

Rancang Bangun Spectrum Analyzer Menggunakan Fast Fourier Transform Pada Single Board Computer

Afandi Nur Aziz Thohari*¹, Agfianto Eko Putro²,

¹Program Studi S2/S3 Ilmu Komputer, FMIPA UGM, Yogyakarta

²Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: *¹afandi.nat@gmail.com, ²agfi@ugm.ac.id

Abstrak

Spectrum analyzer merupakan alat yang berfungsi untuk mengubah sinyal dalam ranah waktu menjadi spektrum dalam ranah frekuensi. Sebuah alat penganalisa sinyal umumnya memiliki ukuran yang besar sebab terdiri dari banyak komponen seperti mixer, amplifier, local osilator, ADC dll. Selain ukuran, masalah dari spectrum analyzer yang ada di pasaran adalah tingginya utilitas yang disebabkan oleh banyaknya titik cuplikan dari sinyal. Utilitas yang tinggi mengakibatkan spektrum yang ditampilkan menjadi lambat.

Oleh sebab itu, dalam penelitian ini dirancang purwarupa spectrum analyzer berukuran kecil menggunakan single board computer, RTL-SDR dan LCD touchscreen. Fungsi spectrum analyzer diimplementasikan dalam sebuah perangkat lunak dengan menerapkan algoritma fast fourier transform. Masukan yang diproses berupa sinyal radio yang dicuplik dalam beberapa titik untuk mengetahui utilitas dari SBC.

Kesimpulan yang diperoleh dari hasil pengujian yaitu utilitas SBC dapat menampilkan spektrum dengan normal pada jumlah cuplikan (N) dari 512 sampai 32.768 titik. Sebab penggunaan N lebih dari 32.768 titik akan membebani cpu dan memori sehingga spektrum yang ditampilkan menjadi lambat. Kemudian jangkauan frekuensi yang dapat ditampilkan purwarupa oleh adalah 24 MHz sampai 1.769 MHz. Purwarupa dapat menunjukkan letak spektrum radio secara tepat setelah dilakukan perbandingan level spektrum menggunakan spectrum analyzer Anritsu MS2720T.

Kata kunci— Sinyal Radio, Cuplikan, Utilitas, level spektrum

Abstract

Spectrum analyser is device that function to change signal in time domain to spectrum to frequency domain. An analyzer signal device normally have big size cause consist so many component such as mixer, amplifier, local osilator and ADC etc. Beside of size, the problem of spectrum analyzer which on market is high utilities that caused by so many sample from signal. High utilities causing spectrum that displayed become slow.

Therefore, in this research designed spectrum analyser prototype with small size using single board computer, RTL-SDR, and LCD touchscreen. Function of spectrum analyser implemented on a software with applying fast fourier transform algoritm. Input that processed is radio signal which sampled into some point to find out SBC utilities.

Conclusions that obtained from testing result that are SBC utilities can showing normal spectrum on sampling number (N) from 512 until 32.768 points. Because N utilization more than 32.768 point will burden cpu and memory so that spectrum are shown become slow. Then frequency range that could prototype shown is 24 MHz until 1.769 MHz. Prototype can showing position radio spectrum appropriately after did comparision spectrum lever using spectrum analyser Anritsu MS2720T.

Keywords— Radio Signal, Sampling, Utilities, Spectrum Level

1. PENDAHULUAN

Perkembangan teknologi khususnya dalam bidang elektronika mengalami perkembangan yang cukup pesat. Salah satu diantaranya dibidang pengolahan sinyal digital. Dalam teori pengolahan sinyal digital terdapat sub bidang yang membahas analisis spektrum frekuensi atau disebut *spectrum analyzer*. Piranti ukur *spectrum analyzer* berfungsi untuk mengukur besarnya frekuensi dan daya suatu sinyal[1].

Kebutuhan *spectrum analyzer* di Indonesia sangat banyak, utamanya bagi telekomunikasi dan pendidikan[2]. Dalam bidang telekomunikasi, piranti ukur digital ini dimanfaatkan untuk melakukan pengujian pada alat telekomunikasi saat proses produksi dan perawatan. Kemudian digunakan pula untuk mengetahui alokasi frekuensi yang belum dipakai. Sedang dalam bidang pendidikan *spectrum analyzer* banyak dipakai di laboratorium fisika dan elektronika untuk praktikum analisa sinyal digital.

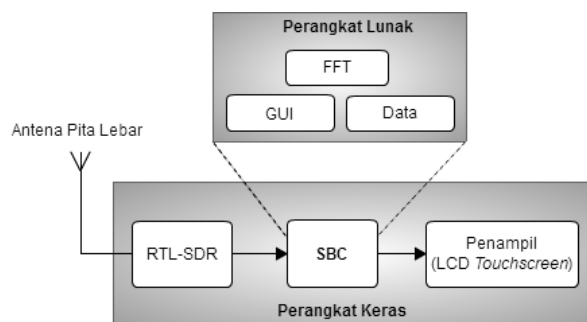
Spectrum analyzer yang ada di pasaran, umumnya memiliki ukuran yang besar sehingga tidak mudah dibawa kemana-mana (*portable*). Hal ini dikarenakan banyak komponen-komponen penyusun *spectrum analyzer* seperti mixer, amplifier, local osilator, ADC dll. Selain itu juga jangkauan frekuensi yang dapat ditampilkan hanya berkisar antara HF (*High Frequency*) sampai VHF (*Very High Frequency*) dengan utilitas sistem yang besar.

