catu daya DC 5 volt.
2. Keluaran potensiometer dihubungkan dengan voltmeter dan Pin ADC0 mikrokontroler ATmega8
3. Mikrokontroler ATmega8 dihubungkan dengan PC menggunakan USB to TTL PL2303 dengan komunikasi USART.
4. Tegangan keluaran potensiometer diatur dengan mengubah nilai resistansinya sehingga memberikan tegangan keluaran berbeda-beda dengan besar 1 volt, 2 volt, 3 volt, 4 volt, dan 5 volt.
5. Tegangan keluaran potensiometer diamati menggunakan voltmeter.
6. Hasil kuantisasi diamati menggunakan PC dengan *software Parallax Serial Terminal*.

- **Hasil pengujian**

Dalam Gambar 4.21 ditunjukkan hasil kuantisasi yang diamati dengan menggunakan *software Parallax Serial Terminal*, sedangkan dalam Tabel 4.2 hasil kuantisasi berdasarkan pengujian dibandingkan dengan hasil kuantisasi berdasarkan perancangan.



Gambar 4.21 Hasil Pengujian Proses Kuantisasi

Sumber: Hasil Pengujian

Tabel 4.2 Hasil Pengujian Proses Kuantisasi

| Tegangan Masukan (volt) | Hasil Kuantisasi Berdasarkan Perancangan | Hasil Kuantisasi Berdasarkan Pengujian | LSB <i>Error</i> |
|----------------------------|--|--|------------------|
| 1 | 51 | 52 | 1 |
| 2 | 102 | 104 | 2 |
| 3 | 153 | 155 | 2 |
| 4 | 204 | 206 | 2 |
| 5 | 255 | 255 | 0 |

Sumber: Hasil Pengujian

- **Analisis hasil pengujian**

Dalam Tabel 4.2 dapat diketahui bahwa *least significant bit* (LSB) *error* sesuai dengan *datasheet* yaitu ± 2 LSB, misalnya untuk data 3 volt data hasil perhitungan didapatkan sebesar :

$$\begin{aligned}
 \text{Hasil kuantisasi} &= \frac{3}{5} (2^8 - 1) \\
 &= \frac{3}{5} \times 255 \\
 &= 153
 \end{aligned}$$

Apabila hasil kuantisasi berdasarkan pengujian sebesar 155 dibandingkan dengan hasil kuantisasi berdasarkan perancangan sebesar 153, hasil kuantisasi memiliki kesalahan 2 nilai pada bit paling kanannya (*least significant bit*), sehingga proses kuantisasi yang diatur dalam mikrokontroler telah sesuai dengan proses kuantisasi yang dirancang.

4.4.5 Pengujian Proses Coding

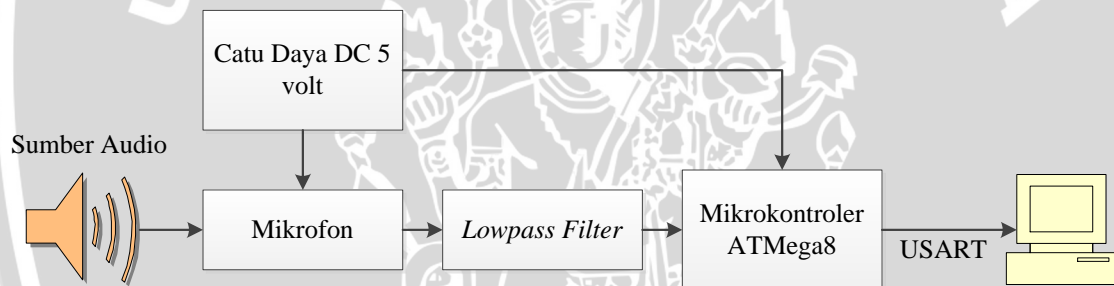
- **Tujuan pengujian**

Pengujian ini bertujuan untuk mengetahui kesesuaian proses *coding* yang dilakukan oleh mikrokontroler dengan perancangan.

- **Alat-alat yang digunakan**

1. Mikrokontroler ATmega 8
2. Catu daya DC 5 volt
3. PC
4. *Software Parallax Serial Terminal*
5. USB to TTL PL2303
6. Sumber audio berupa sinyal suara dengan frekuensi 3,8 kHz
7. Kabel konektor

- **Diagram blok pengujian**



Gambar 4.22 Diagram Blok Pengujian Proses Coding

Sumber: Pengujian

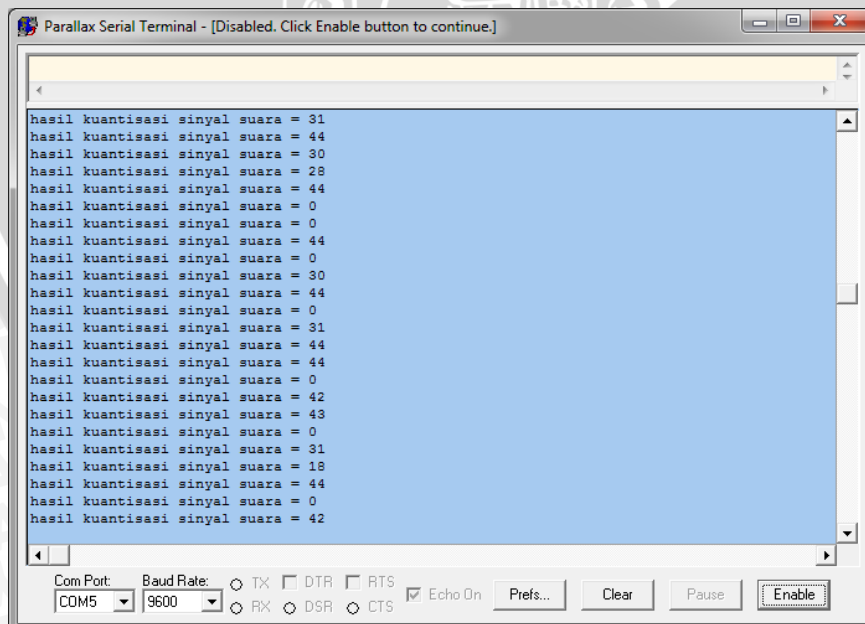
- **Langkah-langkah pengujian**

Pengujian proses coding dilakukan dengan langkah-langkah sebagai berikut :

