

Implementasi Sistem Penghilang Derau Adaptif Menggunakan Algoritma LMS pada FPGA Altera Flex10KLC84

A. Eko Putra dan S. Noor Azhar

Abstract— *The Adaptive Noise Cancelling System has been implemented in FPGA's Altera Flex10KLC84, chip which has 576 LE and 3 EAB. The system has data communication capability between PC and system and the baudrate is 9,600bps. Data format using 8-bit data width two's complement integer and 8 scale factor. Total resources which has been used is 564 LE and 2 EAB, after optimization. The accuracy is 100% according to the MATLAB results for the same computation equation. System respond for square wave is better than sinusoidal wave.*

Keywords: *Adaptive Filter, LMS, FPGA.*

1. PENDAHULUAN

Menurut Douglas (1999), sebuah tapis adaptif adalah sebuah 'alat' komputasi yang mampu memodelkan hubungan antara dua sinyal secara *real-time* (waktu nyata) dan secara iterasi (berulang-ulang). Tapis adaptif biasanya merupakan algoritma atau program yang dijalankan oleh mikroprosesor atau komputer atau emping PSD (Pemrosesan Sinyal Digital) atau bahkan dapat diimplementasikan dalam bentuk gerbang-gerbang logika di dalam sebuah FPGA (*Field Programmable Gate Array*)[1].

Sebuah tapis adaptif didefinisikan dengan empat aspek, yaitu: (1) Sinyal yang diproses oleh tapis, (2) struktur yang mendefinisikan bagaimana sinyal keluaran dihasilkan dari sinyal masukan, (3) parameter-parameter di dalam struktur yang dapat diubah secara iteratif untuk mengubah hubungan masukan-keluaran; serta (4) algoritma adaptif yang menentukan bagaimana parameter-parameter tersebut diubah dari waktu ke waktu[1].

Dengan memilih struktur tapis adaptif tertentu, maka perlu juga dipilih berapa parameter, dan tipe parameter yang akan dilibatkan (yang nilainya bisa berubah-ubah, menyesuaikan sepanjang waktu). Algoritma adaptif digunakan untuk memperbaharui nilai parameter-parameter tersebut sedemikian hingga diperoleh kriteria ralat seminimal mungkin dan biasanya dalam bentuk prosedur optimasi.

Algoritma LMS atau *Least Mean Square* digunakan dalam tapis adaptif untuk menemukan koefisien-koefisien tapis yang berkaitan dengan hasil rerata kuadrat terkecil (LMS) dari sinyal ralat $e(n)$. Metode ini merupakan jenis metode penurunan gradien stokastik (*stochastic gradient descent*) yaitu metode adaptasi tapis yang hanya berdasar dari ralat saat itu. Ditemukan pada tahun 1960 oleh seorang guru besar Universitas Stanford bernama Bernard Widrow dan mahasiswa Ph.D. pertamanya Ted Hoff[2].

Penelitian ini bertujuan melakukan implementasi penapis adaptif untuk aplikasi penghilang derau menggunakan algoritma LMS ke dalam emping FPGA, sebagai antarmuka dan sekaligus bahan perbandingan dengan komputasi numerik dari sisi perangkat lunak digunakan skrip Matlab.

FPGA merupakan salah satu jenis *integrated circuit* (IC) yang populer yang dapat digunakan untuk membuat purwarupa sebuah sistem digital. FPGA sendiri sudah secara luas digunakan dalam berbagai penelitian, salah satu jenis FPGA yang digunakan dalam penelitian ini adalah seri keluarga Flex10K dari Altera.

Matlab digunakan sebagai antarmuka, *simulator* dan sekaligus penghasil contoh data untuk diolah FPGA yang kemudian hasilnya akan ditampilkan lagi melalui Matlab. Penelitian ini dilakukan untuk membuktikan apakah rancangan tapis adaptif dengan algoritma LMS dapat diimplementasikan pada FPGA Altera seri Flex10KLC84.