Salah satu alternatif untuk mengatasi masalah tersebut, dirancanglah sebuah purwarupa *spectrum analyzer* yang terdiri dari beberapa perangkat seperti *single board computer*, RTL-SDR dan LCD *touchscreen*. Sedangkan untuk fungsi *spectrum analyzer* diimplementasikan dalam sebuah perangkat lunak. Dalam perangkat lunak diterapkan algoritma *fast fourier transform* yang digunakan untuk mengubah sinyal menjadi spektrum secara cepat dan efisien[3].

2. METODE PENELITIAN

2.1 Analisis Sistem

Sistem yang dibangun pada penelitian ini adalah sebuah purwarupa alat yang dapat melakukan fungsi kerja seperti *spectrum analyzer*. Masukan dari purwarupa *spectrum analyzer* ini adalah sinyal radio, dan keluarannya berupa spektrum frekuensi. Perancangan purwarupa meliputi perancangan perangkat keras dan perancangan perangkat lunak. Perancangan perangkat keras dilakukan dengan cara mengintegrasikan *Single Board Computer*, RTL-SDR dan LCD *touchscreen* melalui port yang tersedia. Sedangkan perancangan perangkat lunak dilakukan dengan membuat program yang mengimplementasikan algoritma FFT untuk mengubah sinyal menjadi spektrum.



Gambar 1 Diagram Blok Sistem

Gambar 1 menunjukkan diagram blok rancangan sistem keseluruhan yang menggambarkan komponen perangkat keras dan perangkat lunak. Penjelasan bagian-bagian dari diagram blok Gambar 1 adalah sebagai berikut :

1. Antena Pita Lebar

Perangkat yang digunakan untuk menerima gelombang radio yang dipancarkan oleh berbagai stasiun pemancar.

2. RTL-SDR

Perangkat yang digunakan untuk mengolah sinyal mentah yang diterima antena.

3. SBC (*Single Board Computer*)

Perangkat yang digunakan untuk melakukan pemrosesan sinyal menjadi spektrum. Komponen perangkat lunak yang akan diimplementasikan pada SBC terdiri dari :

a. Data

Data yang akan diolah menjadi spektrum adalah data sinyal digital. Data ini di peroleh dari RTL-SDR setelah melakukan pengolahan sinyal radio analog.

b. FFT (*Fast Fourier Transform*)

Merupakan suatu metode yang digunakan untuk mengubah sinyal menjadi spektrum. Metode FFT terdiri dari beberapa proses yaitu melakukan pembalikan bit, menentukan *twiddle factor*, dan perhitungan dengan diagram kupu-kupu.

c. GUI (*Graphical User Interface*)

Tampilan perangkat lunak yang terdiri dari tampilan spektrum dan menu program.

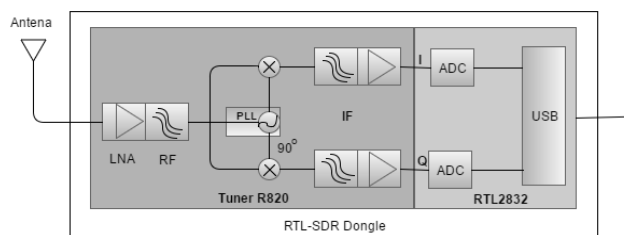
4. LCD *Touchscreen*

Perangkat yang digunakan untuk menampilkan menu program dan visualisasi spektrum yang dihasilkan.

Perancangan sistem pada Gambar 1 menjelaskan bahwa antar perangkat saling terintegrasi melalui port yang tersedia. Setelah semua perangkat terhubung dan dapat melakukan fungsi kerjanya, langkah selanjutnya adalah membuat perangkat lunak *spectrum analyzer*. Perangkat lunak dibuat dengan menerapkan algoritma FFT, data yang diolah adalah sinyal digital, dan tampilannya dalam bentuk GUI.

SBC yang digunakan untuk mengimplementasikan sistem adalah Raspberry Pi. Digunakan Raspberry Pi sebab memiliki pin GPIO yang dapat dihubungkan dengan perangkat eksternal lain seperti LCD. Selain itu Raspberry Pi juga mempunyai port USB yang pada penelitian ini digunakan untuk berkomunikasi dengan RTL-SDR. Raspberry pi memiliki 40 pin yang terdiri dari 28 pin GPIO, 4 pin *power* (baik 3,3V maupun 5V), dan 8 pin *ground*. Namun dari 40 pin SBC, hanya 26 pin saja yang digunakan untuk terhubung dengan LCD.