1. Mikrofon dan mikrokontroler ATmega 8 dicatu dengan catu daya DC 5 volt.
2. Keluaran mikrofon dihubungkan dengan *lowpass filter*.
3. Keluaran *lowpass filter* dihubungkan dengan Pin ADC0 mikrokontroler ATmega8
4. Mikrokontroler ATmega8 dihubungkan dengan PC menggunakan USB to TTL PL2303 dengan komunikasi USART.
5. Sinyal audio dengan frekuensi 3,8 kHz diberikan pada mikrofon.
6. Hasil kuantisasi sinyal diamati menggunakan PC dengan *software Parallax Serial Terminal*.
7. Batas kuantisasi ditentukan berdasarkan hasil kuantisasi.
8. Variabel pulsa ditambah jika hasil kuantisasi melebihi batas kuantisasi
9. Jumlah pulsa dihitung selama 1 detik.
10. Jumlah pulsa diamati menggunakan PC dengan *software Parallax Serial Terminal*.

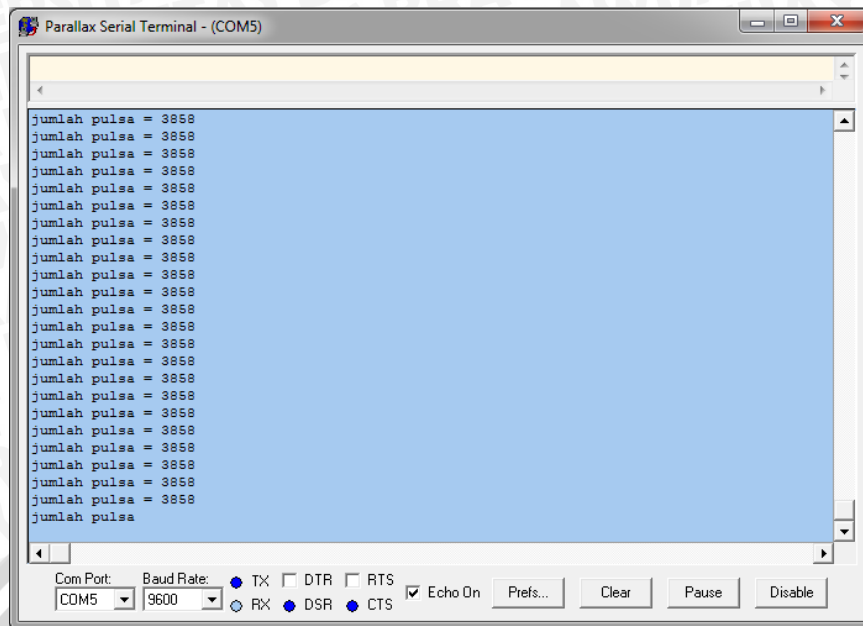
- **Hasil pengujian**

Hasil pengujian proses *coding* ditunjukkan dalam Gambar 4.23 dan Gambar 4.24.



Gambar 4.23 Hasil Proses Kuantisasi pada Sinyal Suara dengan Frekuensi 3800 Hz

Sumber: Hasil Pengujian



Gambar 4.24 Hasil Proses *Coding* pada Sinyal Suara dengan Frekuensi 3800 Hz

Sumber: Hasil Pengujian

- **Analisis hasil pengujian**

Dalam Gambar 4.23 diketahui bahwa nilai kuantisasi untuk sinyal suara dengan frekuensi 3800 Hz berkisar antara 18 hingga 44. Batas kuantisasi yang digunakan pada proses *coding* menggunakan nilai tengah dari nilai kuantisasi yang diperoleh yaitu sebesar 31. Dalam Gambar 4.24 diketahui jumlah pulsa dalam satu detik adalah sebesar 3858. Hasil proses *coding* ini sebanding dengan frekuensi sinyal suara yang diberikan yaitu sebesar 3800 Hz dengan nilai *error* 1,52 %.

4.4.6 Pengujian *Digital Bandpass Filter*

- **Tujuan pengujian**

Pengujian ini bertujuan untuk mengetahui kinerja *digital bandpass filter* yang dibuat dalam meredam sinyal dengan frekuensi dibawah 3,7 kHz dan diatas 3,9 kHz.

- **Alat-alat yang digunakan**

1. Mikrokontroler ATmega8
2. Catu daya DC 5 volt
3. PC
4. *Software Parallax Serial Terminal*
5. USB to TTL PL2303
6. Kabel konektor

- **Diagram blok pengujian**



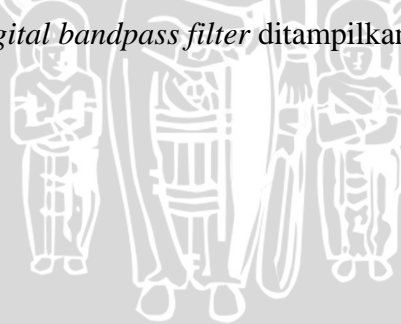
Gambar 4.25 Diagram Blok Pengujian *Digital Bandpass Filter*

Sumber: Pengujian

- **Langkah-langkah pengujian**

Pengujian *digital bandpass filter* dilakukan dengan langkah-langkah sebagai berikut :

1. Mikrokontroler ATmega 8 dicatu dengan catu daya DC 5 volt.
2. Mikrokontroler ATmega8 dihubungkan dengan PC menggunakan USB to TTL PL2303 dengan komunikasi USART.
3. Diberikan algoritma sinyal uji sebagai masukan filter yang berupa sinyal sinusoida diskrit dengan besar frekuensi berbeda yaitu sebesar 500 Hz, 1000 Hz, 2000 Hz, 3000 Hz, 3500 Hz, 3600 Hz, 3700 Hz, 3800 Hz, 3900 Hz, 4000 Hz, 4100 Hz, 4200 Hz, 4400 Hz, 4600 Hz, 4800 Hz, dan 5000 Hz.
4. Amplitudo sinyal keluaran *digital bandpass filter* diamati melalui *software Parallax Serial Terminal*, kemudian dicatat dalam tabel.
5. Hasil pengujian *digital bandpass filter* ditampilkan dalam grafik.