Agfianto Eko Putra, Prodi. Elektronika dan Instrumentasi, Jurusan Ilmu Komputer dan Elektronika, FMIPA, UGM, Email: agfi@ugm.ac.id

Susaf Noor Azhar, Prodi. Elektronika dan Instrumentasi, Jurusan Ilmu Komputer dan Elektronika, FMIPA, UGM, Email: azhar_5301679@yahoo.com

2. METODE PENELITIAN

Penelitian mengenai struktur rancangan tapis yang lebih baik dilakukan beberapa peneliti, seperti Perwaiz dan Khan[3] yang melakukan penelitian tentang *LMS bitstream*, dimana menggunakan optimasi *bitstream* untuk meningkatkan efisiensi dan mengurangi penggunaan sumber daya dalam operasi perkalian dibandingkan pada operasi perkalian secara konvensional. Allred dan kawan-kawan[4] membuat sebuah tapis adaptif tanpa menggunakan unit pengali dengan menggantinya dengan *look-up-table* (LUT) berdasarkan metode *distributed arithmetic*, sehingga bisa mempercepat proses.

Implementasi-implementasi tapis adaptif juga telah dilakukan beberapa peneliti. Misalnya, penelitian Dornean dan kawan-kawan[5] yang melakukan penelitian tentang sistem identifikasi menggunakan algoritma LMS. Penelitian lain juga dilakukan oleh Elhossini[6] sebagai *Adaptive Noise Cancelling* (ANC) dalam pengolahan data audio. Dalam penelitiannya dilakukan perbandingan antara implementasi perangkat lunak secara penuh dengan implementasi perangkat keras secara penuh, juga dengan implementasi gabungan keduanya. Sedangkan implementasi untuk pemrosesan video dilakukan oleh Turney dan kawan-kawan[7] yang menggunakan algoritma Kalman dan Wiener untuk membuat penapis data video secara *realtime*. Selanjutnya Mohammed[8], melakukan implementasi ANC berdasarkan *Adaptive Line Enhancer* (ALE) dengan algoritma *Normalized Least Mean Square* (NLMS). Implementasi dasar tapis adaptif dengan algoritma LMS dilakukan oleh Meyer-Bease dalam FPGA sebagai pengidentifikasi sistem. Yang terakhir El-Tarhuni dan Sheikh[9] melakukan penelitian implementasi algoritma LMS untuk akuisisi *Direct-Sequence Spread-Spectrum Code*.

Penggunaan FPGA Altera Flex10KLC84 dalam penelitian yang terkait juga telah dilakukan beberapa peneliti. Misalnya, implementasi operasi *floating point* oleh Putra dan Rifa'i[10], yang menunjukkan bahwa rancangan unit aritmatika menggunakan operasi *floating point* dapat dilakukan dalam Flex10K dengan melakukan optimasi rancangan, dengan memakai hampir semua sumber daya FPGA. Tanpa optimasi maka sumber daya tidak mencukupi untuk implementasi rancangan operasi *floating point*. Penelitian implementasi

sistem komunikasi serial dalam FPGA dengan komputer dilakukan oleh Fakhurozi[11]. Sedangkan implementasi sebagai tapis FIR 16 titik dilakukan oleh Santosa[12].

Berdasarkan berbagai kajian terhadap penelitian yang telah dilakukan, belum ditemukan percobaan implementasi tapis adaptif berbasis algoritma LMS dengan sistem bilangan bulat atau *integer* sebagai penghilang derau pada FPGA Flex10KLC84 serta menggunakan Matlab sebagai antarmuka dan komunikasi serial untuk melakukan pertukaran data (antara FPGA dengan Matlab).

3. RANCANGAN ALAT

Algoritma LMS perlu disesuaikan untuk menangani operasi *integer*. Nilai koefisien perlu dipecah menjadi tiga bagian, sehingga menjadi $\mu = \mu_1 \cdot \mu_2 \cdot \mu_3$. Menggunakan sifat komutatif dari perkalian, maka letak μ_1 , μ_2 dan μ_3 dapat dipindahkan, sehingga algoritma LMS menjadi seperti persamaan 1 hingga 4.

