

## Perancangan Dan Simulasi Filter IIR Menggunakan Graphical User Interface (GUI)

Umi Murdika<sup>1</sup>, Yessi Mulyani<sup>2</sup>

Jurusan Teknik Elektro Universitas Lampung, Bandar Lampung  
Jl. Prof. Sumantri Brojonegoro No.1 Bandar Lampung 35145

<sup>1</sup>Umi.murdika@eng.unila.ac.id

<sup>2</sup>Yessimulyani@eng.unila.ac.id

**Intisari---** Pada makalah ini telah disajikan metode mensimulasikan filter IIR menggunakan GUI. Metode ini tidak hanya tergantung pada kode matlab, tetapi juga menampilkan control penggunaan yang dibangun berdasarkan GUI, sehingga semua operasi pemfilteran dapat diselesaikan oleh GUI. Makalah ini menggunakan metode Pole-zero placement, Impulse invariant, Matched z-transform, dan Bilinear z-transform dalam merealisasikan filter IIR dan untuk sebagai contoh untuk merancang filter low-pass, high-pass and band-pass. Dari simulasi menunjukkan bahwa perancangan berdasarkan GUI ini sangatlah nyaman, cepat, dan fleksibel

**Kata kunci---** GUI, IIR, Filter Digital, Simulasi

**Abstract---** A method to simulate the IIR filter based on GUI(Graphic User Interface) is introduced in this paper. This method not only depended on Matlab code, but also made use of controls which generate a GUI, All the operations have been done by GUI. This paper took Pole-zero placement, Impulse invariant, Matched z-transform method to realize IIR filter for example to design digital low-pass, high-pass and band-pass filters. The simulation results show that the design based on GUI is convenient, fast, and flexible.

**Keywords---** GUI, IIR, Digital Filter, Simulation

### I.PENDAHULUAN

Proses filter adalah melewatkan sinyal yang diinginkan dengan menggunakan filter dan bertujuan untuk membuang partikel-partikel yang tidak diinginkan atau impuritas. Dalam sinyal prosesing, filter digunakan untuk membuang bagian sinyal yang tidak diinginkan, misalnya noise.

Filter IIR merupakan filter yang memegang peranan yang cukup penting dalam pengolahan sinyal digital. Dibandingkan dengan filter analog, filter ini memiliki banyak manfaat dan keuntungan, yaitu dalam hal keandalan yang tinggi dan stabilitas yang baik tanpa banyak menimbulkan masalah derau.

Simulasi filter IIR ini dilakukan dengan bahasa Matlab yang memiliki kelebihan seperti efisiensi programmable yang tinggi,

instruksi sederhana dan mudah dan data yang divisualisasikan. Proses mendesain filter dimulai dengan spesifikasi filter, yang mana di dalamnya ada respon frekuensi magnitude dan fasa pada, spesifikasi tipe filter, dan orde filter. Sebagai catatan spesifikasi filter harus ditentukan, langkah selanjutnya adalah menentukan koefisien filter agar menghasilkan filter yang diinginkan. Lalu filter siap didesain. Dalam makalah ini akan merancang filter digital khususnya IIR (Infinite Impulse Response). Jadi metode untuk mensimulasikan filter IIR berbasis GUI (Graphic User Interface) diusulkan dalam makalah ini. Metode ini tidak hanya bergantung pada kode Matlab, tapi juga memanfaatkan sepenuhnya keunggulan GUI: kenyamanan, intuisi dan interaksi yang baik antara pengguna dan komputer.

## II. METODE PERHITUNGAN KOEFISIEN PADA PERANCANGAN FILTER IIR

IIR filter dikarakterisasi oleh persamaan berikut:[1]

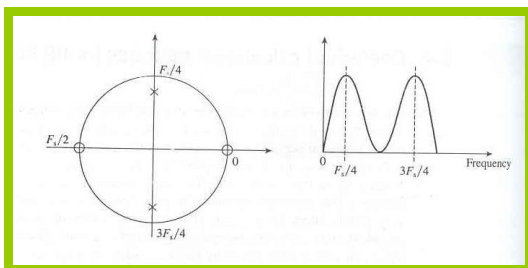
$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \quad (1)$$

$$y(n) = \sum_{k=0}^N b_k x(n-k) - \sum_{k=0}^M a_k y(n-k) \quad (2)$$

Dimana  $h(k)$  merupakan impulse response dari filter dimana secara teoritis infinite dalam durasi,  $b_k$  dan  $a_k$  adalah koefisien filter sedangkan  $x(n)$  dan  $y(n)$  adalah input dan output pada filter. Terdapat 4 metoda untuk menghitung koefisien IIR filter, yaitu : Pole-zero placement, Impulse invariant, Matched z-transform, Bilinear z-transform.

