

IMPLEMENTASI FILTER DIGITAL FIR (*FINITE IMPULSE RESPONSE*) PADA *FIELD PROGRAMMABLE GATE ARRAYS* (FPGA)

Rizki Jumadil Putra, Mochammad Rif'an, ST., MT., dan Raden Arief Setyawan ST.,MT.

Abstrak—Filter FIR merupakan filter digital yang mempunyai *impulse* terbatas. FPGA merupakan salah satu *hardware* yang dapat mengaplikasikan Filter FIR. Kelebihan memakai FPGA untuk pemrosesan sinyal digital antara lain adalah mempunyai kemampuan untuk menangani beban komputasi yang berat karena FPGA bekerja secara paralel.

Hasil pengujian Filter FIR yang telah diimplementasikan pada FPGA menunjukkan ketika filter digital FIR diberi masukan sinyal sinusoida 20-22.000 Hz, maka sinyal dibawah 20.000 Hz tetap diloloskan, sedangkan sinyal diatasnya akan teredam. Hal ini menunjukkan kinerja dari filter FIR, baik dengan metode Hamming window, Blackman window maupun Hanning window, filter akan meloloskan sinyal yang frekuensi nya lebih rendah dari frekuensi cut-off yaitu sebesar 20KHz, dan akan meredam frekuensi diatasnya

Kata Kunci— FPGA, Filter FIR, Modul Hamming window, Hanning window, Blackman window.

I. PENDAHULUAN

Filter adalah suatu perangkat yang menghilangkan bagian dari sinyal yang tidak di inginkan. Filter di gunakan untuk mengekstraksi sinyal yang di inginkan dari sinyal noise. Filter digital merupakan suatu prosedur matematika atau algoritma yang mengolah sinyal masukan dan menghasilkan isyarat keluaran digital yang memiliki sifat tertentu sesuai tujuan filter. Filter digital dapat dibagi menjadi dua, yaitu filter digital FIR (*Finite Impulse Response*) dan filter digital IIR (*Infinite Impulse Response*). Pembagian ini berdasarkan pada tanggapan impulse filter tersebut. FIR memiliki tanggapan impulse yang panjangnya terbatas, sedangkan IIR tidak terbatas. FIR tidak memiliki pole, maka kestabilan dapat dijamin sedangkan pada IIR memiliki pole-pole sehingga lebih tidak stabil. Secara matematis, persamaan filter FIR dapat ditulis :

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

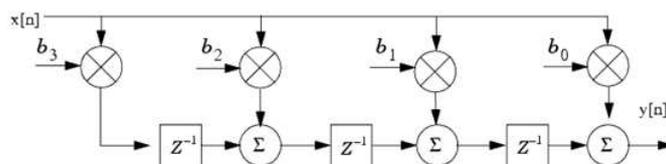
Dengan menggunakan transformasi Z, didapat fungsi alihnya sebagai berikut :

$$H(z) = \sum_{k=0}^{N-1} \binom{n}{k} h(k) Z^{-k} \dots \dots \dots (2)$$

Rizki Jumadil Putra adalah mahasiswa program sarjana Teknik Elektro Universitas Brawijaya, Malang, Indonesia (penulis dapat dihubungi melalui email: rizki.jumadil@yahoo.com).

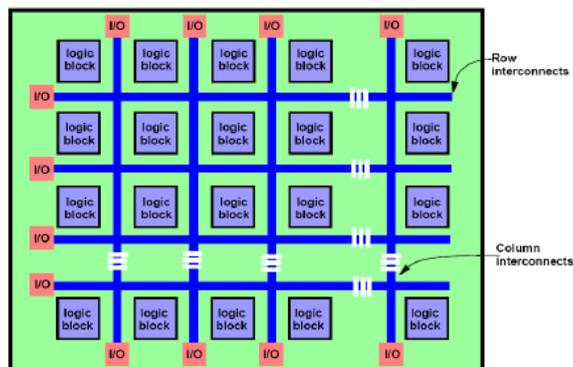
Mochammad Rif'an, ST., MT. dan Raden Arief Setyawan ST., MT adalah staf pengajar program sarjana Teknik Elektro Universitas Brawijaya, Malang, Indonesia (email: rif_an91@yahoo.com; rarief@ub.ac.id).

