

Implementasi Programmable DAC pada FPGA Xilinx Spartan-6 Berbasis VHDL

Ahmad Haneef Zuhdy*¹

¹Prodi Elektronika dan Instrumentasi, FMIPA UGM, Yogyakarta
e-mail: *¹ahmadhaneef@mail.ugm.ac.id

Abstrak

Perkembangan desain digital telah mengalami kemajuan yang pesat. Hal ini memungkinkan untuk mengimplementasikan sebuah desain sistem digital dengan banyak variasi basis sistem. FPGA merupakan salah satu basis sistem yang dapat digunakan dengan keunggulan kecepatan pemrosesan karena desain diimplementasikan hingga tataran perangkat keras. Era saat ini telah mengalami digitalisasi, dimana besaran-besaran analog pun telah dikonversikan untuk disimpan dalam bentuk digital. Fungsi konversi digital ke analog (DAC) menjadi salah satu fungsi yang sangat umum digunakan. Penelitian ini dilakukan untuk mengimplementasikan fungsi DAC ke dalam FPGA dan dapat diprogram oleh pengguna, sehingga sistem ini disebut programmable DAC.

Programmable DAC diimplementasikan pada FPGA Spartan-6 LX45 yang tertanam pada papan pengembangan Atlys. Implementasi menggunakan VHDL untuk mendesainnya. Programmable DAC dirancang untuk melakukan rekonstruksi sinyal menggunakan 2 metode modulasi yaitu PWM dan delta-sigma. Penelitian ini menggunakan MATLAB mengkalkulasi nilai sinyal. Kemudian nilai tersebut dikirim menggunakan USB-UART yang terdapat pada papan pengembang Atlys. Nilai sinyal tersebut diolah oleh programmable DAC yang telah diprogram pada FPGA. Untuk memperhalus luaran digunakan tapis pelewat rendah dari komponen pasif RC. Sinyal kemudian dimunculkan melalui osiloskop. Sinyal hasil rekonstruksi memiliki bentuk dan frekuensi yang serupa. Implementasi menggunakan 1 % sumber daya dari FPGA, tepatnya 75 LUTs dan 65 Slices Register.

Kata kunci—Programmable DAC, FPGA, VHDL, PWM, Delta-Sigma

Abstract

Development of digital design is on rapid progress. This makes it possible to implement a digital system design with many variations of base system. FPGA is a base system that can be used with advantage on speed processing because the design is implemented on hardware level. This current era have been digitalization, where the analogue value have been converted to digital form. Therefore function of digital-to-analogue conversion (DAC) became one of commonly function used. This research implement DAC function into FPGA and can be programmed by users, this system called programmable DAC.

The programmable DAC has been implemented on FPGA Spartan-6 LX45 embedded in Atlys Development Board. This implementation using VHDL to design. Programmable DAC was designed to reconstruct the signal using 2 method ie PWM and delta-sigma. This research uses MATLAB to calculate signal value. Then the data sent using USB-UART on Atlys development board. The signal value was processed by programmable DAC that been programmed on the FPGA. For smoothing the output using low pass filter with passive RC components. Then the Signal was generated through an oscilloscope. The reconstructed signal had same shape and frequency. Implementation uses 1% of the FPGA resources, precisely 75 LUTs and 65 Slices Register.

Keywords—The programmable DAC, FPGA, VHDL, PWM, Delta-Sigma

1. PENDAHULUAN

Dengan kepadatan dan kecepatan kemajuan seperti saat ini seorang digital desainer mampu membuat sistem-sistem digital yang kompleks dengan banyak pilihan platform. Kebanyakan *embededd system* menggunakan mikrokontroler sebagai basisnya. Namun, untuk desain yang membutuhkan logika yang lebih tinggi dan fungsi khusus, mikrokontroler perlu dikombinasikan dengan FPGA (*Field Programmable Gate Arrays*) atau ASSP (*Application Specified Standard Product*) [1].

Field Programmable Gate Arrays atau FPGA merupakan sebuah chip yang berisikan ribuan logic circuits dimana bisa dikonfigurasi untuk menjalankan fungsi-fungsi yang diinginkan. Arsitektur umum FPGA terdiri atas *interconnect technology*, *logic cells*, FPGA memori, standar I/O, sistem detak dan *multipliers* [1].

Tantangan kedepan adalah mengimplementasikan fungsi-fungsi utama atau umum kedalam FPGA. Salah satu fungsi tersebut adalah konverter digital ke analog (DAC). Sudah sangat jarang sebuah sistem yang tidak memanfaatkan fungsi konverter tersebut [2].