Selain pin konektor, SBC juga memiliki port USB yang nantinya terhubung dengan RTL-SDR. Perangkat penerima sinyal radio ini mengubah sinyal analog radio menjadi sinyal digital lalu mengalirkannya ke SBC. Komponen yang terdapat pada RTL-SDR terdiri dari port RF yang terhubung ke antena, lalu *tuner* yang berfungsi mencari sinyal RF, terakhir ADC (*Analog digital Converter*) yang berfungsi mengubah sinyal analog menjadi digital. Agar lebih jelas mengenai komponen yang terdapat pada RTL-SDR, maka dapat dilihat diagram konseptual pada Gambar 2.



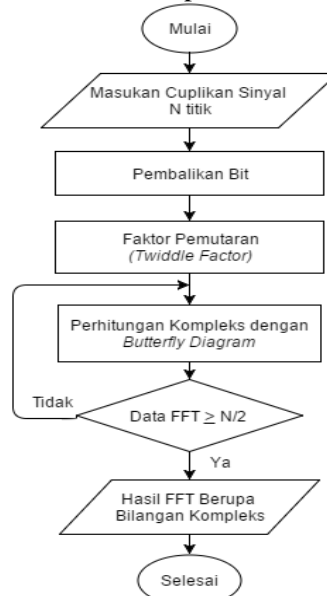
Gambar 2 Diagram Konseptual RTL-SDR[4]

Setelah semua perangkat keras terhubung dan dapat melakukan fungsi kerjanya, maka langkah selanjutnya adalah melakukan perancangan perangkat lunak. Proses perancangan perangkat lunak terdiri dari perancangan visualisasi spektrum, perancangan algoritma FFT, dan perancangan antarmuka pengguna. Perancangan visualisasi spektrum frekuensi ditunjukkan pada Gambar 3.



Gambar 3 Diagram Alir Proses Visualisasi Spektrum

Pada Gambar 3 terdapat beberapa langkah yang dilakukan. Program diawali dengan inisialisasi variabel yang digunakan seperti jumlah sampel, nilai frekuensi tengah, dll. Kemudian data sampel yang sudah didapatkan diproses menggunakan FFT dan hasilnya diubah dalam bentuk dBm. Terakhir hasil dalam dBm ditampilkan dalam bentuk spektum frekuensi.

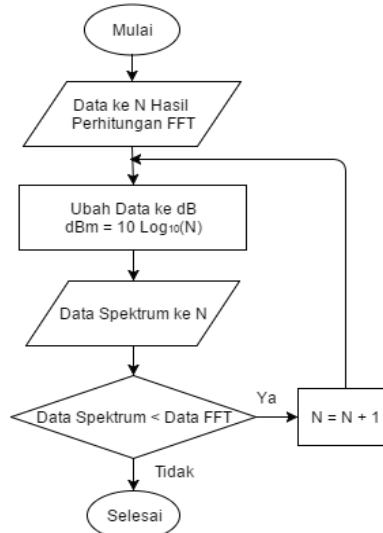


Gambar 4 Diagram Alir Komputasi FFT DIT

Gambar 4 menunjukkan diagram alir dari proses komputasi FFT DIT. Masukan berupa sinyal digital yang dicuplik sebanyak N titik. Setelah jumlah sampel sebanyak N titik didapatkan, langkah selanjutnya adalah merangkai nilai dari sampel tersebut dengan pembalik bit. Hasil dari proses pembalik bit adalah pengelompokan titik sesuai bilangan genap dan ganjil. Setelah dipisah dalam genap dan ganjil, langkah selanjutnya adalah menentukan nilai faktor pemutar (*twiddle factor*). Besarnya faktor pemutar ditentukan dengan persamaan berikut.

$$W_N^k = e^{-\frac{j2\pi k}{N}} \quad (1)$$

Nilai faktor pemutar yang telah didapatkan menggunakan persamaan (1), kemudian dihitung menggunakan metode kupu-kupu. Hasil dari proses perhitungan dengan metode kupu-kupu akan membagi data menjadi separuh dari jumlah cuplikan[5]. Apabila jumlah data belum memenuhi $N/2$ maka akan dihitung kembali dengan metode kupu-kupu.



Gambar 5 Diagram Alir Konversi ke dBm

Gambar 5 menunjukkan proses konversi nilai hasil FFT menjadi bentuk dBm. Hasil perhitungan FFT adalah berupa bilangan kompleks, sehingga belum bisa digunakan untuk menggambarkan level spektrum. Proses mengubah bilangan kompleks menjadi dBm ditunjukkan pada persamaan (2), dimana dBm adalah menunjukkan tingkatan sinyal audio yang terkandung dalam sinyal radio[6].

$$Gain_{dBm} = 10 \text{ Log}_{10} (N) \quad (2)$$

Setiap data FFT dilakukan konversi menjadi nilai dBm, jumlah data spektrum (dBm) harus sama dengan data FFT. Oleh karena itu dilakukan *increment* data spektrum ke N sampai semua data FFT berubah dalam satuan dBm. Selanjutnya nilai dBm didapat diplot ke dalam bentuk spektrum, sehingga diketahui level dari frekuensi.