Secara grafis, struktur filter FIR ditunjukkan dalam Gambar 1 :



Gambar 1. Struktur Filter FIR

Salah satu *hardware* untuk mengimplementasikan Filter digital FIR adalah FPGA. FPGA adalah sebuah intergrated circuit yang didesain untuk dapat di konfigurasi oleh user atau designer setelah keluar dari produksi. Pengkonfigurasiian FPGA pada umumnya adalah spesifik menggunakan deskripsi bahasa hardware atau HDL (*Hardware Description Language*).



Gambar 2. Arsitektur FPGA

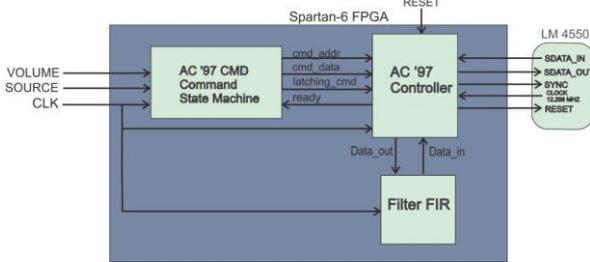
Dalam skripsi ini, penulis menggunakan board Atlys, salah satu board berbasis FPGA Xilinx Spartan 6 LX45 yang mempunyai bermacam peripheral, salah satunya adalah AC97 Audio



Gambar 2. Atlys circuit board

II. PERANCANGAN DAN PEMBUATAN PROGRAM

Perancangan dimulai dengan membuat blok diagram sistem. Gambar 1 menunjukkan blok diagram system Implementasi Filter FIR pada FPGA.



Gambar 3. Blok Diagram Sistem Implementasi Filter FIR pada FPGA
Penjelasan mengenai blok diagram di atas adalah sebagai berikut:

- 1) Input dan output dari sistem adalah Audio (sinyal analog) yang diterima oleh ADC LM4550
- 2) Blok AC'97 Controller berfungsi untuk *generate* sinyal dan mengubah 18 bit data parallel menjadi data serial untuk mengantarmuka ke *chip* AC97 dengan FPGA
- 3) Blok AC97 Command State Machine adalah untuk menanganiregister pada AC97
- 4) Blok Filter FIR berfungsi untuk mengolah sinyal dengan menggunakan Filter FIR metode Hamming Window

A. Perancangan AC97 codec

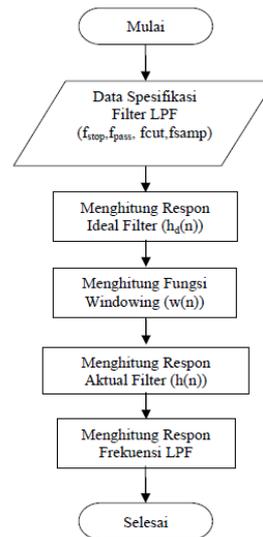
Perancangan program menggunakan software ISE Xilinx *Design Suite* 14.3 dan bahasa pemrograman VHDL (VHSIC Hardware Design Language).

Blok ini memiliki input n_reset , clk , $ac97_sdata_in$, $ac97_bitclk$, L_out , R_out , $latching_cmd$, cmd_addr , cmd_data dan output $ac97_sdata_out$, $ac97_sync$, $ac97_n_reset$, $ac97_ready_sig$, L_in , R_in .