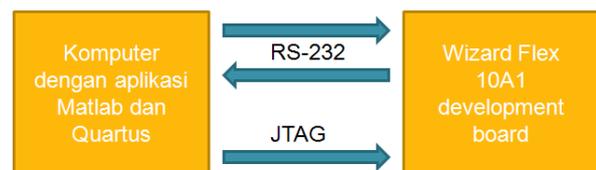
$$x[n] = \mu_1 \cdot x_{in}[n]; \quad d[n] = d_{in}[n] \quad (1)$$

$$y[n] = \sum_{k=0}^{L-1} f_k[n] \cdot x[n-k] \cdot \mu_3 \quad (2)$$

$$e[n] = d[n] - y[n] \quad (3)$$

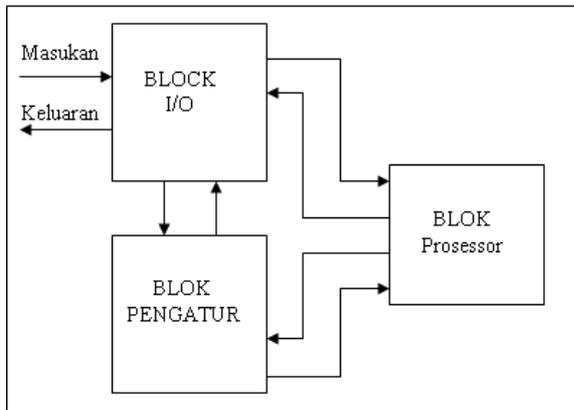
$$f_k[n+1] = f_k[n] + \mu_2 \cdot e[n] \cdot x[n-k] \quad (4)$$

Adapun besarnya nilai yang optimal adalah $\mu_1 = 1/22$, $\mu_2 = 1/27$ dan $\mu_3 = 1/210$. Skema sistem secara keseluruhan ditunjukkan pada Gambar 1, komputer terhubung dengan FPGA melalui komunikasi serial dan JTAG. Komunikasi serial RS-232 berfungsi sebagai saluran data, sedangkan JTAG berfungsi untuk mengkonfigurasi FPGA.



Gambar 1. Skema Sistem Penghilang Derau Adaptif Berbasis FPGA

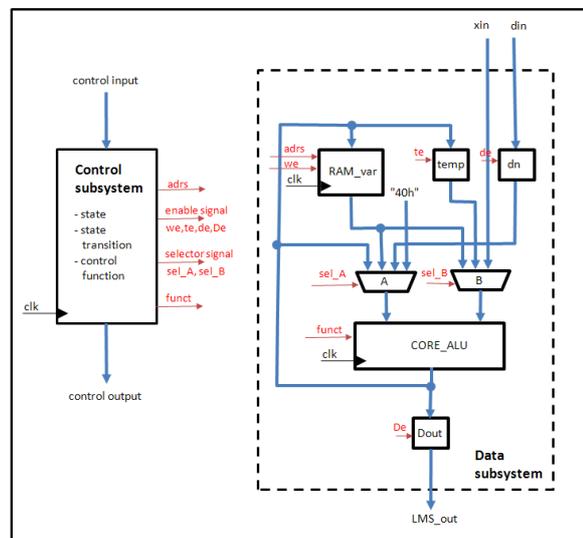
Adapun skema top level sistem dalam FPGA dapat dilihat pada Gambar 2. Dimana dalam FPGA akan seperti Gambar 3. Hierarki, atau struktur sistem dapat dilihat pada Tabel 1. Setiap entitas memiliki fungsi masing-masing. Rancangan prosesor pengolah sinyal digital dirancang secara RTL seperti pada Gambar 4. Dalam prosesor, urutan atau langkah-langkah perhitungan dilakukan sesuai dengan Gambar 5. Karena menggunakan sistem *unimodule*, maka operasi aritmatika yang bisa dilakukan dalam setiap prosesnya menjadi terbatas dan harus dibagi menjadi beberapa langkah. Urutan tersebut kemudian dikelompokkan menjadi kondisi atau *state*, seperti mesin kondisi pada Gambar 6. Tabel 2 dan Tabel 3 memberikan penjelasan proses dalam setiap *state*.