2.2 Implementasi Sistem

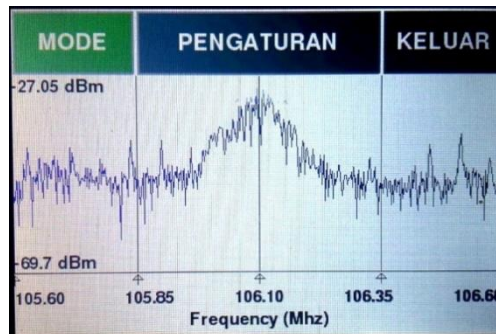
Implementasi sistem keseluruhan meliputi perakitan dan konfigurasi alat yang dipakai agar dapat mewujudkan sebuah *spectrum analyzer*. Gambar 6 menunjukkan implementasi keseluruhan perangkat keras setelah dirakit. Setelah semua perangkat dapat bekerja, langkah berikutnya adalah membuat program yang memiliki unjuk kerja *spectrum analyzer* menggunakan bahasa python.



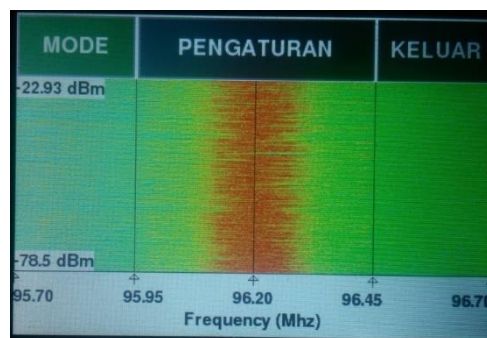
Gambar 6 Implementasi Keseluruhan Sistem

Gambar 7 menunjukkan implementasi program *spectrum analyzer* yang telah dibangun. Tampilan utama antarmuka program yang dibangun menampilkan spektrum frekuensi dengan tambahan 3 tombol menu. Tombol mode berfungsi untuk mengubah tampilan spektrogram menjadi *waterfall spectrum*. Tombol pengaturan digunakan untuk mengubah nilai frekuensi

tengah, *sample rate*, *gain* dan nilai maksimum minimum spektrum. Kemudian tombol keluar digunakan untuk keluar dari program



Gambar 7 Tampilan Antarmuka Halaman Utama



Gambar 8 Tampilan Antarmuka *Waterfall Spectrum*

Gambar 8 menunjukkan tampilan antarmuka *waterfall spectrum*. Tampilan *waterfall spectrum* menunjukkan bahwa adanya sinyal *carrier* pada frekuensi tertentu. Sinyal *carrier* adalah sinyal yang mempunyai level spektrum lebih tinggi dibanding level spektrum pada sinyal informasi. Warna oranye pada tampilan *waterfall spectrum* menunjukkan adanya sinyal *carrier* tersebut. Sedangkan warna hijau menunjukkan frekuensi sinyal informasi.

3. HASIL DAN PEMBAHASAN

3.1 Pengujian Pengaruh Jumlah Cuplikan

Pengujian ini dilakukan dengan memasukan jumlah cuplikan (N) yang bervariasi. Mulai dari N = 512 sampai N = 4.194.304. Jumlah cuplikan ini mempengaruhi penggunaan CPU dan memori dari Raspberry. Untuk mengetahui persentase penggunaan CPU pada Raspberry digunakan perintah `#top` pada sistem operasi raspbian. Gambar 9 adalah informasi penggunaan CPU pada Raspberry saat menjalankan komputasi FFT dengan 4194304 titik.

PID	USER	PR	N	VRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
4456	root	20	0	94592	67m	13m	R	100.0	7.8	0:55.64	python
1878	xrdp	20	0	20608	10m	2184	S	3.3	1.2	3:54.86	xrdp
2445	pi	20	0	47956	25m	18m	S	3.3	2.9	2:10.98	lxterminal
2343	pi	20	0	60932	22m	10m	S	2.6	2.6	2:14.31	Xvnc

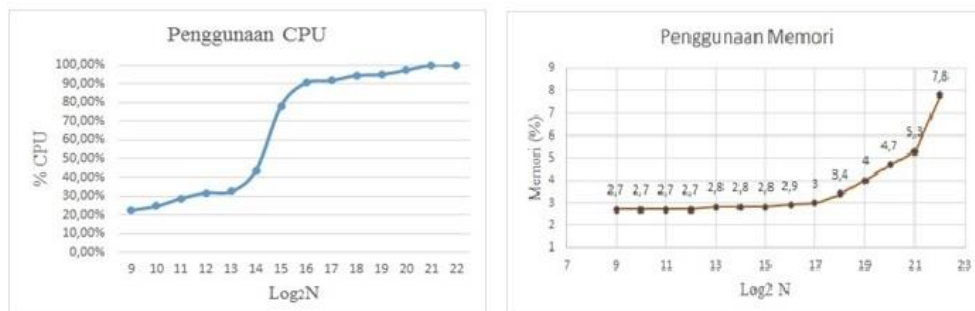
Gambar 9 Informasi Penggunaan CPU pada Raspberry Pi

Gambar 9 berisi informasi dari penggunaan prosesor dan memori (RAM) yang terpakai. Selain itu juga ada informasi mengenai waktu program berjalan, status program, memori fisik yang dipakai dan id program. Lebih jelas mengenai penggunaan CPU dan memori yang digunakan untuk menjalankan program dapat dilihat pada Tabel 1.