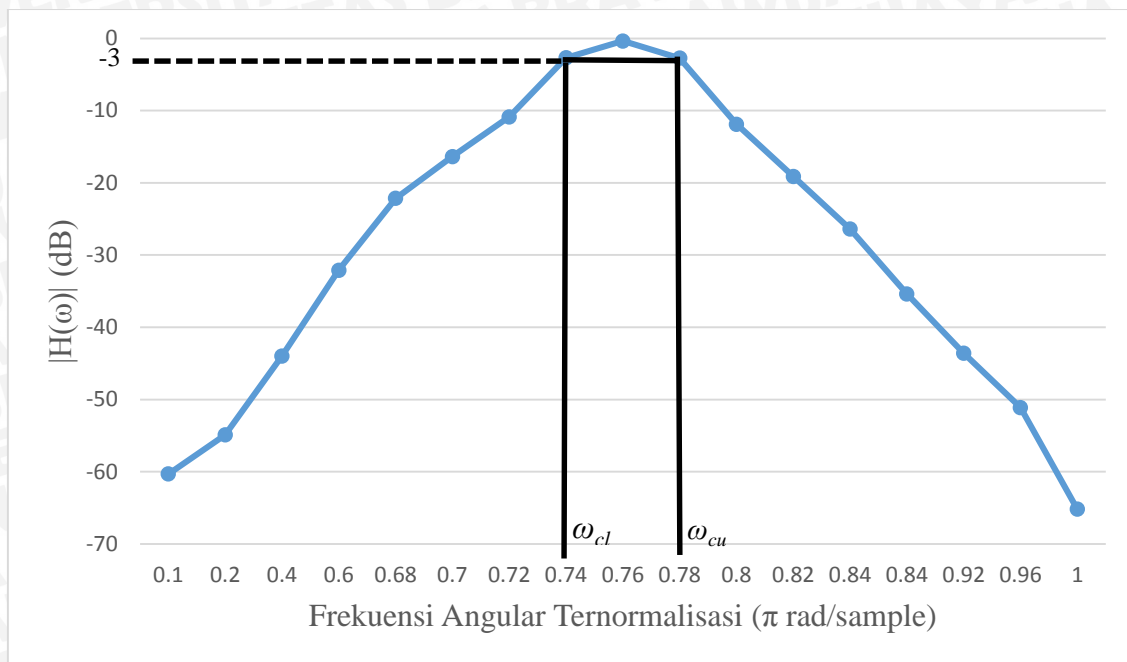
- **Hasil pengujian**

Hasil *pengujian digital bandpass filter* ditunjukkan dalam Tabel 4.3 dan Gambar 4.26.

Tabel 4.3 Hasil Pengujian *Digital Bandpass Filter*

| Frekuensi (Hz) | Frekuensi Angular Ternormalisasi (π rad/sample) | Amplitudo Sinyal Keluaran | $ H(\omega) $ | $ H(\omega) $ (dB) |
|----------------|--|---------------------------|---------------|--------------------|
| 500 | 0,1 | 0,001932 | 0,000966 | -60,3005 |
| 1000 | 0,2 | 0,003444 | 0,001799 | -54,9003 |
| 2000 | 0,4 | 0,012111 | 0,006326 | -43,9779 |
| 3000 | 0,6 | 0,048074 | 0,02483 | -32,1003 |
| 3400 | 0,68 | 0,156027 | 0,078023 | -22,1556 |
| 3500 | 0,7 | 0,304332 | 0,152178 | -16,353 |
| 3600 | 0,72 | 0,5722488 | 0,286639 | -10,8533 |
| 3700 | 0,74 | 1,466312 | 0,733322 | -2,6941 |
| 3800 | 0,76 | 1,919115 | 0,959775 | -0,3566 |
| 3900 | 0,78 | 1,460701 | 0,730604 | -2,7264 |
| 4000 | 0,8 | 0,4898 | 0,254599 | -11,8829 |
| 4100 | 0,82 | 0,221574 | 0,110823 | -19,1074 |
| 4200 | 0,84 | 0,095922 | 0,04797 | -26,3807 |
| 4400 | 0,88 | 0,033962 | 0,017018 | -35,3819 |
| 4600 | 0,92 | 0,013243 | 0,006623 | -43,5794 |
| 4800 | 0,96 | 0,005559 | 0,002784 | -51,1057 |
| 5000 | 1 | 0,000122 | 0,000552 | -65,1566 |

Sumber: Hasil Pengujian



Gambar 4.26 Grafik Respon Frekuensi *Digital Bandpass Filter*

Sumber: Hasil Pengujian

- **Analisis hasil pengujian**

Berdasarkan Tabel 4.3, dapat disimpulkan *digital bandpass filter* yang dibuat telah sesuai dengan yang dirancang, yaitu mampu meloloskan sinyal dengan frekuensi 3,8 kHz, dan meredam sinyal dengan frekuensi dibawah 3,7 kHz dan diatas 3,9 kHz. Ditunjukkan dengan nilai $|H(\omega)|$ pada frekuensi 3,7 kHz sebesar 0,733 (-2,69 dB) dengan nilai *error* $|H(\omega)|$ sebesar 3,67% dan pada frekuensi 3,9 kHz sebesar 0,73 (-2,72 dB) dengan nilai *error* $|H(\omega)|$ sebesar 3,15%.

