

ARSITEKTUR SOFTWARE-DEFINED RADIO (SDR)

Eko Marpanaji, Bambang Riyanto T., Armein Z.R. Langi, Adit Kurniawan,
Andri Mahendra

Teknik Elektro STEI ITB

ekoaji332@students.itb.ac.id, briyanto@lskk.ee.itb.ac.id, langi@lss.ee.itb.ac.id
adit@ltrgm.ee.itb.ac.id

Abstrak

Software-Defined Radio (SDR) menawarkan sebuah sistem komunikasi nirkabel yang fleksibel dan dapat dikonfigurasi ulang sehingga perubahan standar dapat dilakukan pada perangkat lunaknya saja. Sistem ini sangat cocok untuk membangun sistem komunikasi yang relatif murah untuk daerah pegunungan dan pedesaan yang sulit dijangkau oleh jaringan kabel. SDR juga mendukung pengembangan komunikasi seluler dan bergerak untuk generasi mendatang, Next Generation Networks (NGN), serta Rural Next Generation Networks (R-NGN).

Tingkat fleksibilitas SDR sangat ditentukan oleh arsitektur SDR. Makalah ini akan menguraikan arsitektur dan perkembangan teknologi SDR serta hasil penelitian awal tentang contoh-contoh aplikasi SDR untuk komunikasi. Harapannya, dengan menguasai teknologi SDR diharapkan dapat menjadi sarana dalam pengembangan teknologi komunikasi nirkabel di tanah air di masa datang.

Kata Kunci: *software-defined radio, arsitektur, usrp, sinyal IF, modulasi, demodulasi, perangkat lunak.*

1. PENDAHULUAN

Perubahan standar komunikasi yang begitu cepat menjadi masalah yang serius bagi *provider*, karena sebagian besar teknologi komunikasi yang dikembangkan diimplementasikan dalam bentuk perangkat keras[1]. Perubahan standar komunikasi selalu mendorong investasi perangkat keras yang baru dan akan mengakibatkan mahalnya biaya langganan, sehingga masyarakat tingkat ekonomi menengah kebawah tidak mampu menikmati layanan yang diberikan.

Indonesia sebagian besar terdiri dari daerah pedesaan (*rural*) dan pegunungan dengan kondisi masyarakat yang memiliki tingkat ekonomi menengah kebawah, dan sebagian besar belum bisa menikmati layanan komunikasi. Sistem komunikasi nirkabel yang murah merupakan

teknologi alternatif yang sangat diperlukan untuk mengatasi masalah tersebut.

Software-Defined Radio (SDR) merupakan teknologi komunikasi berbasis nirkabel yang ditentukan oleh perangkat lunak dalam menjalankan fungsinya. SDR memiliki sifat fleksibel dan dapat dikonfigurasi ulang sehingga perubahan standar atau fungsi radio dapat dilakukan pada perangkat lunak tanpa harus mengganti perangkat kerasnya.

Kedepan, sistem komunikasi nirkabel berbasis SDR sangat diperlukan dalam pengembangan sistem komunikasi seluler generasi ke-3 (3G) seperti WCDMA dan UMTS, generasi selanjutnya atau *Next Generation Networks* (NGN), serta *Rural Next Generation Networks* (R-NGN). Teknologi SDR juga sangat

cocok untuk pengembangan jaringan komunikasi nirkabel di negara kita.

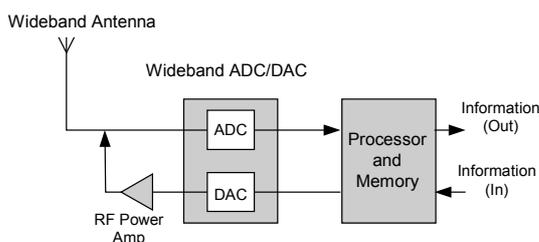
Makalah ini akan menjelaskan arsitektur SDR, perkembangan teknologi SDR serta hasil penelitian awal tentang SDR yang menjawab pertanyaan penelitian (*research question*): Bagaimanakah arsitektur SDR? Dapatkah arsitektur dibangun menggunakan komponen komoditas?