Tabel 1 Penggunaan CPU dan Memori untuk Setiap N titik

No	N	Log ₂ N	FFT		Penggunaan	
			Perkalian	Penjumlahan	CPU (%)	Memori (%)
1	512	9	2.304	4.608	22,5	2,7
2	1.024	10	5.120	10.240	24,6	2,7
3	2.048	11	11.264	22.528	28,7	2,7
4	4.096	12	24.576	49.152	31,8	2,7
5	8.192	13	53.248	106.496	32,5	2,8
6	16.384	14	114.688	229.376	43,8	2,8
7	32.768	15	245.760	491.520	78,3	2,8
8	65.536	16	524.288	1.048.576	90,5	2,9
9	131.072	17	1.114.112	2.228.224	91,8	3,0
10	262.144	18	2.359.296	4.718.592	94,5	3,4
11	524.288	19	4.980.736	9.961.472	95,0	4,0
12	1.048.576	20	10.485.760	20.971.520	97,3	4,7
13	2.097.152	21	22.020.096	44.040.192	100	5,3
14	4.194.304	22	46.137.344	92.274.688	100	7,8

Data pada Tabel 1 menunjukkan bahwa semakin besar nilai N maka akan semakin membebani kerja CPU. Hal ini disebabkan karena semakin banyak nilai N maka akan semakin banyak pula perhitungan kompleks yang dilakukan CPU. Perhitungan kompleks dari FFT menggunakan persamaan $(N/2)\text{Log}_2N$ untuk setiap perkalian dan $N\text{Log}_2N$ untuk penjumlahan kompleks[7]. Selain CPU, besarnya N juga mempengaruhi penggunaan memori sementara (RAM). Hal ini terjadi karena semakin banyak nilai N maka ruang penyimpanan yang harus disediakan juga semakin besar. Perubahan persentase dari CPU dan memori terhadap N ditunjukkan pada Gambar 10



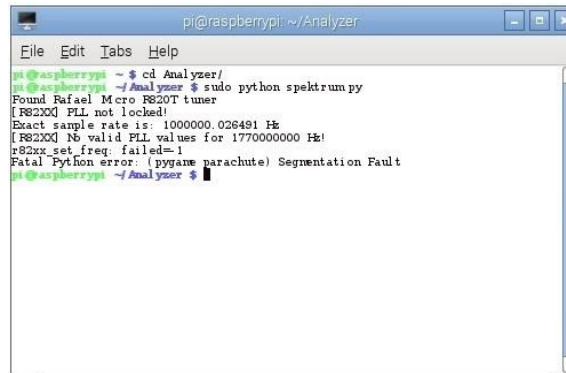
Gambar 10 Grafik Penggunaan CPU dan Memori Terhadap Jumlah Cuplikan

Gambar 10 menunjukkan bahwa kenaikan penggunaan CPU terjadi sangat besar ketika diberikan $N = 32.768$ titik yaitu sebesar 78,3%. Hal tersebut membuktikan bahwa kinerja CPU mulai terbebani dengan komputasi FFT yang menghasilkan 245.760 perkalian dan 491.520 penjumlahan kompleks. Lebih dari 32.768 titik nilai perkalian dan penjumlahan kompleks semakin besar, sehingga menyebabkan beban CPU mencapai lebih dari 90%. Setelah dilakukan pengujian pada aplikasi, beban CPU yang mencapai 90% membuat aplikasi berjalan lambat dan beberapa fungsi menjadi *error*. Oleh sebab itu penggunaan N antara 65.536 titik sampai 4.194.304 titik yang diimplementasikan pada Raspberry lebih baik dihindari.

Selain CPU, besarnya jumlah cuplikan (N) juga mempengaruhi penggunaan memori (RAM). Gambar 10 menunjukkan bahwa semakin besar nilai N maka penggunaan memori yang digunakan juga akan semakin besar. Hal ini terjadi karena adanya ruang memori yang digunakan untuk menyimpan hasil dari perhitungan FFT. Semakin banyak N maka komputasi yang dilakukan semakin banyak dan ruang penyimpanan yang harus disediakan juga semakin besar.