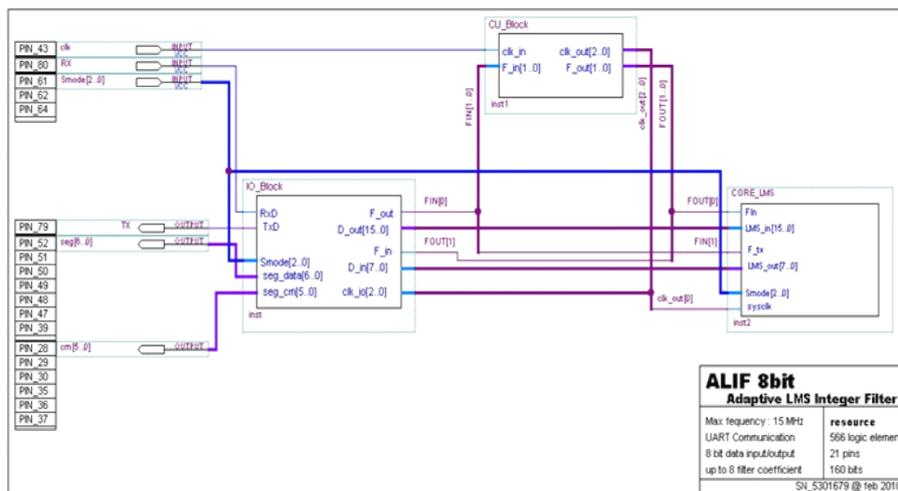
Gambar 2. Arsitektur Sistem Dalam FPGA

Tabel 1. Hierarki Sistem

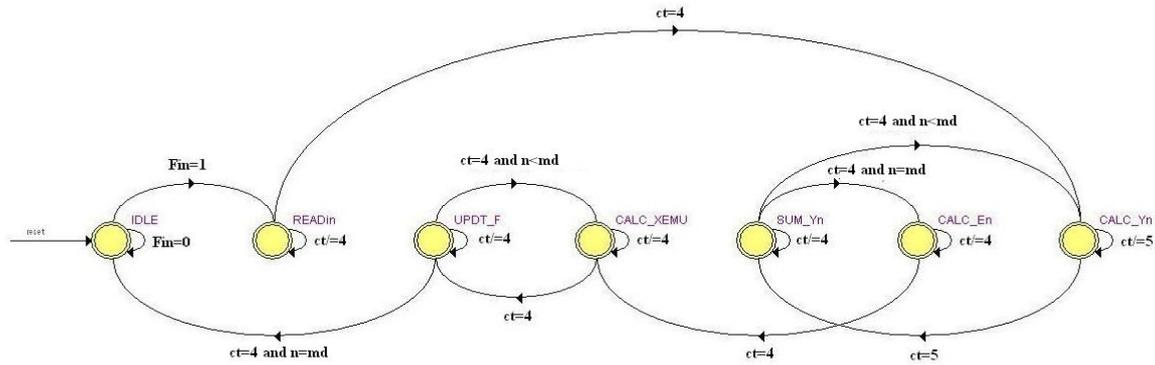
Nama file	Jenis file
Adaptif_LMS (TOP LEVEL)	Skematik
Core_LMS	VHDL : Structural
CORE_ALU	Skematik
LMSMult	VHDL : Wizard
LMSAdd5	VHDL : Wizard
LMSDiv	VHDL : Behavioral
Mux16	VHDL : Wizard
RAM_var	VHDL : Wizard
Adrs_FIFO	Skematik
IO_BLOCK	Skematik
UART_Rec	VHDL : Behavioral
UART_Trans	VHDL : Behavioral
decoder	VHDL : Behavioral
CU_BLOCK	Skematik
clock_gen	VHDL : Behavioral
SYNC	Skematik



