

# Aplikasi Platform Komputasi *Software-Defined Radio* (SDR) untuk *Digital Spectrum Analyzer*

Eko Marpanaji, Kadarisman Tejo Yuwono, Adi Dewanto

eko@uny.ac.id, arispra@uny.ac.id, adi@uny.ac.id

Pendidikan Teknik Elektronika Fakultas Teknik Universitas Negeri Yogyakarta  
Karangmalang, Yogyakarta 55281.

**Abstrak** – Penelitian ini bertujuan untuk mengkaji pengembangan aplikasi platform *Software-Defined Radio* (SDR) yaitu *Universal Software Radio Peripheral* (USRP) dan komputer PC untuk spektrum analiser. Permasalahan utama yang dihadapi adalah bagaimana arsitektur perangkat keras dan perangkat lunak yang diperlukan untuk membangun sebuah spektrum analiser digital berbasis SDR, dan bagaimana unjuk kerja sistem tersebut. Metoda yang digunakan dalam membangun sistem ini adalah mengimplementasikan USRP untuk proses digitasi sinyal analog dan komputer PC sebagai pengolah sinyal dan penampil hasilnya. Kunci utama sistem ini adalah perangkat lunak yang menjalankan fungsi pemisahan komponen frekuensi dan besarnya daya yang membentuk sinyal tersebut, yaitu algoritma *Fast Fourier Transform* (FFT). Pengembangan perangkat lunak yang digunakan dalam penelitian ini dimulai dengan simulasi algoritma-algoritma pengolah sinyal menggunakan Matlab, instalasi platform SDR dan uji coba sistem SDR menjalankan fungsi radio secara umum, dan pengembangan perangkat lunak untuk menjalankan fungsi spektrum analiser. Kesimpulan yang dapat diperoleh adalah arsitektur perangkat keras spektrum analiser digital dapat dibangun menggunakan USRP dan PC, sedangkan algoritma FFT merupakan kunci utama dalam arsitektur perangkat lunak sistem tersebut. Secara umum unjuk kerja prototipe sistem telah dapat bekerja dengan baik meskipun dalam bentuk skala laboratorium dan masih perlu diteliti lebih lanjut terutama kalibrasi dan tambahan fungsi-fungsi lain yang diperlukan.

## I. PENDAHULUAN

Spektrum analiser adalah sebuah alat ukur yang sangat diperlukan dalam pengembangan perangkat telekomunikasi. Manfaat spektrum analiser sangat banyak baik untuk keperluan pabrikasi, penelitian dan pengembangan, serta pendidikan. Fungsi spektrum analiser antara lain untuk pengujian alat telekomunikasi baik pengujian hasil produksi maupun proses perawatan dan perbaikan. Meskipun spektrum analiser memiliki banyak kegunaan dan sangat diperlukan di dalam bidang elektronika telekomunikasi seperti Internet, tidak semua instansi pendidikan dapat memiliki spektrum analiser karena harga sebuah spektrum analiser sangat mahal. Apalagi untuk spektrum analiser digital dengan frekuensi kerja dalam orde GHz dengan fasilitas dapat menyimpan data dan dapat disambungkan dengan komputer atau jaringan komunikasi harganya dapat mencapai ratusan juta rupiah bahkan milyaran rupiah. Untuk itu, perlu pemikiran dalam rangka mencari alternatif lain dalam mewujudkan fungsi spektrum analiser dengan harga yang lebih terjangkau.

Penelitian ini mengkaji alternatif lain dalam mewujudkan sebuah spektrum analiser dengan mengembangkan fungsi sebuah platform yang digunakan dalam penelitian dan pengembangan *Software-Defined Radio* (SDR). Perangkat keras yang digunakan adalah *Universal Software-Radio Peripheral* (USRP) sebagai perangkat ujung depan yang berfungsi melakukan digitasi sinyal RF analog dan komputer sebagai pengolah sinyal dan menampilkan hasil pengolahan sinyal secara grafik. USRP ini telah terbukti dapat digunakan untuk proses digitasi sinyal RF analog sampai orde GHz menjadi sinyal IF digital sehingga hasilnya dapat diproses oleh komputer PC biasa dengan menerapkan pengolahan sinyal digital. Perangkat ini digunakan untuk keperluan pengembangan *software radio* seperti yang sedang dilakukan oleh para peneliti dari kelompok GNU Radio, dengan menggunakan sistem operasi Linux. Seluruh fungsi radio termasuk modulasi, pemilihan saluran, demodulasi, dan *channel coding* diimplementasikan

dalam bentuk perangkat lunak yang dijalankan pada sebuah komputer PC.