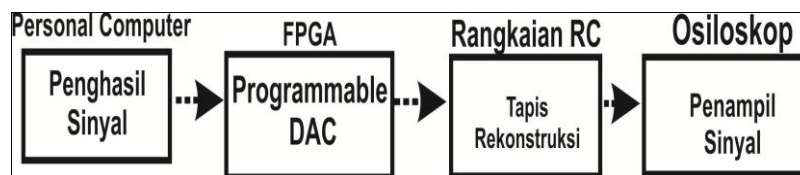
Hingga saat ini terus dikembangkan metode konversi sinyal digital menjadi sinyal analog. Banyak metode DAC baik yang berdasarkan pada rasio resistor seperti *weighted resistor* DAC, *R2R ladder* dan *potentiometric* DAC atau berdasarkan modulasi sinyal seperti PWM (*Pulse Width Modulation*) DAC dan *delta-sigma* DAC [3].

Pengembangan *programmable* DAC bertujuan untuk mengakomodir model konversi digital ke analog terutama berdasarkan modulasi sinyal. *Programmable* DAC memungkinkan pengguna untuk mengkonfigurasi DAC berdasarkan kebutuhan dari segi modulasi (PWM atau *delta-sigma*) dan bit resolusi.

Dengan sifat FPGA yang dapat dikonfigurasi hingga tingkat perangkat keras, diharapkan mampu mengimplementasikan fungsi *programmable* DAC dengan tingkat akurasi yang tinggi.

2. RANCANGAN SISTEM

Penelitian *programmable* DAC ini lebih difokuskan pada perancangan desain digital pada FPGA. Tetapi dalam sebuah sistem DAC berdasarkan pulsa perlu ditambahkan tapis yang berfungsi untuk rekonstruksi dan memperlhalus sinyal. Gambar 1 menunjukkan diagram blok sistem yang saling terhubung antara lain PC, FPGA pada papan pengembang, tapis dari rangkaian RC dan osiloskop.



Gambar 1 Diagram blok rancangan sistem

2.1 Rancangan Penghasil Sinyal

Konversi digital ke analog tentu membutuhkan data sinyal dalam bentuk digital. Prosedur pembentukan dan pengolahan sinyal digital ini menggunakan perangkat lunak yang dapat menghasilkan sinyal-sinyal untuk dianalisa. Dalam penelitian ini menggunakan sinyal sinusoidal dikarenakan sederhana dan mudah untuk diamati.

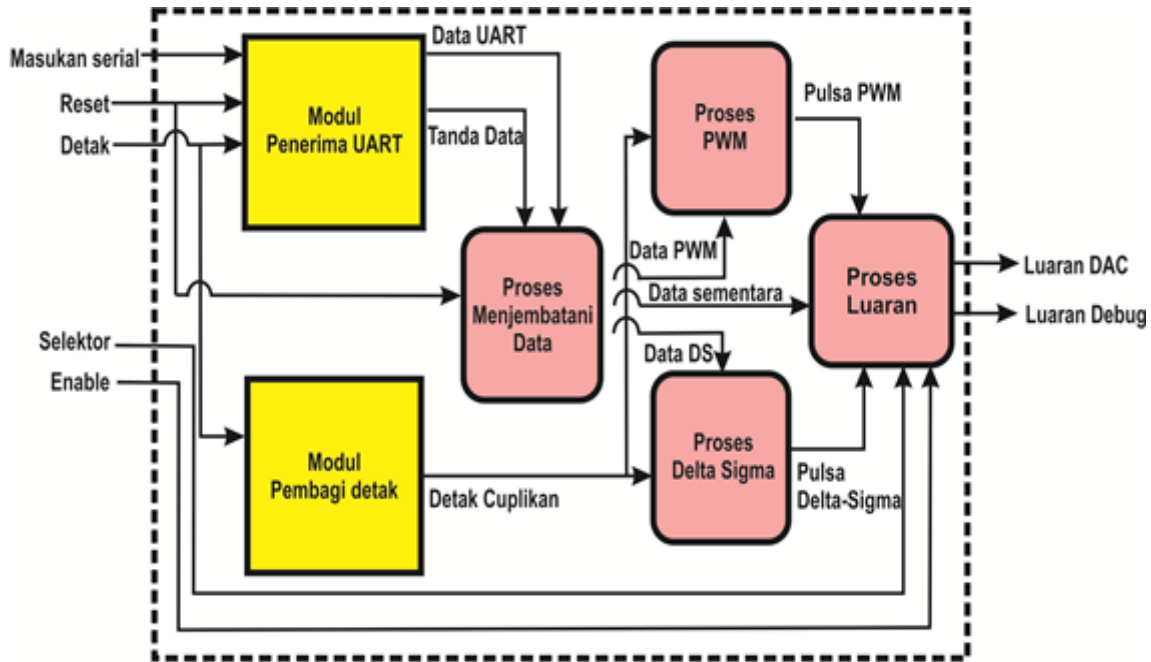
Sinal sinusoidal merupakan sinyal osilasi yang terbentuk dari persamaan sinus yang ditunjukkan oleh persamaan (1). Dimana A adalah amplitudo f adalah frekuensi dan t adalah waktu interval [4].