Gambar 4. Skema prosesor dalam tingkat register (RTL) CORE_LMS



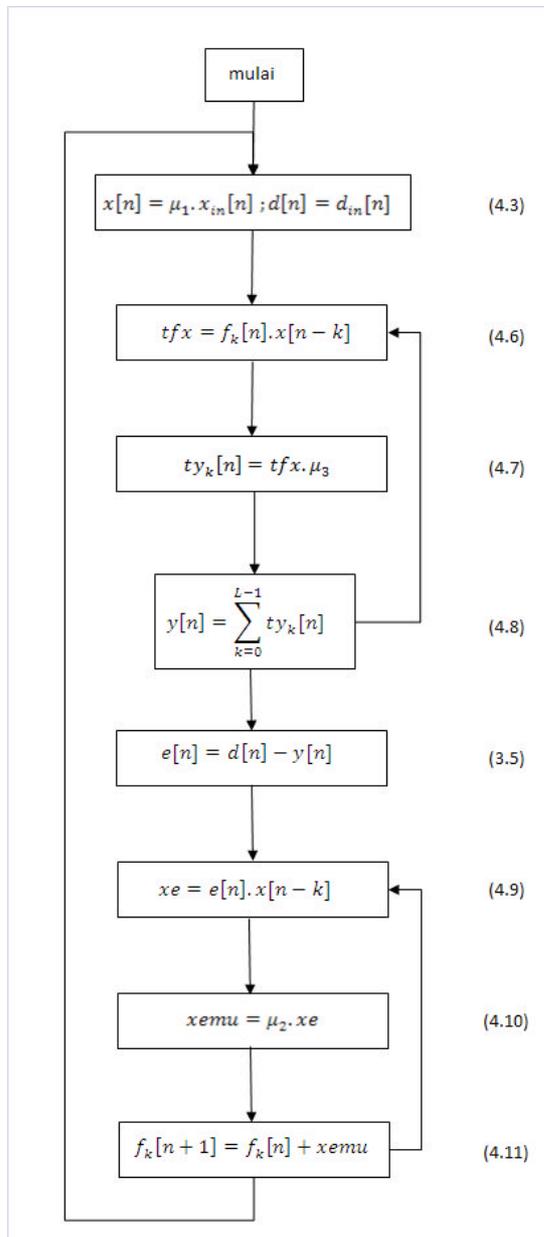
Gambar 3. Skema rancangan sistem tingkat teratas (top level)



Gambar 6. Diagram alir eksekusi

Tabel 2. Transisi mesin kondisi pada CORE_LMS

State	Masukan			Next State
	Ct	Fin	n	
IDLE	X	1	X	IDLE
	X	0	X	READin
READin	< 4	X	X	READin
	= 4	X	X	CALC_Yn
CALC_Yn	< 5	X	X	CALC_Yn
	= 5	X	X	SUM_Yn
SUM_Yn	< 4	X	X	SUM_Yn
	= 4	X	n < md	CALC_Yn
	= 4	X	n = md	CALC_En
CALC_En	< 4	X	X	CALC_En
	= 4	X	X	CALC_Xemu
CALC_Xemu	< 4	X	X	CALC_Xemu
	= 4	X	X	UPDFT_F
UPDFT_F	< 4	X	X	UPDFT_F
	= 4	X	n < md	CALC_Xemu
	= 4	X	n = md	IDLE



Gambar 5. Diagram alir eksekusi

4. HASIL DAN PEMBAHASAN

Berdasarkan hasil kompilasi dan simulasi dalam Quartus didapatkan data-data sebagaimana ditunjukkan pada Tabel 3 dan Tabel 4. Tabel 3 menunjukkan pewaktuan dalam sistem, sedangkan Tabel 4 menunjukkan analisis penggunaan sumber daya FPGA. Dari kedua tabel tersebut terlihat bahwa sistem dapat diimplementasikan karena tidak melebihi kapasitas dan pewaktuan maksimumnya.

Analisis keluaran sistem ditunjukkan dalam Gambar 7 hingga Gambar 10. Gambar-gambar tersebut menunjukkan perbandingan deviasi E(n) untuk implementasi dalam FPGA dan dalam Matlab. Dari gambar-gambar tersebut terlihat bahwa kedua keluaran berhimpit atau selisihnya sama dengan 0 (nol). Pada pengujian dengan tipe data gelombang kotak didapatkan tanggap dan akurasi yang lebih baik dibandingkan dengan tipe data gelombang sinusoidal. Demikian halnya dengan perbandingan jumlah koefisien tapis yang

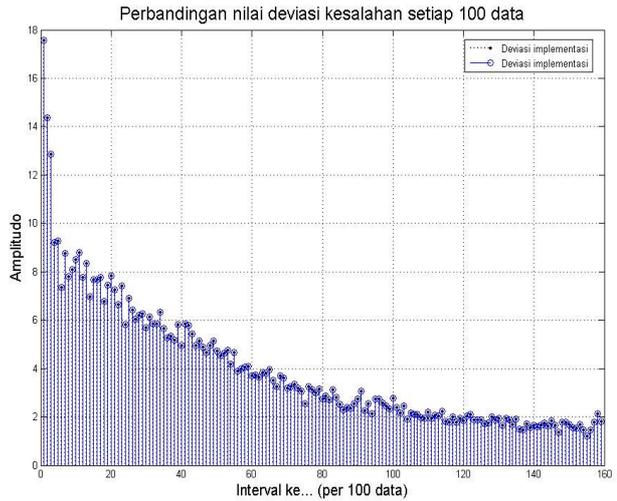
dipakai, semakin banyak jumlah koefisien yang dipakai, semakin besar tanggap dan tingkat kesalahannya.