Sistematika makalah ini adalah: (1) pendahuluan; (2) arsitektur SDR; (3) desain arsitektur SDR; (4) implementasi SDR; (5) aplikasi: radio penerima SDR menggunakan usrp; dan (6) kesimpulan.

2. ARSITEKTUR SDR

Software-Defined Radio (SDR), ada yang menyebut juga *Software Radio* (SWR), diperkenalkan pertama kali pada tahun 1991 oleh Joseph Mitola[2][3]. Istilah SDR ini digunakan untuk menunjuk sebuah kelas radio yang dapat dikonfigurasi ulang atau diprogram ulang[4], sehingga menghasilkan sebuah jenis perangkat komunikasi nirkabel dengan *mode* dan band frekuensi ditentukan oleh fungsi perangkat lunak.

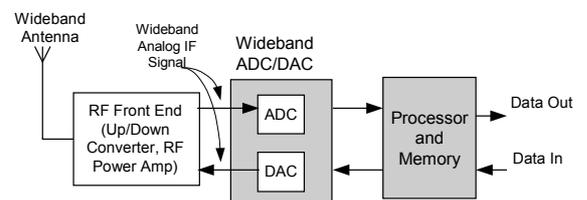
Secara ideal, SDR menawarkan fleksibilitas (*flexibility*), dapat dikonfigurasi ulang (*reconfigurability*), memungkinkan untuk perluasan (*scalability*), dan bahkan memiliki mode sebanyak mungkin (*multi mode*). Arsitektur SDR secara ideal ditunjukkan pada Gambar 1 berikut ini.



Gambar 1. Arsitektur SDR Ideal

Arsitektur SDR ideal akan menempatkan ADC/DAC sedekat mungkin dengan antenna 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[2][4][5][6]. Namun demikian, keterbatasan teknologi dan mahalnya wideband ADC/DAC mendorong untuk sedikit mengubah arsitektur SDR dalam menempatkan ADC/DAC sehingga menjadi realistis seperti yang ditunjukkan pada Gambar 2 berikut ini.



Gambar 2. Arsitektur SDR Realistis

Arsitektur SDR yang lebih realistis menempatkan wideband ADC/DAC setelah Down Converter/Up Converter, sehingga konversi analog ke digital atau sebaliknya berkaitan dengan sinyal *Intermediate Frequency* (IF) yang lebih rendah dibanding sinyal *Radio Frequency* (RF). Arsitektur ini saat ini banyak dikembangkan dan dalam proses penelitian untuk implementasinya. Adapun aturan-aturan yang digunakan dalam proses perancangan arsitektur SDR akan dijelaskan pada bagian berikut ini.

3. DESAIN ARSITEKTUR SDR

Desain arsitektur SDR meliputi desain perangkat keras dan perangkat lunak. Mitola (2000) menyatakan ada 7 (tujuh) tahapan yang harus dilakukan dalam membangun SDR, yaitu: (1) perancangan antenna jalur lebar (*wide band antenna*); (2) rancangan frekuensi RF disesuaikan dengan kemampuan ADC-DAC sehingga diperoleh unjuk kerja yang baik; (3) pemilihan ADC/DAC untuk sinyal jalur lebar; (4) membangun interkoneksi jalur lebar untuk menghubungkan ADC/DAC dengan prosesor; (5) membangun

prosesor untuk pengolahan sinyal digital, termasuk proses modulasi dan demodulasi; (6) merancang perangkat lunak, diutamakan berbasis obyek termasuk *message passing* dan *encapsulation*; (7) merancang aplikasi.

SDR memiliki keuntungan karena sifat fleksibilitas (*flexibility*), lengkap dan dapat dikonfigurasi ulang secara mudah (*complete and easy reconfigurability*), serta dapat diskala (*scalability*)[2]. Sedangkan Cristensen (2004) menyatakan bahwa isu kunci utama sebuah platform SDR *embedded* adalah sifat fleksibilitas (*flexibility*), dapat diperluas (*expandability*), dapat diskala (*scalability*), dapat dikonfigurasi ulang (*reconfigurability*), dan dapat diprogram ulang (*reprogrammability*). Sifat-sifat ini harus muncul dalam produk akhir dalam desain sebuah SDR[7].