Dalam Gambar 4.26 ditunjukkan grafik respon frekuensi *digital bandpass filter*. Pada frekuensi rendah hingga frekuensi *cutoff* bawah (3,7 kHz), nilai $|H(\omega)|$ semakin besar, sedangkan pada frekuensi 3,8 kHz nilai $|H(\omega)|$ sebesar 0,959 (-0,35 dB), hal ini menunjukkan tidak dilakukan peredaman sinyal keluaran pada sinyal dengan frekuensi 3,8 kHz, kemudian nilai $|H(\omega)|$ akan semakin rendah pada frekuensi tinggi dimulai dari frekuensi *cutoff* atas (3,9 kHz).

4.4.7 Pengujian Keseluruhan Sistem

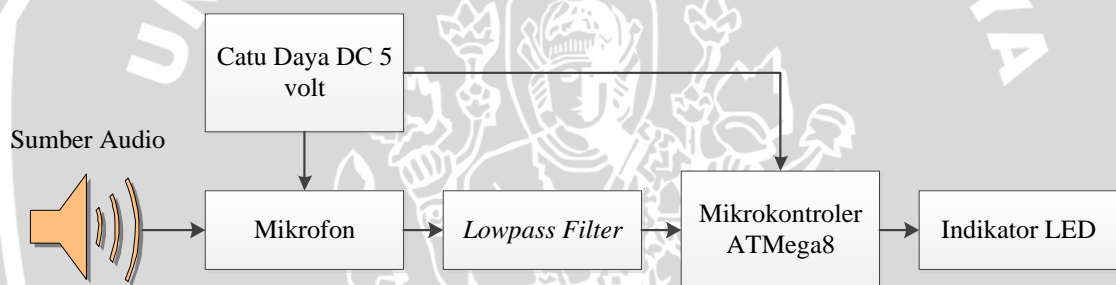
- **Tujuan pengujian**

Pengujian ini bertujuan untuk mengetahui kesesuaian sistem *sound activation* yang dibuat dengan perancangan.

- **Alat-alat yang digunakan**

1. Mikrokontroler ATmega8.
2. Catu daya DC 5 volt.
3. Sumber audio yang terdiri dari sinyal suara dengan frekuensi 3,8 kHz, musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara.
4. Indikator LED Hijau dan Merah.
5. Kabel konektor

- **Diagram blok pengujian**



Gambar 4.27 Diagram Blok Pengujian Keseluruhan Sistem

Sumber: Pengujian

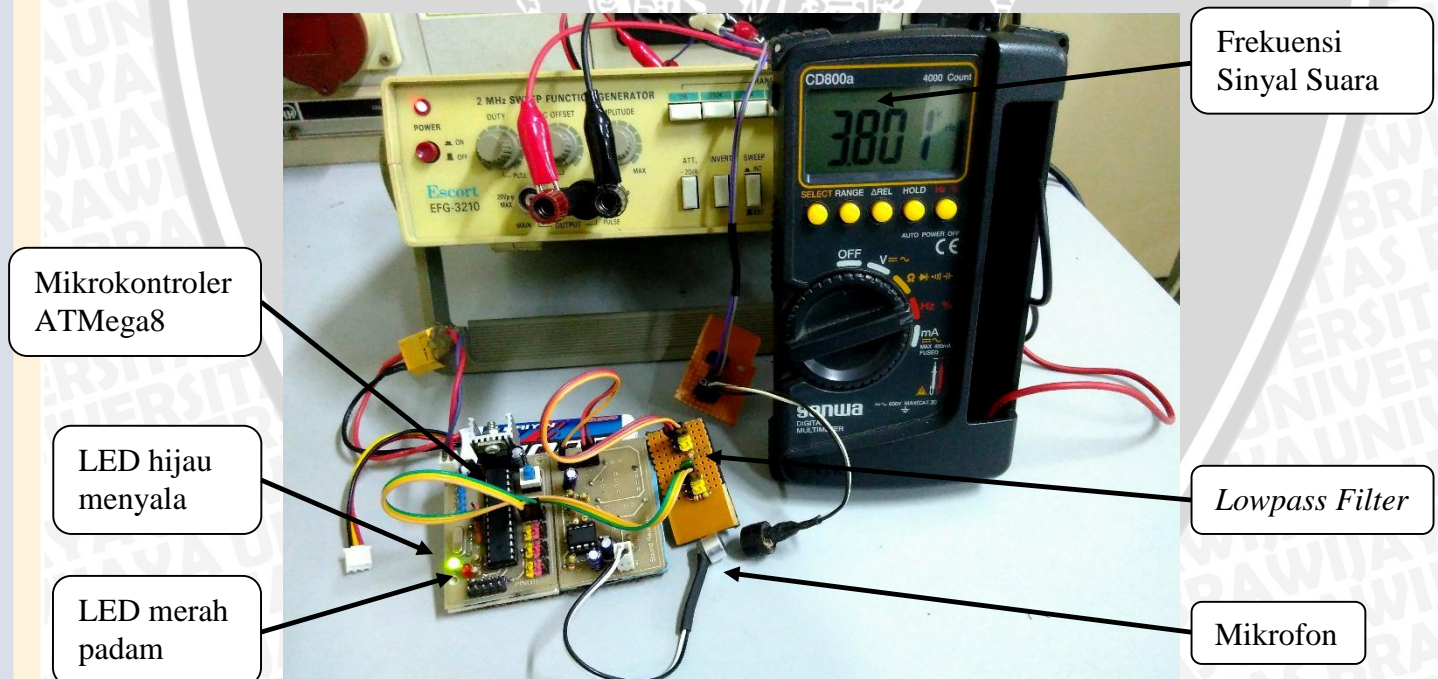
- **Langkah-langkah pengujian**

Pengujian keseluruhan sistem dilakukan dengan langkah-langkah sebagai berikut:

1. Mikrofon dan mikrokontroler ATmega8 dicatu dengan catu daya DC 5 volt.
2. Keluaran mikrofon dihubungkan dengan *lowpass filter*.
3. Keluaran *lowpass filter* dihubungkan dengan Pin ADC0 mikrokontroler ATmega8.
4. Keluaran mikrokontroler ATmega8 dihubungkan dengan indikator LED.
5. Diberikan sumber audio yang terdiri dari sinyal suara dengan frekuensi 3,8 kHz, musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara.
6. Hasil pengujian keseluruhan sistem diamati melalui indikator LED, LED hijau akan menyala ketika diberikan sinyal suara dengan frekuensi 3,8 kHz dan indikator LED merah akan menyala ketika sinyal suara yang diberikan bukan merupakan sinyal suara dengan frekuensi 3,8 kHz.