Dalam blok sistem ini terdapat 4 blok proses yaitu proses Delay AC97 reset, proses *Generate Signal Pulse*, proses AC Link Output Frame, proses AC Link Input Frame

B. Perancangan Filter FIR

Dalam merancang filter FIR terdapat beberapa tahap, yang pertama adalah menentukan data spesifikasi filter FIR yaitu frekuensi stop (F_{stop}), frekuensi pass (F_{pass}), frekuensi *cut-off* (F_{cut}), dan frekuensi sampling (F_{samp}). Frekuensi cutoff (F_c) di dapat dari frekuensi batas pendengaran manusia yaitu 20.000 Hz, sedangkan frekuensi sampling (F_s) di dapat pada spesifikasi audio AC97 yang di sediakan oleh board Atlys™ yaitu sebesar 48.000 Hz. Diagram alir pada gambar 4 menunjukkan beberapa tahap dalam merancang filter FIR



Gambar 4. Tahap perancangan filter FIR

Setelah menentukan spesifikasi filter FIR, selanjutnya menghitung *response impulse* filter dalam keadaan ideal. Persamaan ... menunjukkan penghitungan response impulse filter FIR dalam keadaan ideal :

$$h_d(n) = \frac{\sin[\omega_c(n-\alpha)]}{\pi(n-\alpha)}$$

(1)

Lalu setelah didapat response impulse ideal filter FIR, dihitung fungsi windowing ($W(n)$). Penulis menggunakan 3 metode windowing yaitu Hamming window, Blackman window dan Hanning window. Yang mana masing-masing fungsi window ditunjukkan oleh Persamaan 2, Persamaan 3 dan Persamaan 4

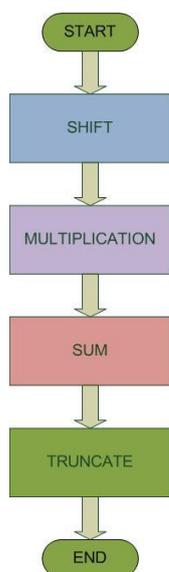
$$w(n) = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi n}{M-1}\right), & 0 \leq n \leq M-1 \\ 0 & \text{lainnya} \end{cases} \quad (2)$$

$$w(n) = \begin{cases} 0.42 - 0.5 \cos\left(\frac{2\pi n}{M-1}\right) + 0.08 \cos\frac{4\pi n}{M-1}, & 0 \leq n \leq M-1 \\ 0 & \text{lainnya} \end{cases} \quad (3)$$

$$w(n) = \begin{cases} 0.5 \left[1 - \cos\left(\frac{2\pi n}{M-1}\right)\right], & 0 \leq n \leq M-1 \\ 0 & \text{lainnya} \end{cases} \quad (4)$$

Dari kedua perhitungan di atas, didapat respon impulse aktual ($H(n)$) dari sebuah filter dengan mengalikan respon impulse ideal ($H_d(n)$) dan fungsi masing windowing ($W(n)$).

Setelah mendapatkan koefisien response impulse, maka di lanjutkan dengan merancang algoritma dari filter FIR.



Gambar 5. Input filter pada 18KHz

III. PENGUJIAN DAN ANALISIS

Pengujian dan analisis dilakukan untuk menganalisis performansi FPGA yang telah tertanam algoritma filter FIR.

A. Pengujian Device FPGA

Pengujian Device FPGA ini bertujuan untuk melihat seberapa banyak penggunaan komponen yang terdapat pada FPGA. Komponen yang akan dilihat antara lain adalah Slices registers, LUT, IOB, BUFG/BUFGMUX, dan DSP48A1. Pengujian ini dilakukan dengan cara simulasi oleh software ISE Xilinx Design Suite. Program yang telah dibuat di *Synthesize* kemudian di *Mapping, placing, dan routing*.

Pengujian dilakukan sebanyak 3 kali, yaitu pada Filter FIR metode hamming, blackman dan kaiser. Kemudian dilihat perbedaan banyaknya penggunaan komponen di *device* FPGA. Hasil pengujian device FPGA secara berturut-turut ditunjukkan dalam tabel 1, tabe 2, tabel 3

TABEL 1
HASIL PENGUJIAN DEVICE FPGA
METODE HAMMING WINDOW

Selected Device : XC6SLX45	Digunakan	Jumlah tersedia	Persen penggunaan
Number of Slice Registers	4.293	54.576	7%
Number of Slice LUTs	2.520	27.288	9%
Number of MUXCYs	1.896	13.644	13%
Number of bonded IOBs	15	218	6%
Number of BUFG/BUFGMUXs	4	16	25%
Number of DSP48A1s	50	58	86%