3.2 Pengujian Jangkauan Maksimal dan Minimal Frekuensi

Pengujian pengukuran jangkauan minimal dan maksimal frekuensi dilakukan dengan mengganti nilai frekuensi tengah pada program yang dibangun. Jangkauan frekuensi dari program tergantung pada kemampuan dari *tuner* yang dipakai. Pada sistem digunakan *tuner* tipe Rafael Micro R820T dengan jangkauan 24 MHz sampai 1.766 MHz [8]. Namun setelah dilakukan pengukuran, didapatkan hasil bahwa nilai maksimal frekuensi adalah 1.769 MHz. Apabila diberikan nilai frekuensi melebihi 1.769 MHz, maka sistem akan berhenti bekerja. Gambar 11 menunjukkan pesan *error* yang terjadi pada saat diberikan nilai frekuensi 1.770 MHz.



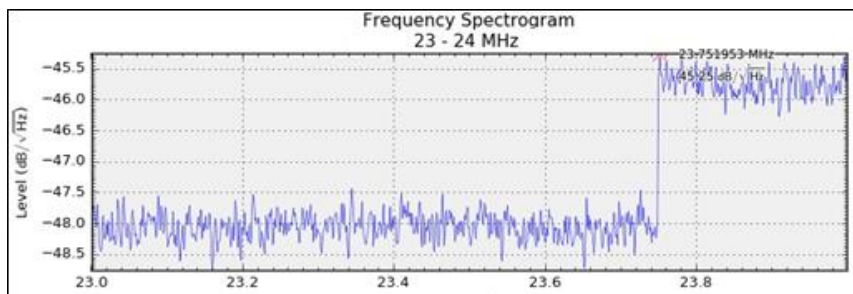
```

pi@raspberrypi: ~/Analyzer
File Edit Tabs Help
pi@raspberrypi ~$ cd Analyzer/
pi@raspberrypi ~/Analyzer $ sudo python spektrum.py
Found Rafael Micro R820T tuner
[R820T] PLL not locked!
Exact sample rate is: 1000000.026491 Hz
[R820T] No valid PLL values for 1770000000 Hz!
r82xx_set_freq: failed=-1
Fatal Python error: (pygame parachute) Segmentation Fault
pi@raspberrypi ~/Analyzer $

```

Gambar 11 Pesan *Error* Program

Sedangkan untuk batas jangkauan minimal frekuensi dari sistem adalah 24 MHz. Hal ini berdasarkan hasil pengukuran, apabila diberikan nilai frekuensi kurang dari 24 MHz maka program tidak dapat menampilkan adanya sinyal radio. Gambar 12 menunjukkan tampilan *waterfall spectrum* pada frekuensi tengah 23 MHz dengan *sample rate* 2 MHz.



Gambar 12 Tampilan Visualisasi Spektrum Nilai Minimal Frekuensi

Gambar 12 menunjukkan tampilan hasil visualisasi spektrum pada frekuensi 23 sampai 24 MHz.. Pada frekuensi 23,75 MHz level spektrum menurun jauh karena *tuner* tidak dapat melakukan *scanning* frekuensi dibawah 24 MHz. Sehingga penggunaan minimal frekuensi yang dianjurkan adalah 24 MHz atau lebih.

3.3 Pengujian Sinyal RF Pada Alat Komunikasi Radio

Salah satu fungsi dari *spectrum analyzer* adalah untuk menguji alat komunikasi radio baik pengujian saat produksi, perawatan maupun jika terjadi kerusakan[9]. Oleh karena itu untuk menguji kemampuan purwarupa *spectrum analyzer* yang telah dibuat, perlu adanya proses pengujian untuk menampilkan spektrum dari sinyal RF. Keluaran sinyal RF diperoleh dari alat komunikasi radio. Tabel 2 adalah hasil pengujian pengukuran sinyal RF menggunakan beberapa alat komunikasi radio. Terdapat 3 alat yang digunakan dalam pengujian ini yaitu *Handy Talky* (HT), *remote car* dan *remote control*.

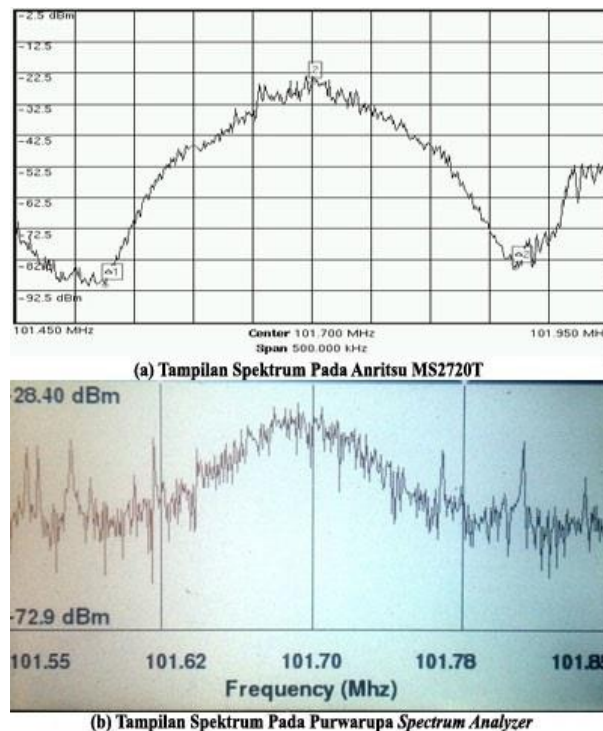
Tabel 2 Pengujian Daya Pancar Sinyal RF

No.	Alat	Pengukuran	
		Frekuensi	Level
1.	HT Olinca TH-888A	142,1 MHz	-26,74 dBm
2.	HT Baofeng UV5R	520 MHz	-26,78 dBm
3.	<i>Remote Car</i>	433,94 MHz	-22,63 dBm
4.	<i>Remote Control</i>	27.15 MHz	-22,83 dBm