$$f(x) = A \cdot \sin(2 \cdot \pi \cdot f \cdot t) \quad (1)$$

2.2 Rancangan Programmable DAC

Perancangan *programmable* DAC ini menjadi fokus utama dari penelitian ini. Gambar 2 menunjukkan *top level function* dari *programmable* DAC yang terdiri atas 2 modul dan 4 proses. Untuk komunikasi terdapat 5 masukan, 2 luaran dan 6 sinyal interkoneksi.

2.2.1 Modul Penerima UART

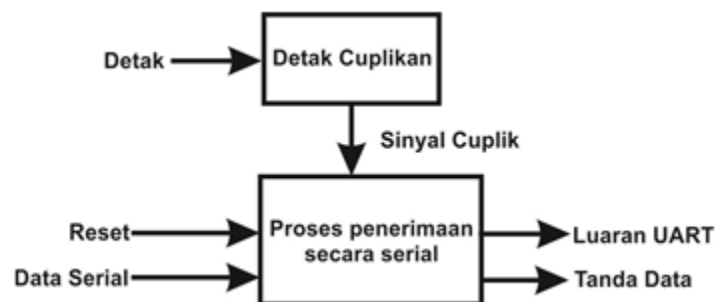


Gambar 2 Diagram blok top level *function* *programmable* DAC

Komunikasi diperlukan untuk mengirim dan menerima data. Disini untuk komunikasi dibuat modul tersendiri sehingga apabila dibutuhkan dapat diubah dengan protokol komunikasi yang lain, untuk penelitian ini digunakan protokol UART.

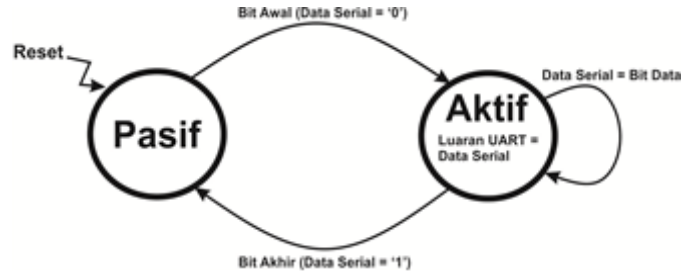
Modul UART penerima merupakan sebuah modul yang berfungsi untuk menerima data secara serial yang dikirimkan port COM dari PC ke FPGA melalui USB-UART. Diagram blok dari modul ini seperti pada Gambar 3.

Detak cuplikan berfungsi untuk mengatur pencuplikan dari sebuah data sinyal yang dikirimkan. Hal ini bertujuan agar sinyal cuplikan tepat pada tiap bit data yang dikirim.



Gambar 3 Diagram blok modul penerima UART

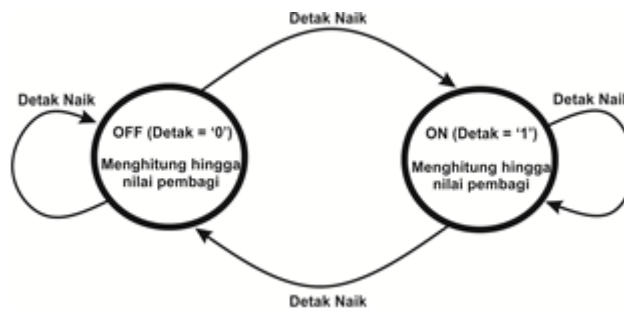
UART bersifat *asynchronous* sehingga tidak dibutuhkan detak sebagai acuan, proses akan mencari bit mulai untuk mulai menerima data dan berakhir ketika bit akhir didapatkan untuk kembali pada kondisi pasif. Proses penerimaan secara serial sesuai dengan diagram keadaan yang terdapat pada Gambar 4.