TABEL 2
HASIL PENGUJIAN DEVICE FPGA

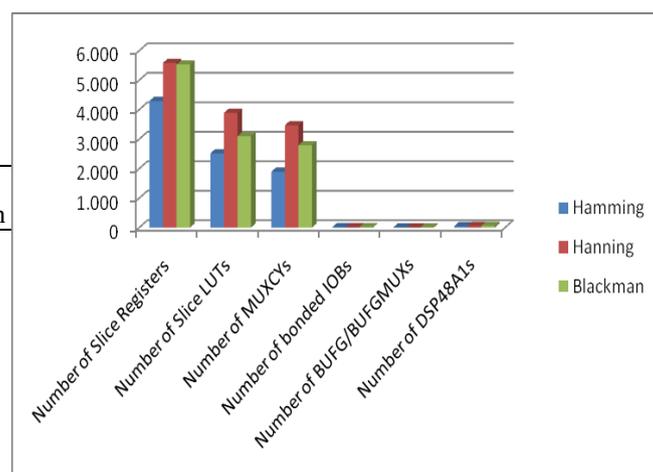
METODE BLACKMANWINDOW

Selected Device : XC6SLX45	Digunakan	Jumlah tersedia	Persen penggunaan
Number of Slice Registers	5.529	54.576	10%
Number of Slice LUTs	3.099	27.288	11%
Number of MUXCYs	2.792	13.644	20%
Number of bonded IOBs	15	218	6%
Number of BUFG/BUFGMUXs	4	16	25%
Number of DSP48A1s	58	58	100%

TABEL 3
HASIL PENGUJIAN DEVICE FPGA
METODE HANNING WINDOW

Selected Device : XC6SLX45	Digunakan	Jumlah tersedia	Persen penggunaan
Number of Slice Registers	5.585	54.576	10%
Number of Slice LUTs	3.889	27.288	14%
Number of MUXCYs	3.472	13.644	25%
Number of bonded IOBs	15	218	6%
Number of BUFG/BUFGMUXs	4	16	25%
Number of DSP48A1s	58	58	100%

Berdasarkan hasil pengujian yang dilakukan, pemakaian komponen yang ada di FPGA terbesar adalah dengan menggunakan metode *Hanning*, dan yang paling hemat penggunaannya adalah dengan metode *Hamming*.



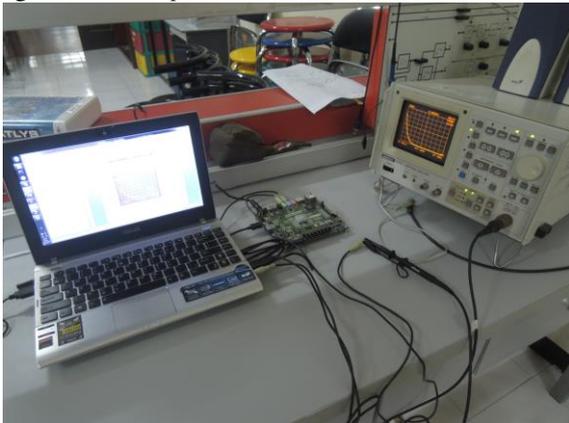
Gambar 6. Grafik perbandingan penggunaan komponen FPGA

B. Pengujian Output Filter FIR

Pengujian Output Filter bertujuan untuk melihat sinyal output yang telah melalui pemrosesan filter FIR dan kemudian dibandingkan dengan sinyal input.