### A. Pole-zero placement

Idenya adalah saat zero diletakkan pada titik yang diberikan pada z-plane, frekuensi response akan menjadi zero pada titik yang tersebut sementara pole menghasilkan nilai puncak pada titik frekuensi tersebut. Sebagai catatan bahwa untuk koefisien filter riil, maka pole dan zero juga harus riil.



Gbr. 1 Penempatan Pole-zero[1]

### B. Impulse invariant

Terdapat beberapa komponen yang penting dan didapatkan dengan menggunakan Transformasi Laplace dan juga Transformasi z. Dimana  $H(s)$  merupakan fungsi transfer analog (analog transfer function),  $h(t)$  impulse response,  $h(nT)$  adalah z transform dengan  $T$  interval sampling dan  $H(z)$  sebagai fungsi transfer (transfer function).[1]

### C. Matched z-transform

Memberikan cara sederhana untuk mengubah analog filter menjadi equivalent digital filter. Idenya adalah tiap tiap poles dan zeros analog filter langsung dipetakan dari s-plane menjadi z-plane menggunakan persamaan berikut :

$$(s-a) \rightarrow (1-z^{-1}e^{aT}) \quad (3)$$

Persamaan memetakan pole atau zero pada lokasi  $s=a$  pada s-plane, menjadi pole atau zero pada z-plane pada  $z=e^{aT}$  [1]

### D. Bilinear z-transform

Merupakan metode yang sangat penting, Idenya adalah untuk mengubah analog filter  $H(s)$  menjadi equivalent digital filter adalah mengganti  $s$  sebagai berikut:

$$s = k \frac{z-1}{z+1}, k=1 \quad \text{or} \quad k = \frac{2}{T} \quad (4)$$

Transformasi tersebut memetakan analog transfer function,  $H(s)$ , dari s-plane menjadi discrete transfer function,  $H(z)$ , pada z-plane.[1]

## III. REALISASI FILTER IIR MENGGUNAKAN GUI MATLAB

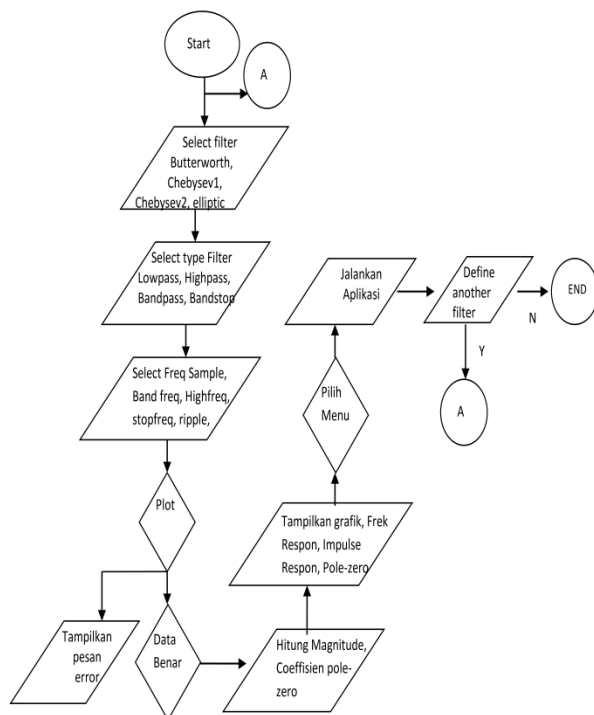
### A. Mengatur tata letak komponen

Untuk mengatur letak komponen maka dapat dipergunakan fasilitas *alignment tool* yang terdapat pada jendela GUI. Tampilan menu yang terdapat pada program simulasi mempergunakan tampilan menu yang sudah ada. Hal ini dilakukan dengan cara mengganti *property editor* dari *figure*. Setelah tampilan dari program simulasi selesai dibuat, maka langkah terakhir yang harus dilakukan adalah menyimpan *figure* yang telah dibuat. Jika *figure* tersebut telah disimpan, maka sudah dapat ditampilkan jendela M-file yang dipergunakan untuk memasukkan program. Untuk memprogram komponen yang dipergunakan, dilakukan dengan menggunakan M-file. Untuk memunculkan jendela M-file dilakukan dengan cara meng-click icon yang terdapat pada Kemudian s editor ditampilkan kita dapat dimulai

pemrograman. M-file terdiri dari tiga bagian yaitu *opening function*, *callback* dan *output function*. Program dimasukkan pada bagian *callback*. Untuk mencari *callback function* dari suatu komponen dapat dilakukan melalui icon dengan label huruf f, sesuai dengan tag dari komponen tersebut. Sebagai contoh untuk mencari *callback function* sebuah *push button* dengan tag “*push button\_run*”.[4]

### B. Memasukkan Parameter-parameter filter IIR

Skema prancangan filter IIR ini meliputi beberapa tahapan yang diperlihatkan pada gambar 1 dibawah ini.