Penelitian tentang SDR sudah dimulai sejak tahun 2006, dan sampai dengan tahun 2009 telah menghasilkan beberapa artikel seminar nasional/internasional dalam negeri serta jurnal nasional. Penelitian tentang SDR dimulai dari arsitektur perangkat keras dan perangkat lunak yang digunakan dalam membangun sebuah sistem komunikasi berbasis SDR, sampai dengan beberapa contoh aplikasinya dengan beberapa jenis modulasi yang digunakan adalah modulasi AM (*Amplitude Modulation*) dan modulasi FM (*Frequency Modulation*) untuk radio konvensional, sampai dengan modulasi *Gaussian Minimum Shift Keying* (GMSK) yang digunakan untuk telepon seluler GSM[1], serta modulasi digital lainnya seperti *Binary Phase Shift Keying* (BPSK) dan *Differential Binary Phase Shift Keying* (DBPSK)[2], *Quaternary Phase Shift Keying* (QPSK) dan *Differential Quaternary Phase Shift Keying* (DQPSK)[3] serta beberapa jenis modulasi digital lainnya.

Permasalahan utama dalam pengembangan aplikasi platform SDR untuk spektrum analiser yang akan dilakukan dalam penelitian ini adalah bagaimana cara mewujudkan sebuah spektrum analiser digital ditinjau dari segi perangkat keras maupun perangkat lunak yang diperlukan, dengan menggunakan perangkat USRP dan PC. Selain itu, bagaimana pula hasil unjuk kerja spektrum analiser dilihat dari frekuensi maksimum yang dapat tangani, resolusi frekuensi, serta fasilitas-fasilitas lain yang dapat dihasilkan dari spektrum analiser digital tersebut. Permasalahan-permasalahan tersebut akan dijawab dalam penelitian dengan mengikuti peta jalan pengembangan aplikasi SDR yang telah dilakukan sebelumnya.

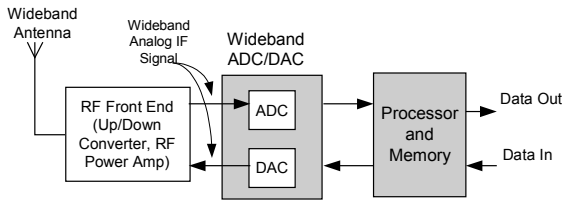
## II. SOFTWARE-DEFINED RADIO (SDR)

*Software-Defined Radio* (SDR), ada yang menyebut juga *Software Radio* (SWR), diperkenalkan pertama kali pada tahun 1991 oleh Joseph Mitola[4][5]. Istilah SDR ini

digunakan untuk menunjuk sebuah kelas radio yang dapat dikonfigurasi ulang atau diprogram ulang[6], sehingga menghasilkan sebuah jenis perangkat komunikasi nirkabel dengan mode dan band frekuensi ditentukan oleh fungsi perangkat lunak.

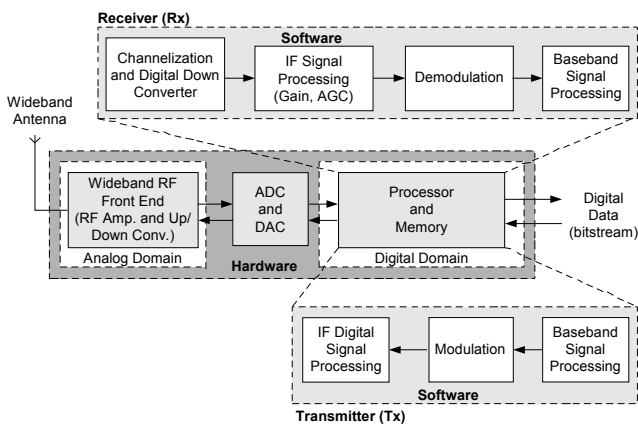
SDR memiliki keuntungan karena sifat fleksibilitas (*flexibility*), lengkap dan dapat dikonfigurasi ulang secara mudah (*complete and easy reconfigurability*), dapat diskala (*scalability*), dapat diprogram ulang (*reprogrammability*), serta dapat diperluas (*expandability*) [4][7].

Arsitektur SDR ideal akan menempatkan ADC/DAC sedekat mungkin dengan antena untuk melakukan konversi analog ke digital atau digital ke analog, sehingga membutuhkan wideband ADC/DAC. Fungsi radio akan dilakukan oleh perangkat lunak yang dijalankan oleh prosesor, sehingga lebih fleksibel[4][6][8][9]. Namun demikian, keterbatasan teknologi dan mahalannya wideband ADC/DAC mendorong untuk sedikit mengubah arsitektur SDR dalam menempatkan ADC/DAC sehingga menjadi realistis seperti yang ditunjukkan pada Gbr 1.



Gbr. 1. Arsitektur SDR Realistis

Arsitektur SDR yang lebih realistis menempatkan wideband ADC/DAC setelah *Down/Up Converter*, sehingga konversi analog ke digital atau sebaliknya dilakukan terhadap sinyal *Intermediate Frequency* (IF) dengan frekuensi yang lebih rendah dibanding sinyal *Radio Frequency* (RF). Arsitektur SDR dilihat dari segi perangkat keras dan perangkat lunak yang diperlukan dengan mengacu pada arsitektur SDR realistis ditunjukkan pada Gbr. 2.