Berdasarkan prinsip-prinsip desain tersebut, banyak pakar melakukan penelitian tentang arsitektur SDR. Gweon (2005) mengembangkan arsitektur *reconfigurable* SDR berbasis W-CDMA dan CDMA menggunakan DSP TMS320C6416-600 MHz. Modulasi-demodulasi W-CDMA dan CDMA masih diimplementasikan dalam perangkat keras, sehingga terbatas untuk dual-mode[8]. Bose (1999) mengembangkan arsitektur SDR menggunakan komputer PC (*general purpose processor*), yang kemudian disebut dengan "virtual radio"[9]. Bose menyarankan untuk mengkaji lebih jauh lagi tentang rancangan filter untuk menangani digitasi sinyal RF jalur lebar (*wideband*) serta pemrosesan sinyal digital secara paralel. Corneloup (2002) mengimplementasikan arsitektur SDR pada board DSP SHARC, dengan pendekatan pemrosesan paralel (sistem multiprosesor) dibawah kendali sistem operasi real-time[10].

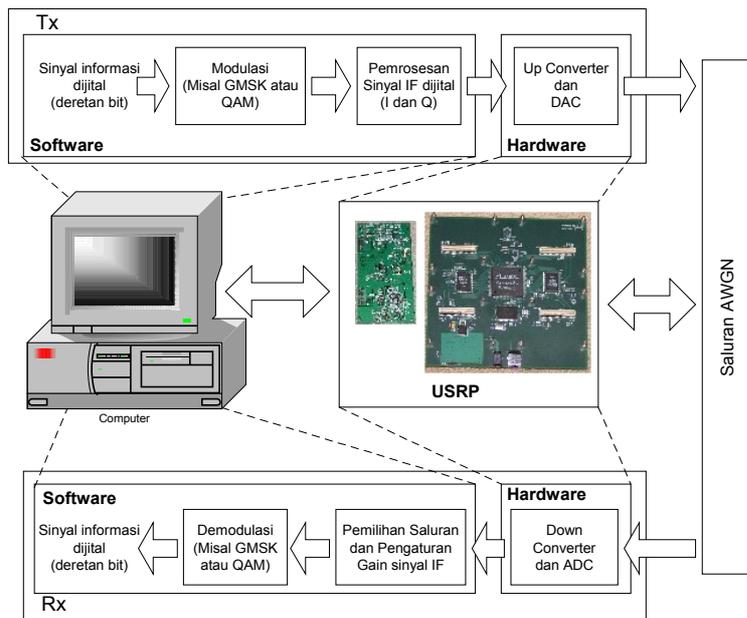
Proses pemilihan saluran (*channelization*) juga mendapat perhatian khusus para peneliti karena menentukan fitur multi-channel ataupun multi-band sebuah SDR. Harris (2003)

menggunakan bank filter polifase dalam pemilihan saluran sistem AMPS, komunikasi radio FM dan SSB[11]. Dick (2001) menggunakan FPGA sebagai prosesor pemilihan saluran[12]. Pucker (2002) membandingkan tiga model proses pemilihan saluran yaitu *Digital Down Converter* (DDC), *Frequency Domain Filtering* (FDF) dan *Polyphase FFT Filter Bank* (PFFB) dengan melihat laju kesalahan bit (*bit error rate or BER*). Hasil yang diperoleh bahwa BER tidak dipengaruhi oleh ketiga arsitektur. Untuk keperluan optimasi, DDC paling tidak efisien untuk saluran yang sangat banyak, PFFB cocok untuk struktur saluran yang memiliki redudansi dan FDF cocok untuk kebutuhan fleksibilitas[13].