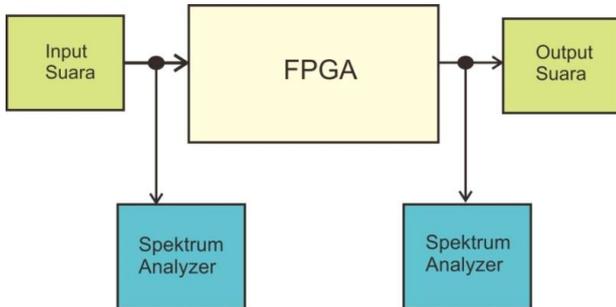
Sebelum melakukan pengujian output, dilakukan pembuatan sinyal input. Sinyal input berupa sinyal sinusoida dengan frekuensi 20-22.000 Hz

Setelah pengujian sinyal input, maka dilakukan pengujian sinyal output. Pengujian sinyal output dilakukan dengan cara memasukkan sinyal input ke FPGA kemudian output dari FPGA disambungkan dengan *Oscilloscope*



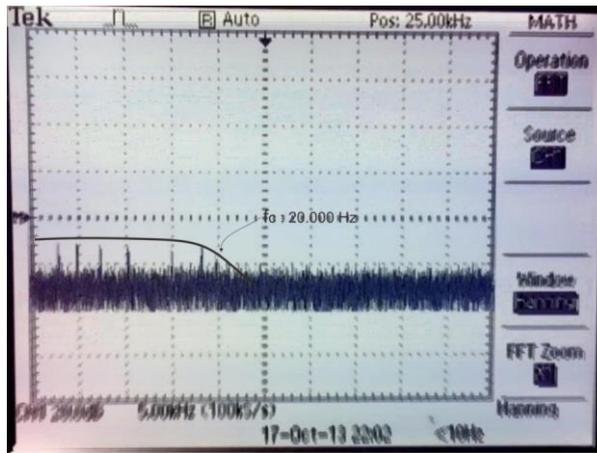
Gambar 9. Pengujian output filter FIR

Diagram blok pengujian sinyal output ditunjukkan oleh Gambar 7

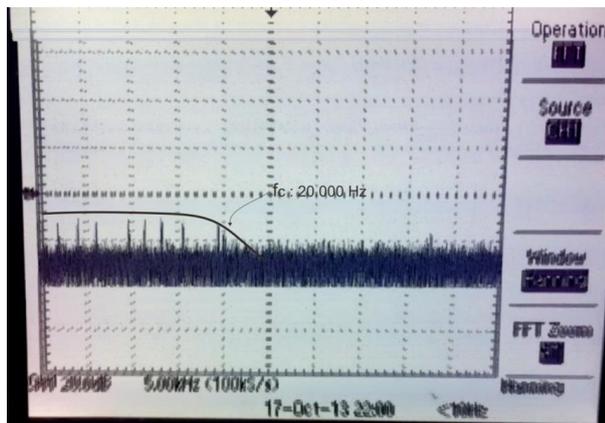


Gambar 7. Diagram Blok Pengujian Sinyal Output

Hasil pengujian sinyal output dengan metode Hamming window, Hanning window, dan Blackman window secara berturut-turut ditunjukkan gambar 8, gambar 9, dan gambar 10.



Gambar 9. Output filter metode Hanning window



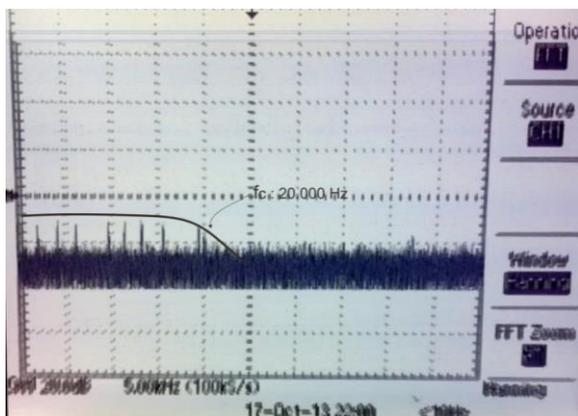
Gambar 10. Output filter metode Blackman window

Dari pengujian output filter dapat dilihat bahwa filter digital FIR telah dapat diimplementasikan pada FPGA., dengan frekuensi cut-off 20.0000 Hz sesuai dengan spesifikasi perancangan filter. Karena pada perancangan ini, penulis menggunakan low-pass filter, maka sinyal dengan frekuensi diatas frekuensi cut-off akan diredam, dan sinyal dengan frekuensi dibawah frekuensi cut-off akan dilewatkan.