- **Hasil pengujian**

Hasil pengujian keseluruhan sistem ditunjukkan pada Gambar 4.28 dan Tabel 4.4.



Gambar 4.28 Hasil Pengujian Keseluruhan Sistem

Sumber: Hasil Pengujian

Tabel 4.4 Hasil Pengujian Keseluruhan Sitem

| Sumber Audio | LED Hijau | LED Merah | Keadaan yang Dikenali |
|---------------------------------------|---------------|---------------|-----------------------|
| Sinyal suara dengan frekuensi 3,8 kHz | Menyala | Tidak Menyala | Suara Aktivasi |
| Musik bambangan cakil | Tidak Menyala | Menyala | Bukan Suara Aktivasi |
| Suara tepuk tangan | Tidak Menyala | Menyala | Bukan Suara Aktivasi |
| Suara orang berbicara | Tidak Menyala | Menyala | Bukan Suara Aktivasi |

Sumber: Hasil Pengujian

- **Analisis hasil pengujian**

Berdasarkan Tabel 4.4, LED hijau menyala apabila diberikan sumber audio dengan frekuensi 3,8 kHz dan LED merah selalu padam. Kemudian pada saat sumber audio berupa musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara diberikan, LED hijau selalu padam dan LED merah menyala, Hal ini menunjukkan sistem sudah dapat mengenali suara aktivasi berupa sinyal suara dengan frekuensi 3,8 kHz, dan mengabaikan sumber audio berupa musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara, kemudian sistem mampu memberikan indikasi penyalan LED yang tepat.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan meliputi, perancangan, implementasi perancangan, dan pengujian sistem *sound activation*, dapat disimpulkan bahwa:

1. Sistem *sound activation* terdiri dari mikrofon yang berfungsi untuk mengubah sinyal suara menjadi sinyal elektrik, *lowpass filter* yang berfungsi untuk meredam sinyal dengan frekuensi diatas 5 kHz, proses *analog to digital converter* (ADC) yang terdiri dari proses *sampling* dengan frekuensi *sampling* 10 kHz, proses kuantisasi dengan 8 bit kuantisasi, dan proses *coding*, berfungsi untuk mengubah sinyal analog menjadi sinyal digital diskrit, dan proses komputasi menggunakan algoritma *digital bandpass filter* yang berfungsi untuk meloloskan sinyal dengan frekuensi 3,8 kHz, dan meredam sinyal dengan frekuensi dibawah 3,7 kHz dan diatas 3,9 kHz.
2. *Digital bandpass filter* yang dibuat mampu meredam sinyal dibawah frekuensi *cutoff* bawah (3,7 kHz) dengan nilai $|H(\omega)|$ sebesar 0,733 (-2,69 dB) dan nilai *error* $|H(\omega)|$ sebesar 3,67 % dan diatas frekuensi *cutoff* atas (3,9 kHz) dengan nilai $|H(j\omega)|$ sebesar 0,73 (-2,72 dB) dan nilai *error* $|H(\omega)|$ sebesar 3,15 %. Sistem *sound activation* mampu mendeteksi suara aktivasi berupa sinyal suara dengan frekuensi 3,8 kHz, dan mengabaikan sumber audio lain yang terdiri dari musik bambangan cakil, suara tepuk tangan, dan suara orang berbicara.

5.2 Saran

Saran yang digunakan untuk pengembangan penelitian ini adalah :

Penggunaan jenis filter analog *prototype* lainnya yaitu *chebyshev 1*, *chebyshev 2*, dan *elliptic* sebagai dasar untuk membuat desain *digital bandpass filter*, kemudian dibandingkan dengan menggunakan jenis filter analog *prototype butterworth* untuk mengetahui hasil respon respon frekuensi dari masing-masing filter analog *prototype* tersebut.

DAFTAR PUSTAKA

- Antoniou, A. 1993. *Digital Filters: Analysis, Design, and Applications*. Singapore: McGraw-Hill Inc.
- Atmel Corporation. 2003. *ATmega8/ATmega8L, 8-bit AVR Microcontroller with 32 kbytes in System Programmable Flash*. [http://www.atmel.com/Images/Atmel-2486-8-bit-AVR-microcontroller-ATmega8 L_datasheet.pdf](http://www.atmel.com/Images/Atmel-2486-8-bit-AVR-microcontroller-ATmega8_L_datasheet.pdf) . (Diakses 18 Mei 2015).
- Atmel Corporation. 2008. *AVR223: Digital Filter with AVR*. www.atmel.com/Images/doc2527.pdf. (Diakses 7 Juli 2015).
- Premier Farnell. 2011. *Electret Condenser Microphone ABM-713-RC*. www.farnell.com/datasheets/1505849.pdf, (Diakses 15 Mei 2015).
- Ramdhani, M. 2008. *Rangkaian Listrik*. Jakarta: Erlangga.
- Smith, S. W. 1997. *The Scientist and Engineer's Guide to Digital Signal Processing*. California: Technical Publishing.
- Sophocles J. O.. 2010. *Introduction to Signal Processing*. New Jersey: Prentice-Hall Inc.
- Tanudjaja, H. 2007. *Pengolahan Sinyal Digital Dan Sistem Pemrosesan Sinyal*. Yogyakarta: Andi.
- Winoto, A. 2010. *Mikrokontroler AVR ATmega8/16/32/8535 dan Pemrogramannya dengan Bahasa C pada WinAVR*. Bandung: INFORMATIKA.