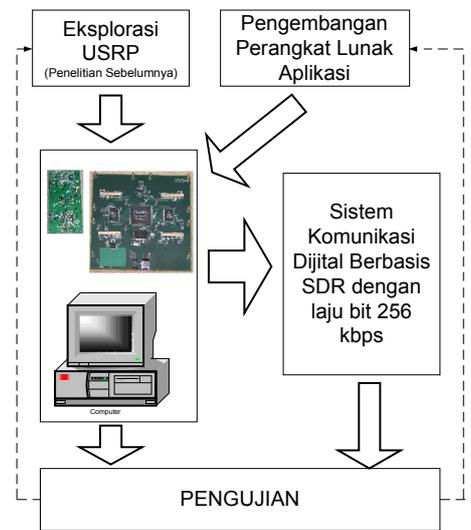
Isu penting dalam desain SDR adalah kemampuan ADC/DAC dan kapasitas atau kecepatan komputasi prosesor untuk pengolahan sinyal digital. Kedua isu ini akan menentukan unjuk kerja SDR. Unjuk kerja SDR dapat dilihat dari *Quality of Service* (QoS), yaitu *bit rate*, *bit error rate* (BER), *packet loss rate* (PLR), dan *delay spread*[4]. Kapasitas komputasi prosesor biasanya dinyatakan dalam *millions operations per second* (MOPS) dan dihitung berdasarkan kebutuhan komputasi untuk pengolahan sinyal digital.

4. IMPLEMENTASI SDR

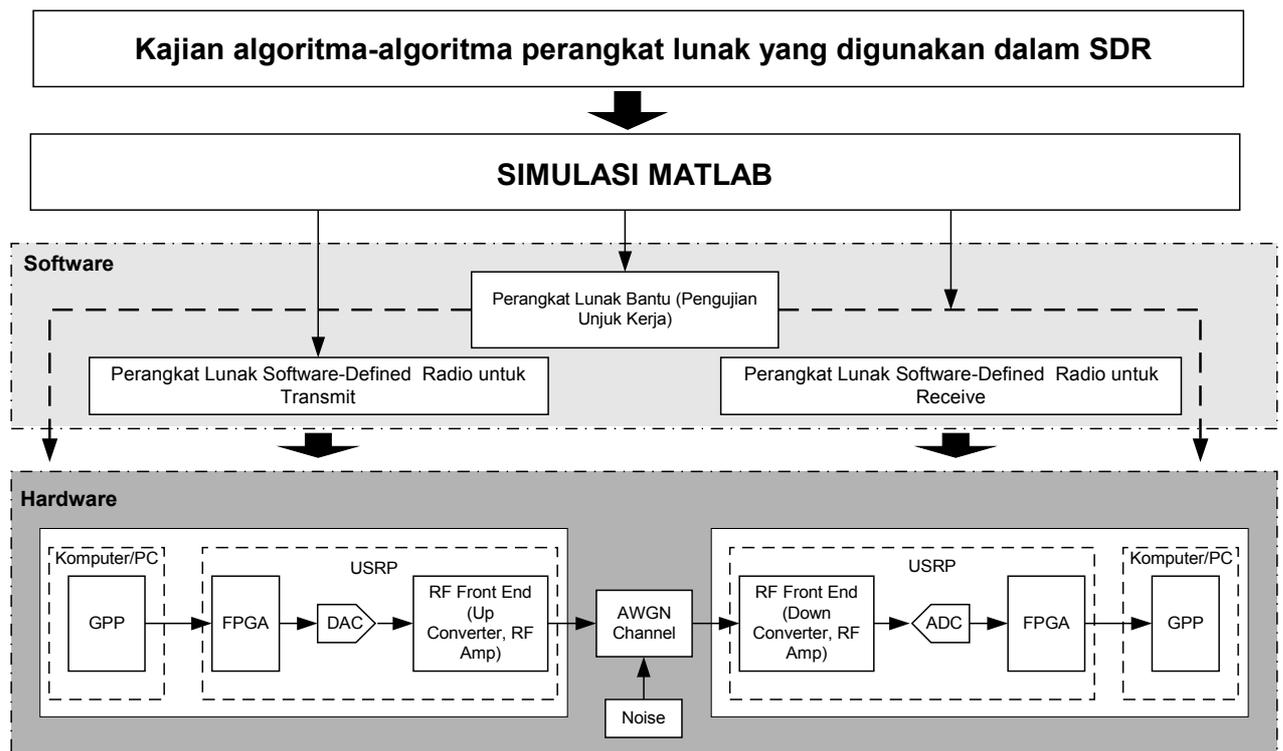
Pengembangan arsitektur SDR mempertimbangkan aspek ekonomi, sehingga dalam penelitian ini menggunakan arsitektur SDR realistik seperti yang ditunjukkan Gambar-2. Arsitektur ini menempatkan ADC/DAC setelah rangkaian Down Converter/Up Converter, sehingga pemrosesan dilakukan pada sinyal IF bukan sinyal RF. Cara ini akan menurunkan aspek komputasi. Kapasitas komputasi prosesor dapat diturunkan dari kebutuhan pemrosesan sinyal digital untuk sistem penerima dan sistem pemancar seperti yang ditunjukkan pada Gambar 3 berikut ini.