Hasil pengujian Tabel 2 membuktikan bahwa purwarupa yang dibangun mampu menunjukkan nilai frekuensi dan level dari sinyal yang dipancarkan oleh alat komunikasi radio. Purwarupa yang dibangun dapat menampilkan level spektrum dari frekuensi VHF sampai UHF. Saat pengujian daya pancar, alat pengujian harus diletakan pada jarak yang dapat dideteksi oleh purwarupa *spectrum analyzer*.

3.4 Pengujian Perbandingan Pengukuran Spectrum Analyzer

Pengujian ini dilakukan dengan cara membandingkan hasil spektrum dari sistem yang dibangun dengan *spectrum analyzer* pabrikan. Tujuan dari pengukuran adalah untuk mengetahui tingkat sensitifitas sistem terhadap sinyal radio yang dipancarkan oleh stasiun pemancar. Tipe *spectrum analyzer* yang dipakai untuk membandingkan adalah Anritsu MS2720T. Contoh perbandingan pengukuran spektrum dari Anritsu MS2720T dengan sistem ditunjukkan pada Gambar 13.

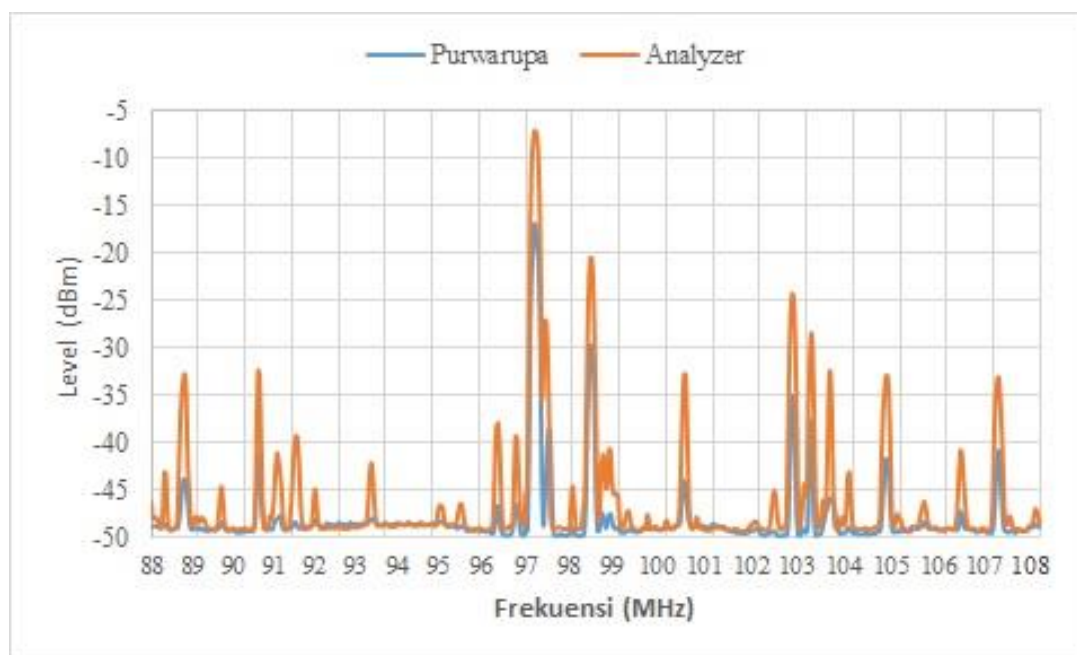


Gambar 13 Perbandingan Power Spektrum

Gambar 13 menampilkan visualisasi spektrum dari stasiun radio swaragama FM (101,7 MHz). Pada Anritsu MS2720T *power* spektrum dari frekuensi 101,7 adalah -25,66 dBm, sedangkan pada purwarupa tercatat -28,40 dBm. Kemudian dilakukan pengukuran terhadap sinyal radio FM lain yaitu pada jangkauan 88 sampai 107 MHz[10]. Hasil perbandingan

pengukuran spektrum antara Anritsu MS2720T dengan purwarupa ditunjukkan pada Gambar 14.

Terdapat perbedaan *power* spektrum diantara keduanya. Pada MS2720T memiliki nilai *power* yang besar dibanding purwarupa. Hal ini terjadi akibat antenna penerima yang dipakai oleh Anritsu MS2720T berada pada tower dengan ketinggian 10 meter sehingga mampu menerima sinyal lebih kuat. Namun meskipun memiliki *power* spektrum yang berbeda-beda, frekuensi sinyal radio yang ditunjukkan kedua alat adalah sama.