Gbr. 2 Skema Perancangan Filter IIR

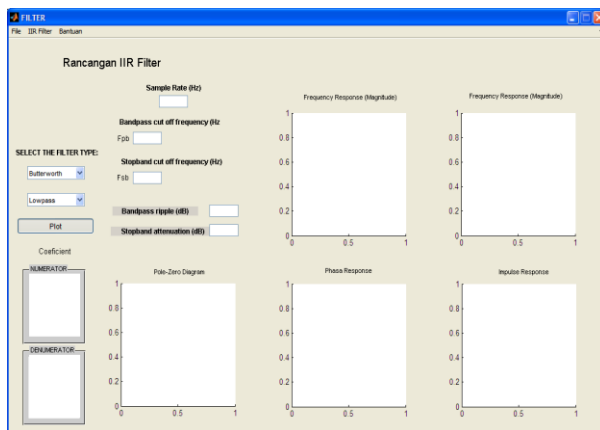
Untuk mendapatkan jenis IIR filter yang diinginkan, ketikkan spesifikasi filter yang kita inginkan dengan memilihnya pada pop up menu type filter. Adapun langkah-langkah yang dilakukan dalam merancang filter IIR ini meliputi :

- Pilihlah jenis spesifikasi jenis filter yang diinginkan, apakah jenis Butterworth, chebyshev I, Chebisev II, ataukah jenis Eliptic, dengan memilih pada pop up menu yang terdapat pada GUI interface.

- Dalam merancang filter FIR yang diinginkan, pertama kali pilihlah salah satu dari filter ideal apakah LPF, HPF, BPF, atau BPF. Jika memilih filter LPF atau HPF maka hanya diperlukan satu frekuensi cut off saja, tetapi jika memilih filter BPF atau BSF diperlukan dua frekuensi cut off yaitu frekuensi cut off 1 (low) dan frekuensi cut off 2 (high).
- Masukkan nilai frekuensi sampling, dimana nilai setengah frekuensi sampling adalah harus lebih besar dari frekuensi cut off, kemudian frekuensi ripple dan attenuasi juga diinputkan. Jika frekuensi input satuannya Hz maka sebaiknya frekuensi sampling juga menggunakan besaran Hz. Lalu apabila parameter-parameter diatas sudah terpenuhi, maka gambar respon input bisa ditampilkan dengan meng-klik tombol ‘plot’.
- Setelah parameter-parameter di atas sudah ditentukan dengan benar, maka hasilnya siap ditampilkan.
- Klik pada tombol plot maka akan muncul gambar respon magnitude dan respon fasa, diagram pole-zero juga respon impulse dari IIR filter. Juga akan ditampilkan nilai koefisien denominator dan numerator beserta orde dari filter tersebut.

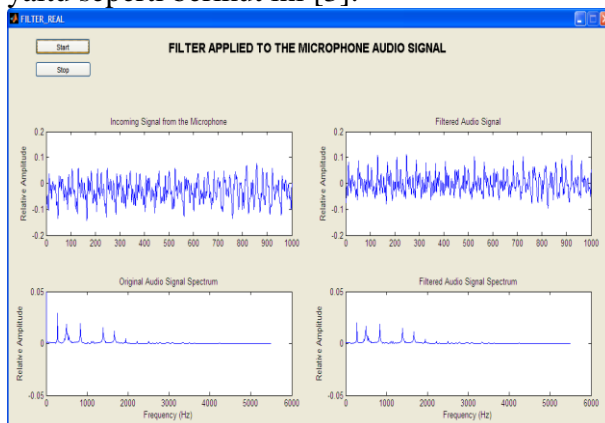
## IV. SIMULASI FILTER IIR

Perancangan filter yang dibuat disini terdiri dari 3 buah M-file yang dijalankan menggunakan Matlab yaitu file utama dengan nama FILTER.m yang merupakan file utama yang berisi parameter-parameter perancangan Filter IIR dan disinilah kita bisa menjalankan program perancangan filter IIR ini, FILTER\_FILE.m berisi komponen komponen yang dipakai dalam aplikasi filter menggunakan file dan FILTER\_REAL.m juga merupakan program aplikasi untuk aplikasi real time. Untuk menjalankan program ini, pertama-tama buka M-file utama yaitu FILTER.m kemudian jalankan program dengan mengklik “run FILTER.m”. Sehingga akan muncul Gui yang sudah kita rancang sebagai berikut:



Gbr. 3 Gui kosong Filter IIR

Untuk menjalankan aplikasi, disediakan dua jenis aplikasi yaitu pertama aplikasi filter yang digunakan untuk memfilter suara yang direcord langsung oleh komputer dan yang kedua yang diunduh dari file Wav [2]. Untuk aplikasi pertama, pilihlah dari menu pada tampilan gui FILTER, klik IIR Filter kemudian pilih Aplikasi Real Time maka akan muncul gui FILTER\_REAL, kemudian klik tombol start dan bunyikanlah suara misalnya “AAAAAAAAAAAA”, maka microphone komputer akan merekamnya dan menampilkan spektrum dari audio sinyal yang sudah direkam oleh microphone tadi yaitu seperti berikut ini [3].