Tabel 3. Analisis peawatan sistem

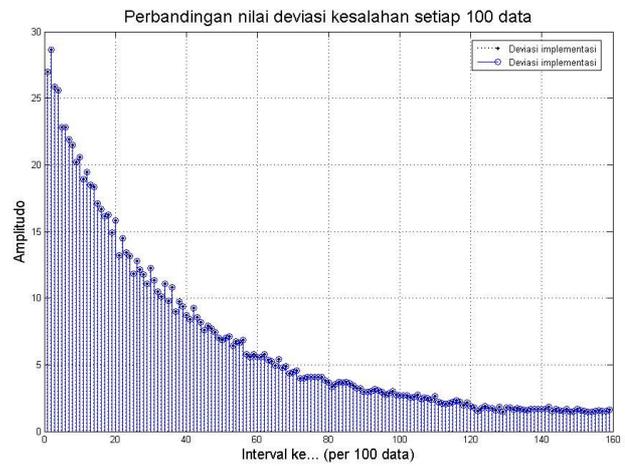
Tipe	Waktu
tsu	20,300nd
tco	24,100nd
tpd	18,500nd
th	4,200nd
Frekuensi maks.	10,53MHz (95,000nd)

Tabel 4. Rangkuman penggunaan sumber daya

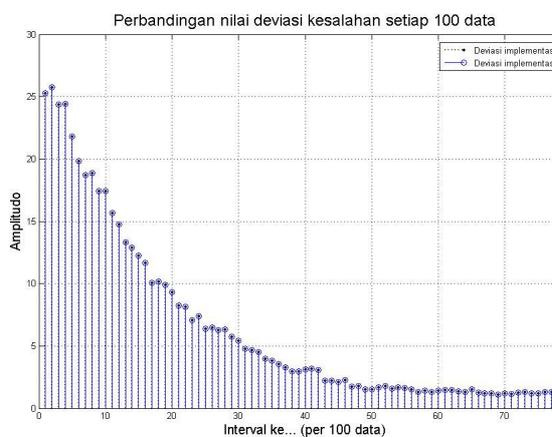
Sumber daya	Penggunaan
Total logic elements	564 / 576 (98 %)
Registers	195 / 576 (34 %)
Logic elements in carry chains	40
User inserted logic elements	0
I/O pins	20 / 59 (34 %)
clock pins	1
dedicated input pins	0 / 4 (0 %)
Global signals	2
EABs	2 / 3 (67 %)
Total memory bits	160 / 6,144 (3 %)
Total RAM block bits	4,096 / 6,144 (67 %)
Maximum fan-out node	CU_Block:inst1 clock_gen:inst1 lpm_counter:cont1_rtl_0 alt_counter_f10ke:wysi_counter counter_cell[1]
Maximum fan-out	185
Highest non-global fan-out signal	CORE_LMS:inst2 ADRS[3]
Highest non-global fan-out	103
Total fan-out	2025
Average fan-out	3.41



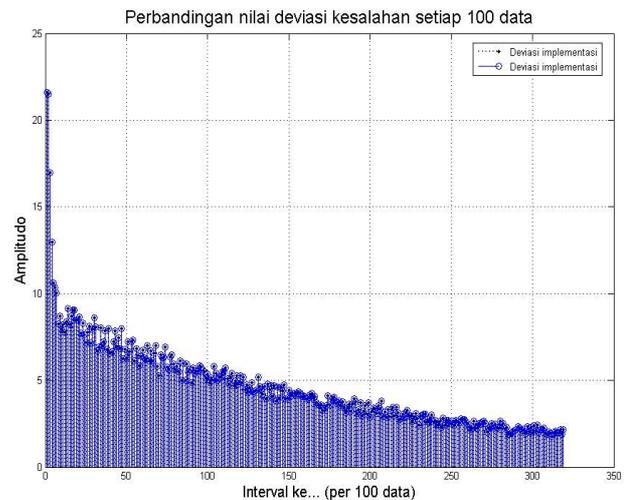
Gambar 8. Grafik deviasi, pengujian 2 (tipe data kotak, 8 koefisien tapis)