Gbr. 2. Arsitektur SDR untuk Transmitter dan Receiver

Arsitektur tersebut saat ini banyak digunakan para peneliti dalam rangka mengembangkan teknologi SDR untuk berbagai keperluan, termasuk peneliti-peneliti SDR yang tergabung dalam kelompok GNU Radio dengan menggunakan ujung depan (front end) yang mereka sebut *Universal Software Radio Peripheral* atau USRP. Istilah *Software Radio* (SWR) digunakan sebagai alternatif istilah

*Software-Defined Radio* yang menggunakan komputer pribadi (PC) atau *General Purpose Processor* (GP) sebagai prosesornya.

#### A. Undersampling

Kesulitan dalam proses digitasi sinyal RF adalah keterbatasan kecepatan pencuplikan (*sampling*) dari ADC. Berdasarkan teori sampling Nyquist, bahwa sinyal baseband dengan frekuensi maksimum  $f_a$  harus dicuplik dengan frekuensi sampling  $f_s \geq 2f_a$ . Proses digitasi sinyal RF atau IF dalam orde ratusan MHz sampai dengan GHz. Cara yang dilakukan untuk menurunkan biaya dalam proses pencuplikan akibat mahalannya ADC untuk frekuensi tinggi adalah dengan menerapkan undersampling dalam proses digitasi. Namun demikian, undersampling harus dilakukan secara cermat dalam memilih frekuensi samplingnya. Kaidah yang digunakan adalah: sinyal bandpass yang terletak antara frekuensi bawah  $f_L$ , frekuensi tengah  $f_C$ , dan frekuensi atas  $f_U$  atau bandwidth  $B = f_U - f_L$  dapat disampling dengan frekuensi sampling  $f_s \geq 2B$  atau dengan frekuensi sampling yang besarnya dapat dipilih berdasarkan persamaan[9][10]:

$$\frac{2f_U}{n} \leq f_s \leq \frac{2f_L}{n-1} \quad (1)$$

dengan nilai  $n$  adalah bilangan integer yang memenuhi syarat:  $1 \leq n \leq \left\lfloor \frac{f_U}{f_U - f_L} \right\rfloor$

Perencanaan frekuensi sampling harus benar-benar hati-hati agar penerapan undersampling dapat berhasil dengan baik. Secara singkat, pemilihan frekuensi dapat dijelaskan sebagai berikut:

- Pilih frekuensi sampling sedemikian rupa sehingga kelipatan frekuensi samplingnya memenuhi:

$$\frac{(n-1)f_s}{2} \leq f_L \text{ dan } \frac{nf_s}{2} \geq f_U$$

sehingga menjamin seluruh komponen spektrum

bandpass sinyal jatuh pada daerah baseband  $\left[0, \frac{f_s}{2}\right]$

hasil undersampling.

- Persamaan tersebut dapat dinyatakan:

$$f_s \leq \frac{2f_L}{(n-1)} \text{ dan } f_s \geq \frac{2f_U}{n} \quad (2)$$

Atau lebih kompaknya menjadi:

$$\frac{2f_U}{n} \leq f_s \leq \frac{2f_L}{(n-1)} \quad (3)$$

Dengan batasan:

$$n \in \mathbb{N} \text{ sedemikian rupa sehingga } 1 \leq n \leq \left\lfloor \frac{f_U}{f_U - f_L} \right\rfloor.$$

- Untuk  $n_{\text{ganjil}}$  maka spektrum hasil sampling **tidak terbalik** aksisnya dan  $n_{\text{genap}}$  maka spektrum hasil sampling **akan terbalik** aksisnya.

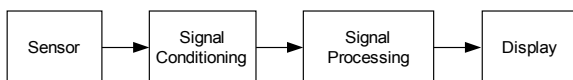
### B. Universal Software Radio Peripheral (USRP)

USRP (*Universal Software Radio Peripheral*) merupakan sebuah perangkat ujung depan (*front end*) dalam sebuah arsitektur SDR yang sekarang banyak digunakan oleh para peneliti SDR dari kelompok GNU Radio. Ujung depan ini melakukan beberapa fungsi antara lain: (1) mengubah frekuensi sinyal RF (*Radio Frequency*) menjadi sinyal IF (*Intermediate Frequency*) atau sering disebut sebagai *down-converter* dan proses sebaliknya yang sering disebut dengan *up-converter*; (2) melakukan konversi dari sinyal IF analog menjadi sinyal IF digital menggunakan *A/D converter* dan proses sebaliknya menggunakan *D/A converter*; (3) melakukan proses *digital down converter* (DDC) dan desimasi untuk menurunkan laju data digital yang akan dikirimkan melalui port USB. Proses ini dilakukan dengan menggunakan sebuah chip FPGA; (4) melakukan komunikasi dengan komputer (mengirim dan menerima sinyal digital) menggunakan antarmuka port USB 2.0. USRP terdiri dari sebuah *main board* untuk menjalankan proses (2), (3), dan (4), serta beberapa *daughterboard* untuk melakukan proses (1). USRP mendukung 4 buah daughterboard yaitu dua buah daughterboard untuk pemancar (Tx) dan dua buah daughterboard untuk penerima (Rx).