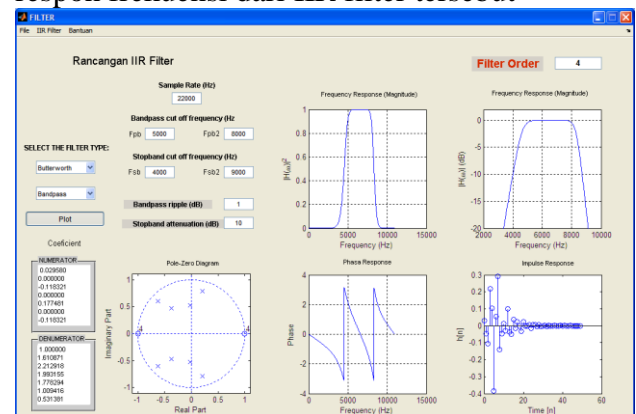
Gbr. 4 spektrum dari audio sinyal direkam oleh microphone dan diterapkan filter

Bila sudah selesai maka klik tombol stop, akan muncul tombol define another filter yang dapat kita tekan, maka layar akan kembali ke menu gui FILTER. Berikutnya kita coba untuk aplikasi kedua menggunakan file yang diunduh dari luar dengan extensi WAV. Pilih dari menu FILTER IIR Filter pilihlah Aplikasi File wav, maka akan muncul gui FILTER\_FILE. Pertama klik

muatkan audio file, kemudian pilihlah file dengan ekstension wav, maka akan tampil pada layar spektrum sinyal audio, kemudian klik Aplikasi IIR filter untuk melihat hasil sinyal suara yang telah mengalami pemfilteran. Atau kita ingin spesifikasi filter yang berbeda misalnya kita ingin merancang Bandpass filter type Butterworth, dengan spesifikasi sebagai berikut:

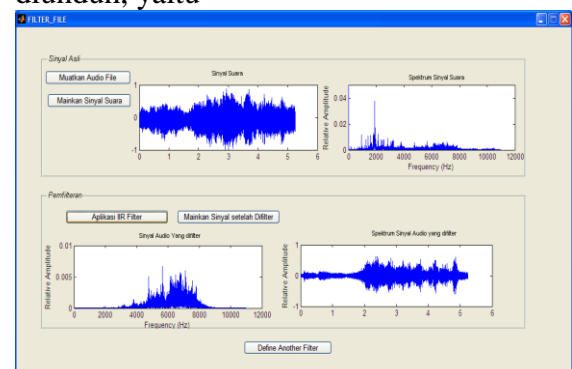
- Frekuensi sampling 22000 Hz
- Bandpass cut off  
Frekuensi 5000Hz - 8000Hz
- Stopband cut off  
Frekuensi 4000Hz – 9000Hz
- Bandpass Riple 1 dB
- Stopband attenuation 10 dB

Sehingga diperoleh hasil tampilan grafik respon frekuensi dari IIR filter tersebut



Gbr. 5 Grafik respon frekuensi dari IIR filter

Maka akan menampilkan hasil aplikasi dengan menggunakan file wav yang sudah diunduh, yaitu



Gbr. 6 Spektrum dari audio sinyal WAV

## V. KESIMPULAN

Filter IIR dari (*Infinite Impulse Response*) adalah salah satu tipe dari filter digital yang diaplikasikan pada *Digital Signal Processing* (DSP). Keuntungan filter IIR antara lain adalah membutuhkan koefisien yang lebih sedikit untuk respon frekuensi yang curam sehingga dapat mengurangi jumlah waktu komputasi. Filter IIR memiliki respon impuls fungsi yang non-nol melalui panjang tanpa batas waktu. Hal ini berbeda dengan respon impuls terbatas (FIR) filter yang memiliki respon impuls durasi tetap.

## REFERENSI

- [1] Imanuel C. Ifeachor, Barrie W. Jervis; *Digital Signal Processing. Practical Approach*, Second edition 2002.
- [2] MathWork, file akses tanggal juni 2017, <http://www.mathworks.com/matlabcentral/fileexchange/19683-audio-filter-gui-demo>.
- [3] MathWork, file akses tanggal juni 2017 <http://www.mathworks.com/matlabcentral/fileexchange/?term=realtime+iir+filter>
- [4] Hunt Lipsman & Rosenberg, *A Guide to MATLAB for Beginners and Experienced Users*, Cambridge University Press, 2001.