Gambar 9. Grafik deviasi, pengujian 3 (tipe data kotak, 8 koefisien tapis)



Gambar 7. Grafik deviasi, pengujian 1 (tipe data kotak, 1 koefisien tapis)



Gambar 10. Grafik deviasi, pengujian 4 (tipe data kotak, 8 koefisien tapis)

5. KESIMPULAN

Dari penelitian yang dilakukan didapat kesimpulan sebagai berikut: Sistem berhasil diimplementasikan dalam FLEX10K dengan total penggunaan 564 Logic Element, 195 register dan 160 bit memory. dengan pengaturan optimasi area. Implementasi desain algoritma dalam prosessor menggunakan 466 Logic Element, 137 Register, 2 EAB dan 160 bit memory. Durasi proses prosessor sebanding dengan jumlah koefisiennya sebesar L dengan persamaan $(9+17*L)*0.417\mu s$. Akurasi keluaran implementasi algoritma integer LMS pada FPGA terhadap simulasi dalam Matlab adalah 100%. Sistem memiliki tingkat akurasi dan respon yang lebih tinggi untuk data gelombang kotak dibandingkan data sinusoidal.

DAFTAR PUSTAKA

- [1] Douglas, S.C., 1999, Introduction to Adaptive Filters, Digital Signal Processing Handbook, CRC Press.
- [2] Haykin, S dan Widrow, B. (editor), 2003, Least-Mean-Square Adaptive Filters, John-Wiley & Sons.
- [3] Perwaiz, A., Khan, S.A., 2008, LMS Bit Stream, Institute of Electrical and Electronic Engineers (IEEE), 0-7695-3114-8/08.
- [4] Allred, D.J., Huang, W., Krishnan, V. Yoo, H., Anderson, D.v., 2004, An FPGA Implementation for High Throughput Adaptive Filter Using Distributed Arithmetic. *Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM'04)*.
- [5] Dornean, I., Topa, M., Kirei, B.S., Szopos, E., 2007, System Identification With Least Mean Square Adaptive Algorithm, "Interdisciplinarity In Engineering" Scientific International Conference, Romania.
- [6] Elhossini, A., Areibi, S., Dony, R., 2006, An FPGA Implementation of the LMS Adaptive Filter for Audio Processing, *Institute of Electrical and Electronic Engineers (IEEE)*, 1-4244-0690-0/06.
- [7] Turney, R.D., Reza A.M., Delva J.G.R. FPGA Implementation of Adaptive Temporal Kalman Filter for Real Time Video Filtering, *Institute of Electrical and Electronic Engineers (IEEE)*, 0-7803-5041-3/99/1999
- [8] Mohammed, J.R., 2007, A New Simple Adaptive Noise Cancellation Scheme Based On ALE and NLMS Filter, *Institute of Electrical and Electronic Engineers (IEEE)*, 0-7695-2835-X/07
- [9] El-Tarhuni , M.G. , Sheikh A.U., Application of Adaptive Filtering to Direct-Sequence Spread-Spectrum Code Acquisition, *Wireless Personal Communications*, 8:185-204/1998.
- [10] Putra, A.E., Rifa'i, I.N., 2009 , Floating-point Unit Implementation in Altera FLEX10K FPGA using VHDL, *Proceeding of The 1st International Seminar on Science and Technology 2009 – UII*, Yogyakarta.
- [11] Fakhurrozi, M.E., 2007, Prototipe UART Berbasis FPGA FLEX10K Menggunakan Bahasa Deskripsi VHDL, *Skripsi SI, Elektronika dan Instrumentasi Universitas Gadjah Mada*.
- [12] Santosa, E.B., 2008 Perancangan Prosessor 16 titik Fast Fourier Transform Menggunakan Deskripsi VHDL berbasis FPGA, *Skripsi SI, Elektronika dan Instrumentasi Universitas Gadjah Mada*.