IV. KESIMPULAN DAN SARAN

Berdasarkan pengujian dan analisis implementasi filter FIR pada FPGA, dapat ditarik kesimpulan sebagai berikut:

- A. Algoritma filter FIR telah dapat di implementasikan pada FPGA dan output filter telah sesuai dengan spesifikasi yaitu dengan frekuensi cut-off 20KHz, ini dapat dilihat dari pengujian output filter bahwa ketika filter digital FIR diberi masukan sinyal sinusoida dengan frekuensi 20 – 22.000 Hz, maka sinyal yang diatas 20.000 Hz (frekuensi cut-off) akan diredam. Hal ini menunjukkan kinerja dari filter FIR, baik dengan metode Hamming window, Blackman window maupun Hanning window, filter akan meloloskan sinyal yang frekuensi nya lebih rendah dari frekuensi cut-off yaitu sebesar 20KHz, dan akan meredam frekuensi diatasnya.
- B. Dari ketiga metode *Windowing* yang penulis



Gambar 8. Output filter metode Hamming window

pakai pada percobaan disimpulkan bahwa pemakaian komponen yang ada di FPGA terbesar adalah dengan menggunakan metode *Hanning*, yaitu penggunaan Slice Registers sebesar 10%, slice of LUTs 14%, MUXCYs 25%, bonded IOBs 6% BUFG/BUFGMUXs 25% dan DSP48A1 100%. dan yang paling hemat penggunaannya adalah dengan metode Hamming dengan penggunaan Slice Registers sebesar 7%, slice of LUTs 9%, MUXCYs 13%, bonded IOBs 6% BUFG/BUFGMUXs 25% dan DSP48A1 86%

6.1. Saran

Saran-saran dalam pengimplementasian maupun peningkatan unjuk kerja sistem ini dapat diuraikan sebagai berikut:

- A. Untuk mengimplementasikan filter dengan orde yang lebih tinggi, penulis menyarankan untuk menggunakan FPGA yang lain.
- B. Frekuensi audio yang digunakan sebagai input terlalu rendah. Untuk penggunaan FPGA yang maksimal bisa digunakan frekuensi yang lebih tinggi seperti frekuensi radio.

V. DAFTAR PUSTAKA

- Baese, U. Meyer. 2007. *Digital Signal Processing with Field Programmable Gate Arrays*, New York : Springer.
- Datasheet of LM4550. National Semiconductor. www.national.com
- Mooniarsih, Neilcy T. 2010. *Desain dan Simulasi Filter FIR Menggunakan Metode Windowing*, Jurnal ELKHA Vol.2, No.1 : 41-47.
- Putra, Agfianto Eko. 2012. *DSP & Embedded Electronics* http://agfi.staff.ugm.ac.id/pdf/sampel_tapis.pdf. (diakses 26 Maret 2012).
- Smith, Steven W. 1999. *The Scientist and Engineer's Guide to Digital Signal Processing*. California: California Technical Publishing.
- Sudarmilah, Endah. *Implementasi Filter Digital Finite Impulse Response Metode Penjendelaan Hamming pada DSP* http://eprints.ums.ac.id/2/1/Emitor_EDS_ImplementasiFilterDigital.pdf (diakses 26 Maret 2013)
- Storey, Tony and Scott Larson. 2012. *AC'97 Codec Hardware Driver Example*. <http://eewiki.net/display/LOGIC/AC%2797+Codec+Hardware+Driver+Example>, (diakses 16 Mei 2013)
- Wang, Yishu. 2005. *Implementation of Digital Filter by Using FPGA*,. Unpublished Thesis of Bachelor. Bentley: Curtin University of Technology.
- Wilson, Peter R. 2007. *Design Recipes for FPGAs*. United Kingdom : Newnes