Spesifikasi main board USRP yang digunakan dalam penelitian ini adalah: port USB 2.0 untuk koneksi dengan komputer, ADC 12-bit dengan kecepatan sampling 64 MSPS sehingga dengan prinsip aliasing dapat melakukan proses digitasi dengan jangkauan frekuensi aliasing -32 MHz s.d. 32 MHz, DAC 14 bit dengan frekuensi clock 128 MSPS sehingga memiliki frekuensi Nyquist sebesar 64 MHz dan sinyal analog yang dihasilkan terbatas 10 mWatt. Sedangkan untuk daughterboard yang digunakan adalah Basic Tx dan Basic Rx sehingga belum ada proses down/up converter dan frekuensi pemancar terbatas maksimum 50 MHz.

## III. METODE PENELITIAN

Spektrum analiser adalah sebuah alat ukur atau alat instrumentasi, sehingga metoda penelitian tetap memperhatikan kaidah-kaidah yang ada di bidang instrumentasi seperti yang ditunjukkan pada Gbr. 3 berikut ini.

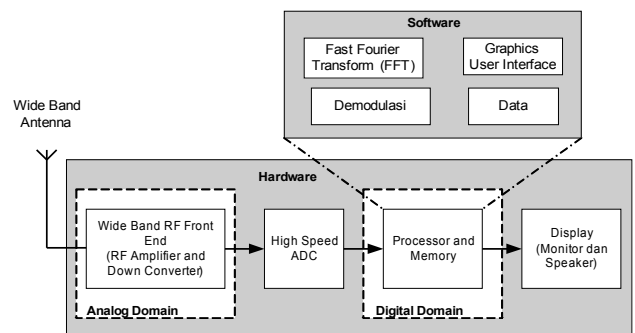


Gbr. 3. Blok diagram sistem instrumentasi

Sedangkan blok diagram spektrum analiser digital yang akan diteliti berdasarkan arsitektur platform komputasi SDR menggunakan USRP dapat dilihat pada Gbr 4. Platform komputasi SDR pada prinsipnya adalah kumpulan perangkat keras yang disusun sedemikian rupa sehingga memungkinkan fungsi radio dapat diimplementasikan dalam bentuk perangkat lunak. SDR mengimplementasikan fungsi

radio dengan menggunakan perangkat lunak, sehingga syarat yang harus dipenuhi adalah bahwa sinyal yang diproses harus berupa sinyal digital. Oleh karena itu, penelitian tentang spektrum analiser digital dengan menggunakan USRP dan komputer PC ini pada prinsipnya adalah pengembangan perangkat lunak yang berfungsi untuk menjalankan fungsi spektrum analiser dalam hal ini adalah komputasi DFT dan dalam pengolahan sinyal digital algoritma yang digunakan adalah algoritma FFT.

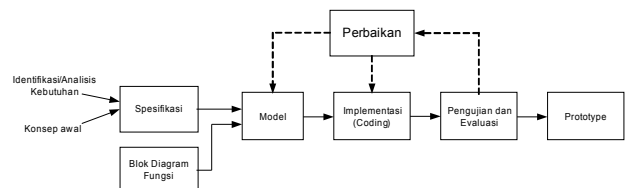
Berdasarkan Gbr. 4, maka antenna berfungsi sebagai pengindera (*sensor*) dari spektrum analiser, RF *amplifier* dan *down converter* berfungsi sebagai pengkondisi sinyal (*signal conditioning*) termasuk ADC, perangkat lunak (*software*) untuk menjalankan fungsi spektrum analiser dan fungsi radio penerima adalah wujud dari pengolahan sinyal (*signal processing*), serta layar monitor atau speaker adalah sebagai wujud dari tampilan (*display*) dalam sistem instrumentasi.



Gbr. 4. Blok diagram spektrum analiser digital menggunakan platform SDR

Arsitektur perangkat keras SDR yang digunakan dalam penelitian ini pada prinsipnya sama dengan arsitektur SDR untuk komunikasi data digital, perbedaannya terletak pada fungsi yang dijalankan pada perangkat lunak yang dijalankan pada prosesor. Hal ini sekaligus membuktikan sifat fleksibilitas sistem SDR sebagai salah satu sifat unggulan dibangunnya sistem SDR. Dengan demikian, pengalaman dalam membangun arsitektur SDR dapat digunakan juga mengembangkan sebuah spektrum analiser berbasis SDR, dan penelitian ini tinggal menekankan pada pengembangan perangkat lunak yang menjalankan fungsi pengolahan sinyal digital untuk mengukur dan menampilkan spektrum frekuensi dan daya sinyal RF yang diproses.

Metoda yang digunakan dalam hal pengembangan perangkat lunak untuk menjalankan fungsi spektrum analiser digital dan radio perangkat lunak ditunjukkan pada Gbr. 5 berikut ini.