Gambar 3. Pemrosesan sinyal digital



Gambar 4. Metodologi Penelitian Arsitektur SDR



Gambar 5. Langkah-langkah dalam pembuatan prototype

Sebagai langkah awal penelitian, ujung depan SDR yang dikembangkan menggunakan *Universal Software Radio Peripheral (USRP)* yang dikeluarkan oleh GNU Radio dengan komputer pribadi (PC) sebagai tulang punggung pemrosesan sinyal seperti yang ditunjukkan

Gambar 4. Alasan pemilihan sistem yang dikembangkan ini adalah kemudahan dalam pengembangan perangkat lunak baik proses *coding* maupun *testing*, serta banyak perangkat lunak *open source* untuk pengembangan

perangkat lunak. Perangkat lunak menggunakan sistem operasi Linux.

Unjuk kerja dapat diukur berdasarkan konsumsi daya, biaya relatif dan kemampuan komputasi atau menggunakan profile QoS (Quality of Service) seperti: BER <math>< 10^{-3}</math>, Packet Loss Rate <math>< 10^{-2}</math>, Delay Spread <math>< 100</math> ms dan GoS (Grade of Service) > 95% (Mitola, 2000). Angka-angka yang digunakan sebagai patokan tersebut disesuaikan rancangan aplikasi sistem SDR yang akan dikembangkan.

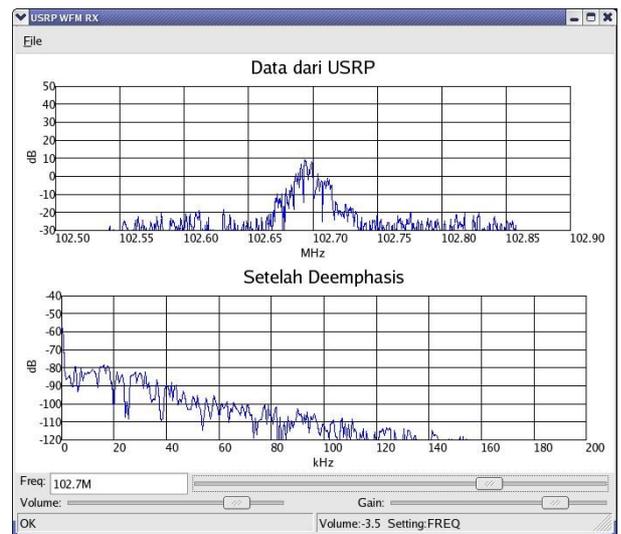
Kegiatan dalam pengembangan arsitektur SDR meliputi simulasi MATLAB pengembangan perangkat lunak aplikasi dan pengujian, implementasi algoritma dalam bahasa pemrograman (menggunakan python dengan sistem operasi Linux), dan pengujian. Secara lengkap tahapan kegiatan tersebut ditunjukkan pada Gambar 5.