Gambar 4 Diagram keadaan proses penerimaan secara serial

2.2.2 Modul Pembagi Detak

Pembagi detak digunakan untuk menyesuaikan frekuensi detak agar sesuai dengan yang dibutuhkan oleh sistem. Dalam beberapa kasus tidak perlu dilakukan penyesuaian frekuensi detak sehingga pembagi detak ini dibuat sebagai modul yang dapat diaktifkan atau tidak. Modul pembagi detak dibuat berdasarkan diagram keadaan seperti yang ditunjukkan pada Gambar 5.



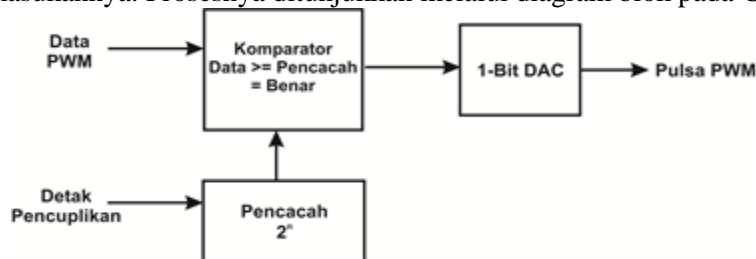
Gambar 5 Diagram keadaan modul pembagi detak

2.2.3 Proses Menjembatani Data

Proses menjembatani berfungsi untuk menyimpan data sementara kemudian diteruskan pada proses modulasi baik PWM dan Delta-sigma. Pada proses ini juga dilakukan pengecekan agar data yang diteruskan sudah sesuai. Pertama dengan mengecek tanda yang kedua dengan mengeluarkan data sementara ke LED agar terlihat nilai data yang akan diproses. Terdapat saklar reset yang berfungsi untuk mereset nilai sementara.

2.2.4 Proses PWM

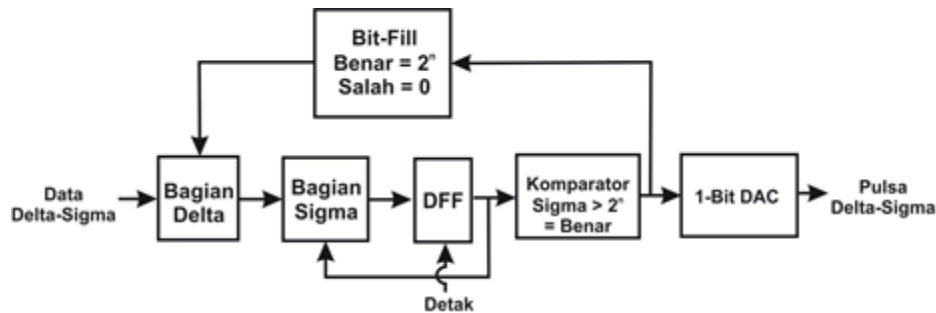
Proses PWM merupakan sebuah modulasi pulsa berdasarkan lebar data. Modulasi ini kemudian akan menghasilkan suatu lebar pulsa dengan *on cycle* dan *off cycle* yang proporsional terhadap nilai masukannya. Prosesnya ditunjukkan melalui diagram blok pada Gambar 6.



Gambar 6 Diagram blok proses PWM

2.2.5 Proses Delta-Sigma

Proses delta-sigma dijalankan secara paralel dengan proses PWM namun secara *synchronous*, artinya detak pencuplikan yang digunakan sama. Sama halnya dengan PWM, modulasi delta-sigma akan menghasilkan pulsa yang proporsional terhadap nilai masukannya. Perbedaannya terletak pada jenis pulsanya, dimana PWM proporsional secara lebar (*width*) sedangkan modulasi delta-sigma akan proporsional secara kerapatan (*density*) oleh sebab itu sering disebut pula sebagai *Pulse Density Modulation* (PDM). Diagram bloknya ditunjukkan pada Gambar 7.



Gambar 7 Diagram blok proses delta-sigma

2.2.6 Proses Luaran

Proses terakhir adalah pengaturan luaran, dimana terdapat selektor untuk memilih luaran dari modulasi PWM atau modulasi delta-sigma yang akan diteruskan. Selain itu terdapat saklar *enable* untuk mengaktifkan proses luaran.