Gbr. 5. Metodologi

Prototype spektrum analiser digital dengan menggunakan platform komputasi SDR diperoleh dari beberapa proses, dimulai dari analisis kebutuhan, penentuan spesifikasi,

desain blok diagram fungsi yang akan menghasilkan sebuah model dari sebuah algoritma, dan dilanjutkan dengan implementasi atau *coding* (pemrograman), dan terakhir adalah pengujian. Proses perbaikan dilakukan berdasarkan hasil pengujian dan evaluasi dan kemungkinan dapat mengarah pada model, implementasi, atau kedua-duanya.

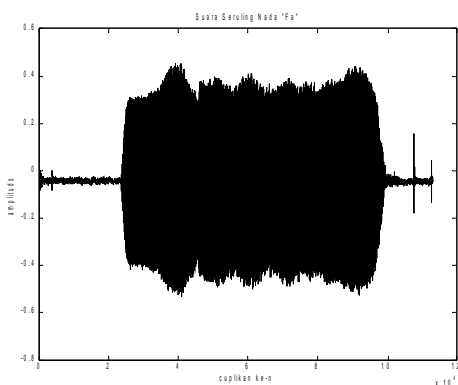
#### IV. HASIL DAN PEMBAHASAN

Kegiatan tahap I yang dilaksanakan pada tahun I menekankan pada pendekatan arsitektur perangkat keras dan perangkat lunak spektrum analiser digital berbasis SDR, mulai dari studi pustaka, simulasi algoritma pengolahan sinyal yang diperlukan untuk mengukur spektrum sinyal, serta implementasi algoritma menggunakan bahasa pemrograman termasuk pemilihan jenis bahasa pemrograman yang sesuai, serta pengujian fungsi pengolahan sinyal meskipun masih dilakukan secara terpisah untuk tiap-tiap bagian blok diagram perangkat lunak untuk menjalankan fungsi spektrum analiser. Hasil penelitian yang dilakukan dijelaskan sebagai berikut.

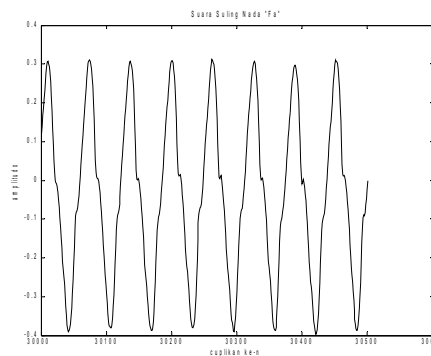
##### A. Simulasi Matlab

Simulasi Matlab tentang ekstraksi komponen frekuensi sinyal digital menggunakan FFT terhadap sinyal digital. Simulasi ini menggunakan sinyal audio digital yang diperoleh dari proses digitasi sinyal audio analog melalui sound card komputer sebagai studi kasus. Dengan asumsi, jika fungsi ekstraksi sinyal audio digital dapat bekerja dengan baik, maka fungsi ekstraksi sinyal digital untuk jangkauan frekuensi yang lebih tinggi juga dapat dilakukan. Suara sinyal audio yang direkam adalah suara seruling dengan mengambil salah satu nada yaitu nada "fa" dan juga nada-nada yang lain untuk pengujian algoritma FFT yang disimulasikan untuk menentukan komponen frekuensi tiap-tiap sinyal audio tersebut.

Gbr. 6 menunjukkan sinyal audio digital suara suling untuk nada "fa" untuk 120.000 cuplikan dengan frekuensi cuplikan sebesar 8000 Hz. Suara suling yang direkam digambarkan pada cuplikan ke-20.000 s.d. 10.000 dimana amplitudo sinyal sangat besar dibanding sebelum dan sesudah cuplikan tersebut. Sinyal audio digital suara seruling untuk 500 cuplikan ditunjukkan pada Gbr 7, dimana sinyal audio tersebut nampak berbentuk sinusoidal.