5. APLIKASI: RADIO PENERIMA SDR MENGGUNAKAN USRP

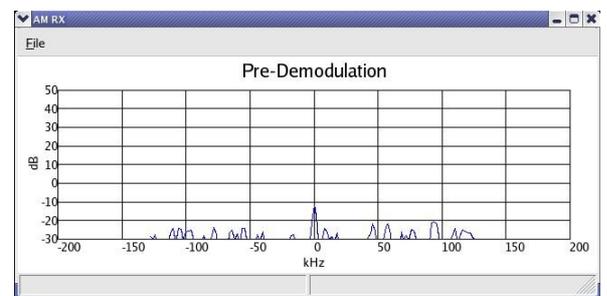
Dalam penelitian ini, SDR dengan menggunakan USRP dan PC digunakan untuk menangkap siaran radio FM broadcast. Hasilnya ditunjukkan pada Gambar 6, yang menampilkan grafik data digital yang dihasilkan oleh USRP dan grafik sinyal deemphasis. Gambar tersebut tampilan siaran stasiun radio FM 102.7 MHz.

SDR juga dapat digunakan untuk menangkap radio AM, seperti yang ditunjukkan pada Gambar 7.

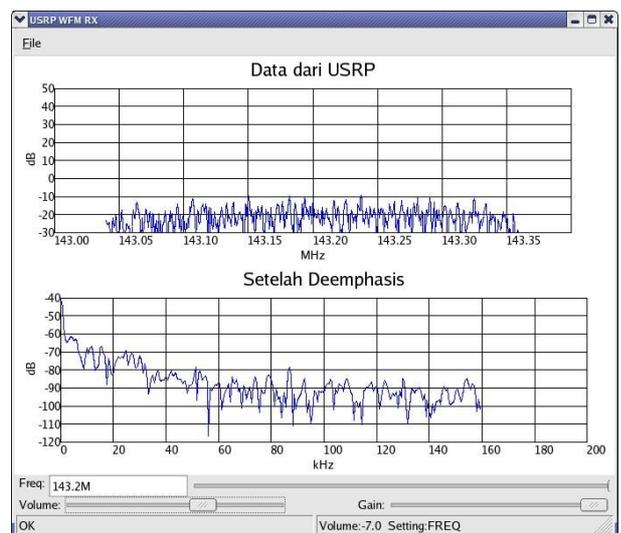
Berdasarkan contoh-contoh tersebut, sistem SDR dicoba untuk dikembangkan menjadi pesawat radio komunikasi 2m dengan jangkauan frekuensi 140-150 MHz, meskipun masih terbatas untuk sistem penerima. SDR digunakan untuk menangkap sinyal RF dari sebuah HT 2m dengan frekuensi 143.2 MHz, seperti yang ditunjukkan pada Gambar 8 dan Gambar 9.



Gambar 6. Penerima FM berbasis SDR (Stasiun FM 102.7 MHz)



Gambar 7. Penerima radio AM

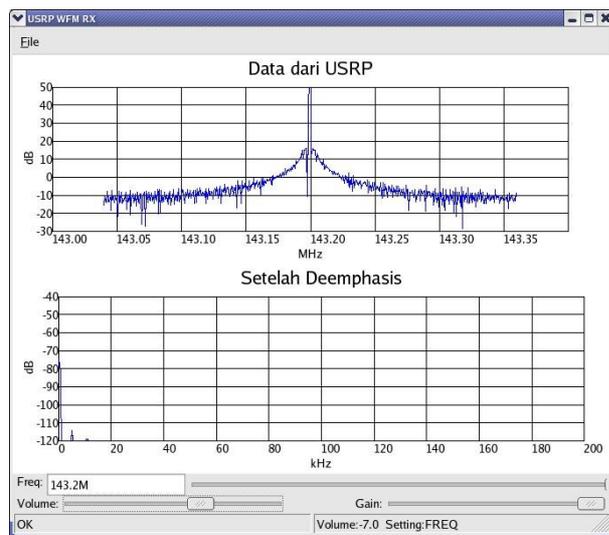


Gambar 8. Penerima FM 2m frekuensi 143.200 MHz tanpa carrier (tidak ada sinyal)

Gambar 8 menunjukkan tampilan sinyal dari USRP untuk frekuensi 143.200 MHz saat tidak ada sinyal pemancar. Sedangkan Gambar 9 menunjukkan sinyal USRP pada saat ada sinyal

pemancar (transmit dari HT) pada frekuensi 143.200 MHz.