2.3 Rancangan Tapis Rekontruksi

Luaran dari *programmable* DAC masih berupa *bitstream* dan sinyal pulsa *rectangular* dengan modulasi yang proporsional. Maka dibutuhkan sebuah tapis untuk merekonstruksi sinyal tersebut.

Tapis rekonstruksi atau disebut juga *anti-image filter* merupakan sebuah tapis yang berfungsi untuk memperhalus sinyal analog dari luaran digital [5].

Dalam sinyal luaran *programmable* DAC terdapat komponen frekuensi PWM dan frekuensi *delta-sigma* yang lebih tinggi dibandingkan sinyal data [3]. Untuk itu dibutuhkan tapis pelewat rendah dimana frekuensi *cut-off* berada diatas frekuensi sinyal data dan dibawah frekuensi sinyal PWM dan *delta-sigma*. Maka yang diteruskan hanya sinyal data yang merupakan komponen frekuensi rendah. Penentuan frekuensi *cut-off* ini memenuhi persamaan (2). f_p adalah frekuensi PWM dan *delta-sigma*, B adalah bit resolusi.

$$f_c = \frac{f_p}{2^{1+B}} \quad (2)$$

Sebuah tapis pelewat sederhana dapat dibuat dengan menggunakan rangkaian R dan C. Untuk mendapatkan frekuensi *cut-off* maka divariasikan R dan C untuk memenuhi persamaan (3).

$$f_c = \frac{1}{2\pi RC} \quad (3)$$

3. HASIL DAN PEMBAHASAN

Pengujian ditujukan untuk mengetahui apakah *programmable* DAC mampu melakukan fungsi untuk mengkonversikan sinyal digital kedalam bentuk analog. Selain itu mengetahui sejauh mana tingkat rekonstruksi dari sistem ini.

Untuk pengujian sistem, dilakukan langkah-langkah sebagai berikut:

1. Penghasil sinyal menggunakan akan menghasilkan sinyal sinusoidal dengan variasi frekuensi seperti pada Tabel 1.
2. Melakukan uji simulasi fungsional dari *programmable* DAC menggunakan perangkat lunak ISim yang terdapat pada ISE Webpack.
3. Implementasi *programmable* DAC yang terdiri atas proses sintesis dari VHDL ke RTL dan konfigurasi pada FPGA Spartan-6 LX45, laporan dan analisis sistem akan dibuat oleh ISE.
4. Pengamatan sinyal hasil rekonstruksi menggunakan osiloskop.

Tabel 1 Variasi frekuensi sinyal sinusoidal

Variasi	Amplitudo	Bit Resolusi	Frekuensi
1	1	8	10 Hz
2	1	8	50 Hz
3	1	8	100 Hz
4	1	8	500 Hz
5	1	8	1000 Hz

4. HASIL DAN PEMBAHASAN

Implementasi sistem ditunjukkan pada gambar 8. Digunakan MATLAB pada komputer sebagai penghasil data sinyal sinusoidal kemudian dikirimkan melalui USB UART pada papan pengembang Atlys yang telah diimplementasikan *programmable* DAC pada FPGA Xilinx Spartan-6 XC6SLX45. Luarannya dilewatkan pada tapis pelewat rendah dari komponen R dan C kemudian ditampilkan pada osiloskop.



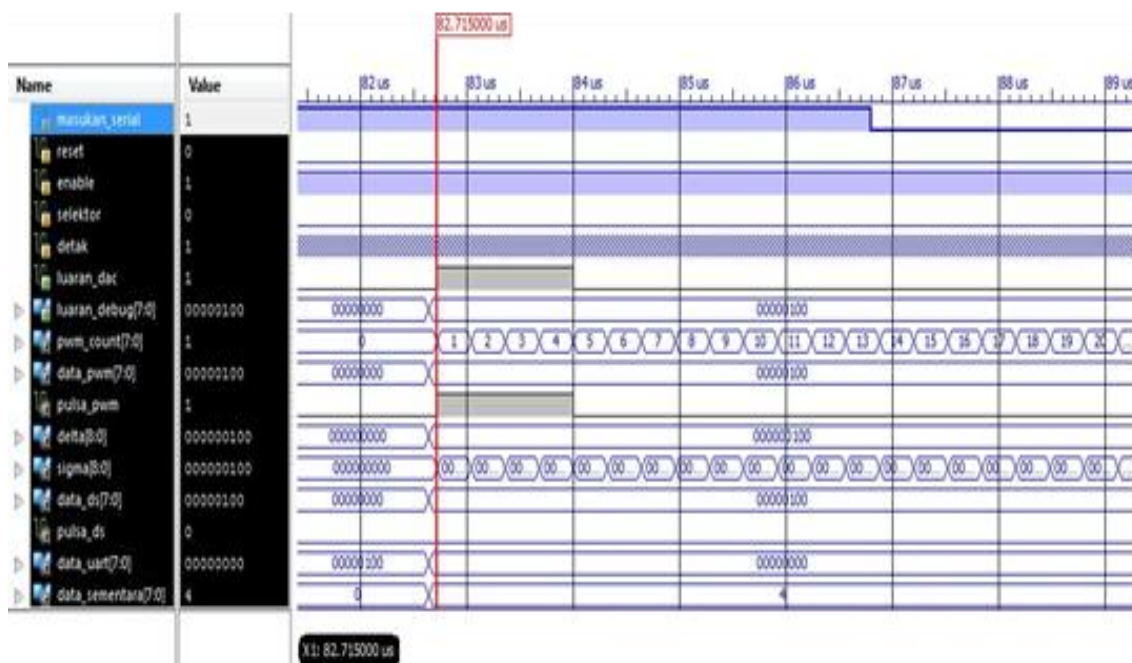
Gambar 8 Implementasi sistem pada penelitian