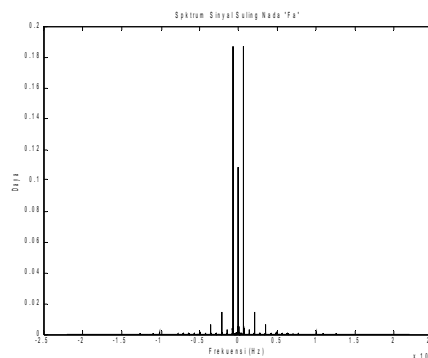


Gbr. 6. Sinyal suara suling nada "Fa" keseluruhan

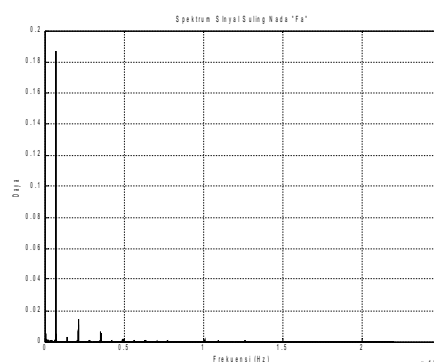


Gbr. 7. Potongan sinyal suara suling nada "Fa" untuk 500 cuplikan

Simulasi Matlab ekstraksi komponen frekuensi sinyal digital hasil rekaman suara suling dengan nada "Fa" menggunakan algoritma FFT ditunjukkan pada gambar berikut ini. Gbr. 8 menunjukkan pasangan spektrum frekuensi. Biasanya spektrum analiser hanya menampilkan salah satu spektrum daya yaitu spektrum daya untuk frekuensi positif. Dengan demikian, skrip Matlab perlu ditambah untuk memisahkan spektrum daya untuk frekuensi positif dan menampilkan spektrum tersebut. Tampilan spektrum daya untuk frekuensi positif ditunjukkan pada Gbr. 9.



Gbr. 8. Spektrum Daya Sinyal suara suling nada "Fa" (pasangan)



Gbr 9. Spektrum Daya Sinyal suara suling nada "Fa" (positif)

Gambar spektrum daya tersebut menunjukkan bahwa komponen frekuensi sinyal rekaman suara suling nada "Fa" tersebut adalah frekuensi suling nada "Fa" kurang lebih sebesar 700 Hz yang memiliki amplitudo paling dominan, serta beberapa frekuensi harmonisa dan noise yang memiliki amplitudo lebih kecil. Penentuan frekuensi nada "Fa" dapat dilakukan dengan menambahkan skrip program yang fungsinya memilih amplitudo terbesar dari spektrum daya

hasil ekstraksi komponen frekuensi sinyal karena amplitudo yang lebih kecil merupakan frekuensi harmonisa dan frekuensi sinyal noise. Hasil pemilihan komponen frekuensi utama menggunakan Matlab menunjukkan bahwa frekuensi paling dominan adalah 703.1937 seperti yang ditunjukkan oleh tampilan keluaran dari Matlab berikut ini.

```
>> nada
    frekuensinya = 703.1937
```

Hasil simulasi ekstraksi frekuensi komponen sinyal untuk nada yang lain ditunjukkan pada Tabel I.

Simulasi Matlab tentang ekstraksi komponen frekuensi dan pemilihan komponen frekuensi utama ini menunjukkan bahwa fungsi ekstraksi komponen frekuensi sebagai fungsi dasar membangun spektrum analiser digital dapat dilakukan dengan mengim-plementasikan algoritma FFT sebagai perangkat lunak yang menjalankan fungsi spektrum analiser.

TABEL I. HASIL PENGUJIAN ALGORITMA FFT UNTUK MENENTUKAN KOMPONEN FREKUENSI SINYAL

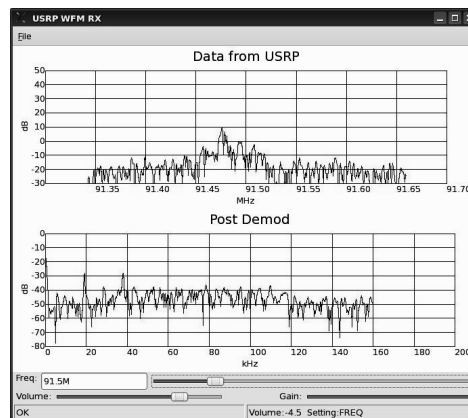
No	Nada	Frekuensi yang di hasilkan (Hz)
1.	"Do"	549,097
2.	"Re"	585,266
3.	"Mi"	660,127
4.	"Fa"	703,194
5.	"Sol"	787,812
6.	"La"	888,918
7.	"Si"	987,163
8.	"Do" tinggi	1.044,900

### B. Uji Coba Software-Defined Radio (SDR)

Ujicoba SDR dimulai dengan instalasi sistem operasi Linux dan instalasi tarball GNU Radio sebagai driver untuk menjalankan USRP. Sistem operasi Linux yang digunakan adalah Fedora Core 6 dengan menyediakan fasilitas kompilier Python beserta perangkat lunak bantu yaitu SWIG dan C++ untuk menjalankan driver USRP dan fungsi pengolahan sinyal digital yang sebagian besar algoritmanya diimplementasikan dalam bentuk bahasa C++. Instalasi tarball GNU Radio digunakan untuk menjalankan USRP sehingga dapat melakukan digitasi sinyal analog dan menghasilkan sinyal digital yang dikirimkan melalui port USB. Beberapa hal penting yang perlu diperhatikan dalam instalasi tar ball GNU Radio ini, adalah: (a) Instalasi Linux harus menjamin bahwa sound card dapat terdeteksi dan berfungsi dengan baik; (b) Instalasi tar ball dimulai dari instalasi Base line yaitu: FFTW, CPP Unit, SWIG dan Boost dan dijamin tidak ada error saat instalasi; (c) Instalasi wxPython terlebih dahulu serta setting path untuk python serta pengujiannya untuk menjamin bahwa wxPython telah berfungsi dengan baik; (d) Pengujian integritas fungsi perangkat lunak sehingga fungsi SDR dapat bekerja mengingat perangkat lunak opensource menggunakan gabungan beberapa sistem yang berbeda.