LAMPIRAN



Lampiran I: Dokumentasi Alat

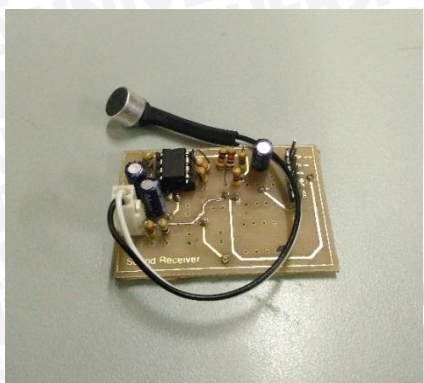


Foto 1. Rangkaian Elektronik Mikrofon

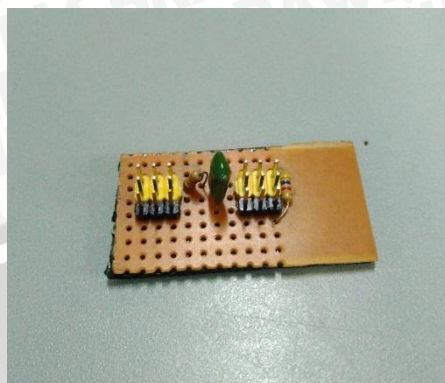


Foto 2. Rangkaian Elektronik *Lowpass Filter*

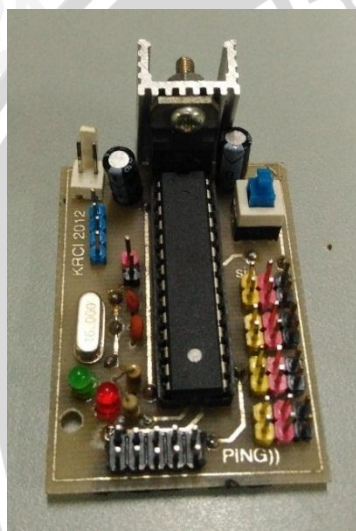


Foto 3. Rangkaian Elektronik Mikrokontroler

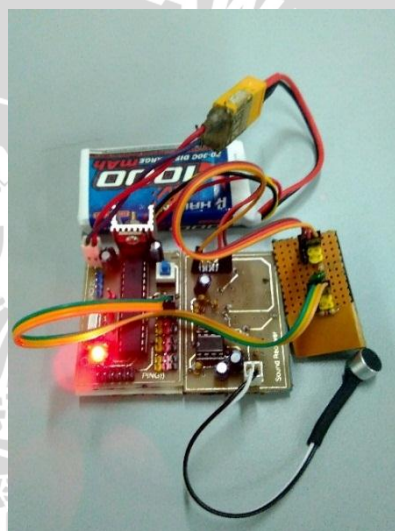


Foto 4. Sistem *Sound Activation* Keseluruhan



Foto 5. Robot KRPAI Subdivisi Beroda dan Berkaki

Lampiran II: Listing Program

```

/*****
Project : Sound Activation KRPAI
Version : 3.1
Date : 7/30/2015
Author : Guntur Eka Putro
Company : Brawijaya Robotic Team

Chip type : ATmega8
Program type : Application
AVR Core Clock frequency: 16.000000 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 256
*****/

#include <mega8.h>
#include <stdio.h>
#include <delay.h>
#include <math.h>
#define DATA 30

#define led_hijau PORTD.6=1;PORTD.7=0;PORTB.2=1;PORTB.3=0
#define led_merah PORTD.6=0;PORTD.7=1;PORTB.2=0;PORTB.3=1

char start=0, command=0, loop=0;
int pulsa=0, freq=0, aktivasi=0;

float
    x[5]={0,0,0,0,0}, //Pembilang fungsi alih
    y[5]={0,0,0,0,0}, //Penyebut fungsi alih
    b[5]={0,0,0,0,0}, //Koefisien Pembilang fungsi alih
    a[5]={0,0,0,0,0}, //Koefisien Penyebut fungsi alih
    sinyal[40],
    hasil[40],
    batas_atas = 0.2,
    batas_bawah = (-0.2),
    radian=0;

int n=4,k=0;

void filter(float data);
void inisialisasi();

// Timer1 overflow interrupt service routine
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
    TCNT1H=0xBDB >> 8;
    TCNT1L=0xBDB & 0xff;

    freq=pulsa;
    pulsa=0; aktivasi=0;

    start=1;
}

```



```

// Timer2 output compare interrupt service routine
interrupt [TIM2_COMP] void timer2_comp_isr(void)
{
    TCNT2 = 0x00;
    TCCR2 = 0x00;
    ADCSRA.7 = 1;
    ADCSRA.6 = 1;
    PORTB.0=~PORTB.0;
}

interrupt [ADC_INT] void adc_isr(void)
{
    int data;

    ADCSRA.6 = 0;
    ADCSRA.7 = 0;

    data=ADCH;
    if(data>=35)pulsa++;
    TCCR2 = 0x0A;
}

void filter(float data)
{
    x[n]=data;

    n=4;
    b[4]= 41;    a[4]= 10000;
    b[3]= 0;    a[4]= -27818;
    b[2]= -83;  a[4]= -37473;
    b[1]= 0;    a[1]= -25267;
    b[0]= 41;   a[0]= -8253;

    k=0;

    do
    {
        y[n]+=(b[n-k]*x[n-k]);
        k=k+1;
    }while(k<=n);

    k=1;

    do
    {
        y[n]+=(a[n-k]*y[n-k]);
        k=k+1;
    }while(k<=n);

    y[n]/=a[n];

    k=n;

    do
    {
        y[n-k]=y[n-k+1];
        x[n-k]=x[n-k+1];
        k=k-1;
    }while(k!=0);
}