Gambar 14 Perbandingan Pengukuran Spektrum dengan Anritsu MS2720T

Selain ketinggian antenna, perbedaan *power* spektrum pada Gambar 14 juga disebabkan oleh perbedaan jenis antenna. Pada purwarupa menggunakan antenna omni yang menangkap sinyal dari berbagai arah (360°), sedangkan pada Anritsu MS2720T digunakan antenna *grid* yang dapat menangkap sinyal sesuai antenna diarahkan (180°). Sehingga ada beberapa nilai *power spectrum* pada Anritsu MS2720T yang lebih besar dibandingkan pada purwarupa karena antenna *grid* tepat mengarah pada stasiun pemancar.

4. KESIMPULAN

Berdasarkan dari hasil penelitian dan pembahasan yang dilakukan maka diperoleh kesimpulan sebagai berikut:

1. Purwarupa *spectrum analyzer* yang dibangun mampu menampilkan spektrum frekuensi sinyal radio dengan lancar pada jumlah cuplikan dari 512 sampai 32.768 titik. Sebab penggunaan jumlah cuplikan lebih dari 32.768 titik akan membebani cpu dan memori dari SBC, sehingga menyebabkan sistem menjadi lambat.
2. Hasil dari pengujian menunjukkan bahwa jangkauan minimal frekuensi dari sistem adalah 24 MHz, sedangkan jangkauan maksimal frekuensi adalah 1.769 MHz.
3. Hasil perbandingan pengukuran menggunakan *spectrum analyzer* Anritsu MS2720T menunjukkan bahwa sistem dapat menunjukkan letak spektrum frekuensi radio FM secara tepat.

5. SARAN

Untuk pengembangan penelitian lebih lanjut, diberikan saran sebagai berikut:

1. Purwarupa *spectrum analyzer* 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).
2. Perlu adanya peningkatan spesifikasi RTL-SDR yang digunakan. Pada penelitian ini menggunakan *chip* R820T, mungkin dapat diganti menggunakan *chip* yang spesifikasinya lebih tinggi seperti E4000 atau R820T2.
3. Perlu dilakukan penelitian lanjutan untuk menambah fasilitas yang ada pada purwarupa *spectrum analyzer*. Misalnya fasilitas untuk mengatur *resolution bandwidth* (RBW), penambahan *marker* untuk mengetahui level dari beberapa sinyal radio dll.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberi dukungan baik secara moril maupun materiil terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] Rentanu, I. W., 2012, Audio Spectrum Analyzer Menggunakan Mikrokontroler Atmega32, *Tugas Akhir*, Jurusan Pendidikan Teknik Elektronika, Univ. Negeri Yogyakarta, Yogyakarta.
- [2] Yogiswara, P. A., Pranowo, F. X. H., Budiman, L. H., 2010, Perancangan dan Implementasi Sistem Perangkat Lunak Client Server Untuk Alat Ukur Spectrum Analyzer, *Skripsi*, Jurusan Ilmu Komputer, Univ. Bina Nusantara, Jakarta.
- [3] Cooley, J.W., dan Tukey, J.W., 2000, An Algorithm for The Machine Calculation of Complex Fourier Series. *Mathematics Of Computation Journal*, Vol 1, hal 297-301.
- [4] Sierra, E. G., 2015, Low Cost SDR Spectrum Analyzer and Analog Radio Receiver Using GNU Radio, Raspberry Pi 2 and SDR-RTL Dongle, *Proceeding of 7th IEEE Latin-American Conference on Communication*, Arequipa, 4 - 6 Nov.
- [5] Quraeshi, F., 2009, Analysis of Twiddle Factor Memory Complexity of Radix 2 Pipelined FFTs, *Proceeding Conference Record of The Forty-Third Asilomar Conference on Signal Systems and Computers*, California, 1 - 4 Nov.
- [6] Cerna, M., Harvey, A.F., 2000, The Fundamentals of FFT-Based Signal Analysis and Measurement in LabView dan LabWindows, <http://www.ni.com/white-paper/4278/en/>, diakses tgl 7 Juni 2016.
- [7] Lyons, R. G., 2001, *Understanding Digital Signal Processing*, Prentice Hall International, New Jersey
- [8] Laufer, C., 2014, The Hobbyist's Guide to the RTL-SDR : Really Cheap Software Defined Radio, <http://www.qsl.net/docs/TheHobbyistguideToRTL-SDR.pdf>, diakses tgl 15 Februari 2016.

- [9] Marpanaji, E., 2012, Aplikasi Platform Komputasi Software-Define Radio (SDR) untuk Digital Spectrum Analyzer, *Prosiding Pertemuan Ilmiah XXV HFI Jateng & DIY*, Purwokerta, 9 April.
- [10] Hioki, W., 1998, *Telecommunications, third edition*, Prentice Hall International, New Jersey.