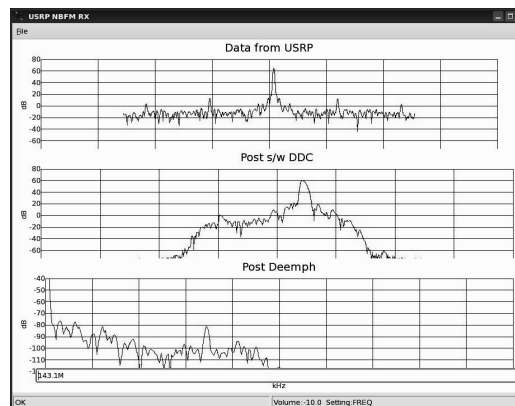
Ujicoba menjalankan fungsi SDR dapat dilakukan jika proses instalasi tarball dan post instalation untuk wxPython dapat berhasil dengan baik. Beberapa hasil ujicoba menjalankan fungsi SDR yang dilakukan dalam penelitian ini adalah sebagai berikut.

Gbr. 10 menunjukkan bahwa fungsi SDR dapat berjalan dengan baik untuk penerimaan radio FM broadcast.



Gbr. 10. Spektrum Sinyal Pemancar Stasiun Radio FM 91.5 MHz

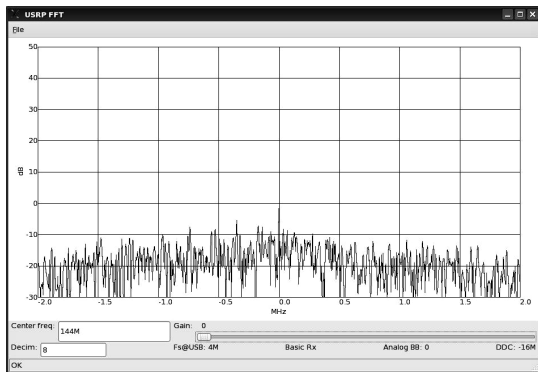
Gbr. 11 menunjukkan bahwa fungsi SDR dapat berjalan dengan baik untuk menerima sinyal radio FM komunikasi (HT).



Gbr. 11. Spektrum Sinyal Pemancar Radio Komunikasi (HT) pada frekuensi 143.1 MHz

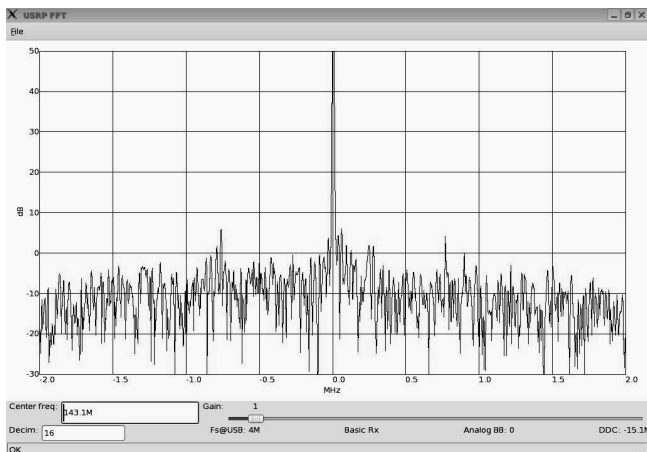
Selanjutnya adalah implementasi algoritma FFT untuk menghasilkan sebuah sistem spektrum analiser digital berbasis SDR dengan menggunakan USRP dan komputer PC. Bahasa pemrograman tetap menggunakan Python dengan beberapa dukungan source code opensource untuk lebih memudahkan dalam mengembangkan sistem ini. Berikut ini hasil pengujian prototipe sistem spektrum analiser digital berbasis SDR yang dilakukan dalam penelitian ini.

Gbr. 12 menunjukkan tampilan spektrum analiser saat tidak ada sinyal pemancar disekitar sistem, sehingga yang ditampilkan adalah spektrum frekuensi noise atau sinyal RF yang dihasilkan oleh beberapa pemancar yang terletak jauh dari sistem.



Gbr. 12. Spektrum Analiser (FFT): Tidak ada sinyal pemancar

Gbr. 13 menunjukkan spektrum sinyal RF dimana terdapat sinyal pemancar yang dipancarkan dari sebuah alat radio komunikasi jinjing (HT) dengan frekuensi 143.100 MHz. Sinyal tersebut memiliki power begitu besar terdeteksi oleh sistem sebagai petunjuk bahwa ada sinyal RF selain sinyal noise yang ditampilkan. Hasil pengujian ini juga menunjukkan bahwa prototipe spektrum analiser digital berbasis SDR dapat bekerja dengan baik meskipun dalam bentuk skala laboratorium dan fungsi tampilan yang masih sederhana.