4.1 Hasil Uji Fungsional

Sebelum diimplementasikan pada FPGA, desain yang telah dirancang dilakukan pengujian simulasi apakah desain sudah berjalan sesuai fungsinya. Dalam ISE WebPack telah terdapat simulator yang berfungsi untuk mengujinya yaitu Isim.

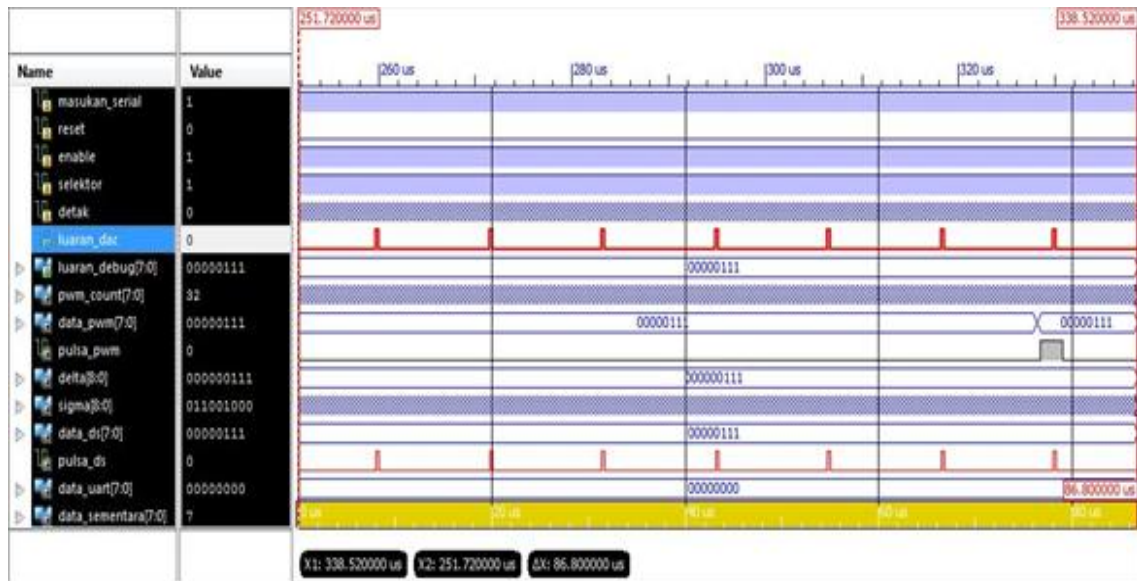
Gambar 9 menampilkan simulasi modul ini dengan luaran PWM DAC. **reset** diberi nilai '0' enable diaktifkan dengan nilai '1', detak diberikan detak dengan periode 10 ns dan selektor akan meneruskan **pulsa_pwm** sehingga diberi nilai '0'. Diberikan nilai 4 (biner 00000100) secara serial melalui **masukan_serial**, maka nilai tersebut akan diambil oleh **data_uart** dan disimpan pada **data_sementara**. Selanjutnya barulah proses modulasi PWM berlangsung, dan terlihat didapatkan luaran **pulsa_pwm** yang aktif (*on cycle*) sepanjang 4 **pwm_count** artinya luarannya sudah proporsional terhadap masukannya. **luaran_dac** merupakan nilai dari **pulsa_pwm** yang diteruskan dengan sinyal yang diberi warna hitam sebagai penanda.