```



```

void main(void)
{
    inisialisasi();
    while (1)
    {
        if(start==1)
        {
            for(loop=0;loop<=DATA;loop++)
            {
                radian=2*3.14*freq*loop/10000;
                sinyal[loop]=sin(radian);
            }

            for(loop=1;loop<=DATA;loop++)
            {
                filter(sinyal[loop]);
                if((y[n]>=batas_atas) || (y[n]<=batas_bawah)) aktivasi++;
            }
            if(aktivasi>=17){command=1;led_hijau;}
            else {command=0;led_merah;}

            start=0;
        }
    }
}

void inisialisasi()
{
    // Input/Output Ports initialization
    // Port B initialization
    // Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out
    // Func1=Out Func0=Out
    // State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0
    // State0=0
    PORTB=0x00;
    DDRB=0xFF;

    // Port D initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
    // Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    // State0=T
    PORTD=0x00;
    DDRD=0xFF;

    // Timer/Counter 1 initialization
    // Clock source: System Clock
    // Clock value: 62.500 kHz
    // Mode: Normal top=0xFFFF
    // OC1A output: Discon.
    // OC1B output: Discon.
    // Noise Canceler: Off
    // Input Capture on Falling Edge
    // Timer1 Overflow Interrupt: On
    // Input Capture Interrupt: Off
    // Compare A Match Interrupt: Off
    // Compare B Match Interrupt: Off

```

```
TCCR1A=0x00;
TCCR1B=0x04;
TCNT1H=0x0B;
TCNT1L=0xDB;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 2000.000 kHz
// Mode: CTC top=OCR2
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x0A;
TCNT2=0x00;
OCR2=0x8F;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x84;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x67;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC Clock frequency: 1000.000 kHz
// ADC Voltage Reference: AVCC pin
// Only the 8 most significant bits of
// the AD conversion result are used
ADMUX=0b01100000;
ADCSRA=0b11001100;

#asm("sei")
}
```

Lampiran III: Datasheet

Electret Condenser Microphone

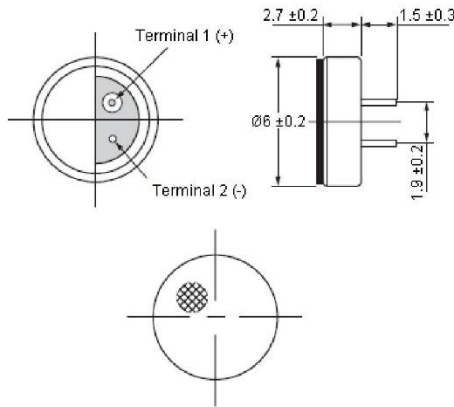
pro-SIGNAL

ABM-713-RC



Specifications:

| | |
|---|--------------------|
| Directivity | : Omni directional |
| Terminal | : Pin type |
| Standard operating voltage | : 2 V |
| Maximum operating voltage | : 10 V |
| Maximum current consumption | : 0.5 mA |
| Output impedance (F = 1 KHz 1 Pa) | : 2.2 KΩ |
| Frequency | : 50 to 16,000 Hz |
| Sensitivity (F = 1 KHz, 0 dB = 1 VPa) | : -42 ±2 dB |
| Sensitivity reduction (at 1.5 V - 3 V) | : -3 dB |
| Signal to noise ratio (F = 1 KHz 1 Pa) a weighted | : 60 dB |
| Electrets type | : Back type |
| Material | : Al |
| Storage temperature | : -25°C to +70°C |
| Operating temperature | : -20°C to +60°C |



Dimensions : Millimetres

Part Number Table

| Description | Part Number |
|-------------------------------------|-------------|
| Electret Microphone, Omni, PCB Pins | ABM-713-RC |

Important Notice : This data sheet and its contents (the "Information") belong to the members of the Premier Farnell group of companies (the "Group") or are licensed to it. No license is granted for the use of it other than for information purposes in connection with the products to which it relates. No licence of any intellectual property rights is granted. The Information is subject to change without notice and replaces all data sheets previously supplied. The Information supplied is believed to be accurate but the Group assumes no responsibility for its accuracy or completeness, any error in or omission from it or for any use made of it. Users of this data sheet should check for themselves the information and the suitability of the products for their purpose and not make any assumptions based on information included or omitted. Liability for loss or damage resulting from any reliance on the Information or use of it (including liability resulting from negligence or where the Group was aware of the possibility of such loss or damage arising) is excluded. This will not operate to limit or restrict the Group's liability for death or personal injury resulting from its negligence. pro-SIGNAL is the registered trademark of the Group. © Premier Farnell plc 2011.

www.element14.com
 www.farnell.com
 www.newark.com
 www.cpc.co.uk

pro-SIGNAL

15/03/11 V1.1



Features

- High-performance, Low-power AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 8K Bytes of In-System Self-Programmable Flash
 - Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - 512 Bytes EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - 1K Byte Internal SRAM
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Three PWM Channels
 - 8-channel ADC in TQFP and MLF package
 - Six Channels 10-bit Accuracy
 - Two Channels 8-bit Accuracy
 - 6-channel ADC in PDIP package
 - Four Channels 10-bit Accuracy
 - Two Channels 8-bit Accuracy
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-lead PDIP, 32-lead TQFP, and 32-pad MLF
- Operating Voltages
 - 2.7 - 5.5V (ATmega8L)
 - 4.5 - 5.5V (ATmega8)
- Speed Grades
 - 0 - 8 MHz (ATmega8L)
 - 0 - 16 MHz (ATmega8)
- Power Consumption at 4 Mhz, 3V, 25°C
 - Active: 3.6 mA
 - Idle Mode: 1.0 mA
 - Power-down Mode: 0.5 μ A



8-bit AVR[®]
with 8K Bytes
In-System
Programmable
Flash

ATmega8
ATmega8L

Rev. 2486M-AVR-12/03

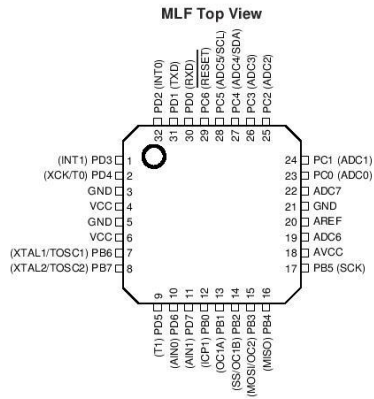
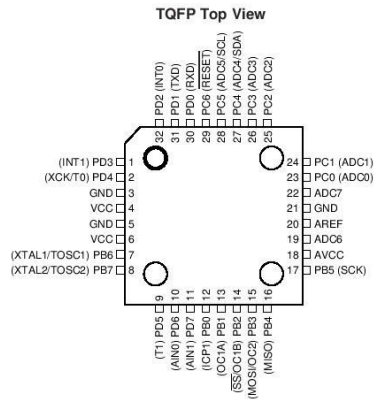