Gbr. 13. Spektrum Analiser (FFT): Sinyal pemancar HT 143.1 MHz

## V. KESIMPULAN

Berdasarkan kegiatan yang dilakukan dalam penelitian ini, secara umum dapat disimpulkan bahwa:

1. Arsitektur perangkat keras spektrum analiser digital menggunakan platform komputasi SDR dapat menggunakan arsitektur perangkat keras SDR yang umum digunakan dengan menjalankan fungsi pengolahan sinyal digital yang berbeda.
2. Arisektur perangkat lunak spektrum analiser digital menggunakan platform komputasi SDR menggunakan algoritma FFT sebagai tulang punggung dalam pengolahan sinyal digital khususnya dalam ekstraksi komponen frekuensi sinyal.
3. Algoritma FFT untuk ekstraksi komponen frekuensi sebagai fungsi utama dalam menjalankan

fungsi spektrum analiser digital menggunakan platform komputasi SDR telah disimulasikan dalam program Matlab dengan studi kasus sinyal audio digital. Hasil simulasi menunjukkan bahwa algoritma FFT mampu melakukan analisa komponen frekuensi sebuah sinyal dan bahkan dapat menentukan komponen fekuensi yang mana yang akan dipilih.

4. Prototipe spektrum analiser digital berbasis SDR yang dihasilkan dapat bekerja dengan baik yaitu dapat mendeteksi daya pemancar beserta frekuensinya yang ditunjukkan dengan ukuran daya yang cukup besar sedangkan spektrum frekuensi sinyal noise ditampilkan dengan ukuran daya yang jauh lebih kecil.
5. Prototipe spektrum analiser digital berbasis SDR masih dalam bentuk skala laboratorium dan diuji coba dengan daerah frekuensi dengan orde MHz, sehingga perlu dikembangkan lebih lanjut dan diteliti untuk jangkauan frekuensi yang lebih tinggi (orde GHz).

## PUSTAKA RUJUKAN

- [1] E. Marpanaji, dkk., Pengukuran Unjuk Kerja Modulasi GMSK pada Software-Defined Radio Platform, *Jurnal Telkomnika*, Vol. 5, No. 2 Agustus 2007, pp. 73 – 84.
- [2] E. Marpanaji, dkk., “Studi Eksperimen Unjuk-Kerja Modulasi DBPSK pada Platform Software-Defined Radio (SDR).” *Jurnal Technoscientia*, Vol. 1 No. 1 Agustus 2008, pp. 14 – 22.
- [3] E. Marpanaji, dkk., “Experimental Study of DQPSK Modulation on SDR Platform.” *ITB Journal of Information and Communication Technology*, Vol. 1 No. 2 November 2008, pp. 84 – 98.
- [4] J. H. Reed, “Software Radio: A Modern Approach to Radio Engineering.” New Jersey, Prentice Hall, 2002.
- [5] Steinheider, J., “Software-defined Radio Comes of Age”, *Mobile Radio Technology*, Feb 1<sup>st</sup>, [Online] [http://www.vanu.com/resources/intro/software-defined\\_radio\\_comes\\_of\\_age.html](http://www.vanu.com/resources/intro/software-defined_radio_comes_of_age.html), 2003.
- [6] J. Mitola III, “Software Radio Architecture. Object-Oriented Approaches to Wireless Systems Engineering”, Canada: John Eiley & Sons, Inc, 2000.
- [7] F. Christensen, “A scalable Software-Defined Radio Development System”. [On-line] [http://www.xilinx.com/publications/xcellonline/xcell\\_51/xcell\\_51\\_pdf/xcell\\_51\\_es-sundance51.pdf](http://www.xilinx.com/publications/xcellonline/xcell_51/xcell_51_pdf/xcell_51_es-sundance51.pdf), 2004.
- [8] W. Lehr, “Software Radio: Implication for Wireless Services, Industry Structure, and Public Policy”, [Online] [http://itc.mit.edu/itel/docs/2002/Software\\_Radio\\_Lehr\\_Fuencis.pdf](http://itc.mit.edu/itel/docs/2002/Software_Radio_Lehr_Fuencis.pdf), 2002.
- [9] J. Guttag, “Software Radio for Adaptive Networking”, [Online] [http://web.mit.edu/deshpande\\_center/downloads/presos/ideastream2003\\_wireless.pdf](http://web.mit.edu/deshpande_center/downloads/presos/ideastream2003_wireless.pdf), 2003.
- [10] M. E. Angoletta, “From Analog to Digital Domain.” [Online]. Available: [http://humanresources.web.cern.ch/humanre-sources/external/training/special/DISP2003/DISP-2003\\_L01A\\_20Feb03.pdf](http://humanresources.web.cern.ch/humanre-sources/external/training/special/DISP2003/DISP-2003_L01A_20Feb03.pdf), 20 Feb 2003.
- [11] H. Harada, R. Prasad, “Simulation and Software Radio for Mobile Communications.” London: Artech House, 2002.