Gambar 10 merupakan hasil simulasi menggunakan delta-sigma DAC sebagai luaran. Luaran dari delta-sigma DAC diperlihatkan melalui sinyal **pulsa_ds** dengan garis sinyal berwarna merah. Luaran dari programmable DAC yaitu **luaran_dac** juga dengan sinyal berwarna merah yang meneruskan sinyal **pulsa_ds**, ini aktif ketika **selektor** bernilai '1'. Diberikan nilai 7 (biner 00000111) secara serial melalui **masukan_serial** kemudian disimpan pada **data_sementara**, dari sini dilakukan modulasi menggunakan delta-sigma **data_sementara** diteruskan pada **data_ds** sebagai sinyal masukan proses modulasi.



Gambar 9 Hasil uji simulasi fungsional *programmable* DAC dengan modulasi PWM DAC

Berbeda dengan PWM DAC, luaran yang proporsional diwakili lewat perbandingan *on cycle* terhadap *duty cycle* atau lebar pulsa. Bentuk proporsional modulasi delta-sigma merupakan kerapatan pulsa sehingga disebut juga PDM (*Pulse Density Modulation*), luaran pada proses delta-sigma akan mengeluarkan jumlah pulsa sesuai dengan nilai masukan, hanya saja nilai bukan nilai terurut seperti pada PWM sehingga akan menghasilkan pulsa dengan kerapatan yang berbeda seperti terlihat pada gambar terdapat 7 luaran pulsa dengan nilai 1 pulsa sinyal **sigma**.



Gambar 10 Hasil uji simulasi fungsional *programmable* DAC dengan modulasi Delta-sigma DAC

4.2 Hasil Sintesis dan Implementasi

Proses sintesis ini merupakan proses translasi HDL menjadi rangkaian RTL (*Register Transfer Level*) yang nantinya akan diimplementasikan pada FPGA menyesuaikan fitur teknologi pada FPGA Spartan-6. Alat sintetis yang digunakan adalah serangkaian dengan perangkat ISE WebPack yaitu XST.

Dalam melakukan sintesis dapat dikonfigurasi untuk optimasi. Optimasi ini meliputi optimasi kecepatan atau optimasi area implementasi FPGA. Dalam penelitian ini kita gunakan optimalisasi kecepatan.

Tabel 2 menampilkan hasil penggunaan sumber daya yang dibutuhkan untuk mengimplementasikan *programmable* DAC pada FPGA Spartan-6 LX45. Jumlah *Look Up Table* (LUT) yang digunakan total 75 dengan rincian 21 LUT untuk modul penerima UART, 4 LUT untuk modul pembagi detak dan 50 LUT digunakan untuk modul *programmable* DAC atau *Top Level Function*. Adapun LUT yang disediakan pada FPGA seri ini adalah 27.288 unit sehingga penggunaan hanya menghabiskan 1 % sumber daya.

Jumlah *slice register* atau FF yang digunakan total 65 dengan rincian 25 unit untuk modul penerima UART, 5 unit untuk modul pembagi detak dan 35 unit untuk modul *programmable* DAC. Adapun *slice registers* yang tersedia pada FPGA Spartan-6 LX45 ini adalah 54.576 unit sehingga penggunaan hanya menghabiskan 1 % sumber daya.

Penggunaan sumber daya ini hanya memanfaatkan 1 % dari komponen yang tersedia dalam FPGA Spartan-6 LX45. Ini berarti *programmable* DAC mampu diimplementasikan tanpa memakan banyak sumber daya sehingga komponen yang lain dapat dialihkan untuk pembuatan modul-modul yang lain.

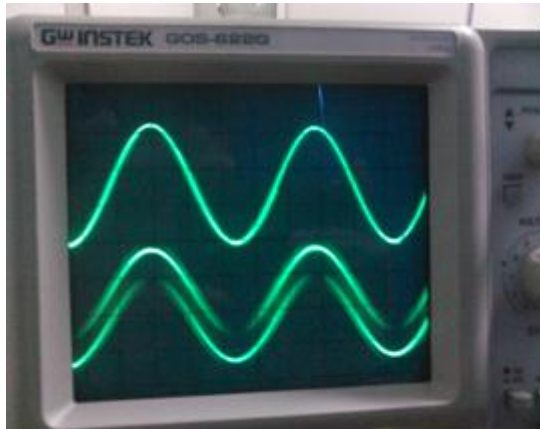
Tabel 2 Penggunaan sumber daya FPGA untuk implementasi *programmable* DAC