Pin Configurations

PDIP

| | | | |
|-------------------|----|----|----------------|
| (RESET) PC6 | 1 | 28 | PC5 (ADC5/SCL) |
| (RXD) PD0 | 2 | 27 | PC4 (ADC4/SDA) |
| (TXD) PD1 | 3 | 26 | PC3 (ADC3) |
| (INT0) PD2 | 4 | 25 | PC2 (ADC2) |
| (INT1) PD3 | 5 | 24 | PC1 (ADC1) |
| (XCK/T0) PD4 | 6 | 23 | PC0 (ADC0) |
| VCC | 7 | 22 | GND |
| GND | 8 | 21 | AREF |
| (XTAL1/TOSC1) PB6 | 9 | 20 | AVCC |
| (XTAL2/TOSC2) PB7 | 10 | 19 | PB5 (SCK) |
| (T1) PD5 | 11 | 18 | PB4 (MISO) |
| (AIN0) PD6 | 12 | 17 | PB3 (MOSI/OC2) |
| (AIN1) PD7 | 13 | 16 | PB2 (SS/OC1B) |
| (ICP1) PB0 | 14 | 15 | PB1 (OC1A) |



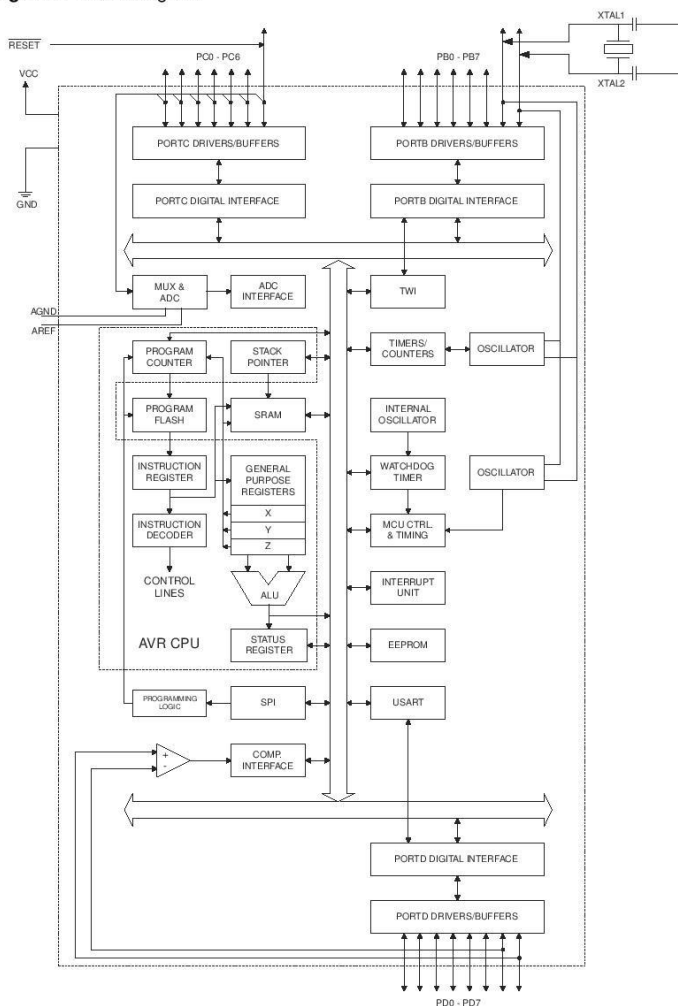
ATmega8(L)

Overview

The ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1 MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 1. Block Diagram





The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega8 provides the following features: 8K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512 bytes of EEPROM, 1K byte of SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, a 6-channel ADC (eight channels in TQFP and MLF packages) where four (six) channels have 10-bit accuracy and two channels have 8-bit accuracy, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next Interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption.

The device is manufactured using Atmel's high density non-volatile memory technology. The Flash Program memory can be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash Section will continue to run while the Application Flash Section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega8 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega8 AVR is supported with a full suite of program and system development tools, including C compilers, macro assemblers, program debugger/simulators, In-Circuit Emulators, and evaluation kits.

Disclaimer

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

ATmega8(L)

Pin Descriptions

| | |
|---|---|
| VCC | Digital supply voltage. |
| GND | Ground. |
| Port B (PB7..PB0) XTAL1/ XTAL2/TOSC1/TOSC2 | <p>Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.</p> <p>Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.</p> <p>If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.</p> <p>The various special features of Port B are elaborated in "Alternate Functions of Port B" on page 56 and "System Clock and Clock Options" on page 23.</p> |
| Port C (PC5..PC0) | <p>Port C is an 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> |
| PC6/RESET | <p>If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.</p> <p>If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 36. Shorter pulses are not guaranteed to generate a Reset.</p> <p>The various special features of Port C are elaborated on page 59.</p> |
| Port D (PD7..PD0) | <p>Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port D also serves the functions of various special features of the ATmega8 as listed on page 61.</p> |
| RESET | <p>Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 36. Shorter pulses are not guaranteed to generate a reset.</p> |



| | |
|--|---|
| AVCC | AVCC is the supply voltage pin for the A/D Converter, Port C (3..0), and ADC (7..6). It should be externally connected to V_{CC} , even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that Port C (5..4) use digital supply voltage, V_{CC} . |
| AREF | AREF is the analog reference pin for the A/D Converter. |
| ADC7..6 (TQFP and MLF Package Only) | In the TQFP and MLF package, ADC7..6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels. |
| About Code Examples | This datasheet contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details. |