Modul	LUTs	Slice Register / FF
Programmable DAC	50	35
Modul penerima UART	21	25
Modul pembagi detak	4	5
Total	75	65

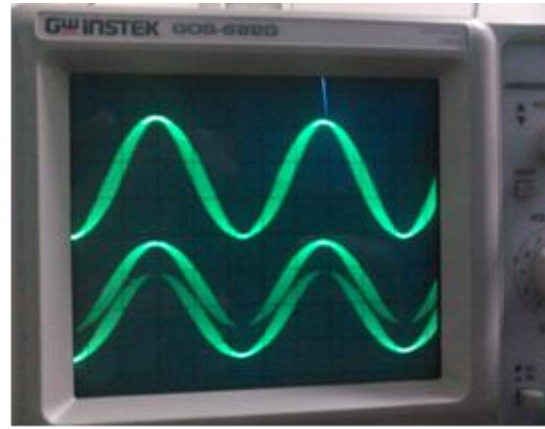
4.3 Analisis Sinyal

Analisis sinyal dilakukan dengan tujuan membandingkan sinyal yang ditampilkan pada osiloskop dengan sinyal yang dihasilkan oleh perangkat lunak MATLAB pada komputer. Sinyal ini divariasikan sesuai Tabel 1 dan didapatkan hasil pada Tabel 3.

Gambar 11 menunjukkan hasil rekonstruksi sinyal sinusoidal, frekuensi sinyal yang dikirimkan adalah 100 Hz dan frekuensi sinyal yang ditampilkan pada osiloskop juga 100 Hz, maka sinyal yang direkonstruksi sudah sesuai baik bentuk dan juga frekuensi luarannya. Untuk amplitudo didapatkan 3.3 V berdasarkan pada tegangan referensi.



Gambar 11 Hasil rekonstruksi sinyal sinusoidal modulasi delta-sigma



Gambar 12 Hasil rekonstruksi sinyal sinusoidal modulasi PWM

Tabel 3 Hasil luaran *programmable* DAC dengan variasi frekuensi

Variasi	Amplitudo Data	Amplitudo Luaran	Frekuensi Data	Frekuensi Luaran
1	1	3.3 V	10 Hz	10 Hz
2	1	3.3 V	50 Hz	50 Hz
3	1	3.3 V	100 Hz	100 Hz
4	1	3.3 V	500 Hz	500 Hz
5	1	3.3 V	1000 Hz	1000 Hz

Berdasarkan Gambar 11 juga didapatkan hasil bahwa sinyal dengan modulasi delta-sigma mampu menghasilkan sinyal lebih halus baik pada saat sebelum tapis rekonstruksi (sinyal bawah) dan juga setelah rekonstruksi (sinyal atas).

5. KESIMPULAN

Implementasi *programmable* DAC dapat dilakukan pada FPGA Spartan-6 LX45 menggunakan VHDL dengan Atlys sebagai papan pengembangnya.

Programmable DAC mampu merekonstruksi bentuk sinyal sinusoidal, sinyal sawtooth dan sinyal segitiga simetris. Hasil rekonstruksi sinyal menggunakan modulasi *delta-sigma* lebih halus dibandingkan menggunakan modulasi PWM pada sinyal dengan rasio frekuensi pencuplikan yang rendah.

Sumber daya yang dibutuhkan untuk mengimplementasikan *programmable* DAC adalah 75 dari 27.888 LUTs (1 %) dan 65 dari 54.576 *slice* register atau FF (1%).

DAFTAR PUSTAKA

- [1] Dubey, R., 2009, *Introduction to Embedded System Design Using Field Programmable Gate Arrays*, Springer-Verlag, London.
- [2] Frenzel, Jr., L. E., 2010, *Electronics Explained : The New Systems Approach to Learning Electronics*, Elsevier Inc, Oxford.
- [3] Grimbleby, J.,B., 2008, *Digital-to-analogue and Analogue-to-digital conversion*. School of Systems Engineering – Electronic Engineering, University of Reading, Berkshire.
- [4] Weisstein, Eric W., 2013, "Sinusoid." From MathWorld--A Wolfram Web Resource, <http://mathworld.wolfram.com/Sinusoid.html> diakses 11 Oktober 2013 pukul 04.07 WIB.
- Tan,Li, 2008, *Digital Signal Processing: Fundamentals and Applications*, Elsevier, London.