Penelitian ini akan terus dikembangkan untuk sistem komunikasi nirkabel lainnya, termasuk jenis-jenis modulasi digital yang akan digunakan untuk membangun sistem komunikasi data digital seperti modulasi GMSK, QAM atau yang lainnya.



Gambar 9. Penerima FM 2m frekuensi 143.200 MHz saat ada sinyal pemancar

6. KESIMPULAN

Berasarkan uraian diatas, maka dapat disimpulkan:

- Komponen-komponen penting untuk membangun arsitektur SDR adalah rangkaian analog RF, ADC/DAC untuk sinyal IF atau RF, dan prosesor sebagai perangkat kerasnya, serta perangkat lunak untuk menjalankan fungsi radio.
- Aspek komputasi dapat diturunkan dari kompleksitas algoritma perangkat lunak dan akan menentukan kapasitas komputasi prosesor yang digunakan
- Fleksibilitas SDR sangat ditentukan oleh partisi dalam desain perangkat keras dan perangkat lunak arsitektur SDR

7. DAFTAR PUSTAKA

- [1] Shah, Alok, "An Introduction to Software Radio", [Online] <http://www.vanu.com/publications/SWRprimer.pdf>, 2002.
- [2] Reed, Jeffrey H, "Software Radio: A Modern Approach to Radio Engineering", New Jersey, Prentice Hall, 2002.
- [3] Steinheider, J., "Software-defined Radio Comes of Age", *Mobile Radio Technology*, Feb 1st, [Online] http://www.vanu.com/resources/intro/software-defined_radio_comes_of_age.html, 2003.
- [4] Mitola, Joseph III, "Software Radio Architecture. Object-Oriented Approaches to Wireless Systems Engineering", Canada, John Eiley & Sons, Inc, 2000.
- [5] Lehr, W., "Software Radio: Implication for Wireless Services, Industry Structure, and Public Policy", [Online] http://itc.mit.edu/itel/docs/2002/Software_Radio_Lehr_Fuencis.pdf, 2002.
- [6] Gutttag, J., "Software Radio for Adaptive Networking", [Online] http://web.mit.edu/deshpandecenter/downloads/presos/ideastream2003_wireless.pdf, 2003.
- [7] Christensen, Flemming, "A scalable Software-Defined Radio Development System". [On-line] http://www.xilinx.com/publications/xcellonline/xcell_51/xc_pdf/xc_es-sundance51.pdf, 2004.
- [8] Gweon-Do JO, "A DSP-Based Reconfigurable SDR Platform for 3G Systems", *IEICE Transaction on Communication*, Vol.E88-B No.2 Feb. 2005.
- [9] Bose, Vanu G., "Design and Implementation of Software Radios Using a General Purpose Processors", [Online] <http://www.sigmobile.org/phd/1999/theses/bose.pdf>. 1999.
- [10] Corneloup, M., "Open Architecture for Software Defined Radio Systems", [Online] <http://www.csem.ch/slats/files/AmblesideCom-Final.pdf> - Similar pages, 2002.
- [11] Harris, F.J., Dick, C., and Rice, M., "Digital Receivers and Transmitter Using Polyphase Filter Banks for Wireless Communications", *IEEE Transaction on Microwave Theory and Techniques*, Vol. 51, No.4, 1395-1412, 2003.
- [12] Dick, C.H., "Design and Implementation of High-Performance FPGA Signal Processing Datapath for Software Define Radios", [Online] http://www.eetasia.com/ARTICLES/2001AUG/2001AUG09_ICD_AMD_RFD_TAC01.PDF, 2001.
- [13] Pucker, L., "Channelization Techniques for Software Radio", [Online] http://www.spectrumsignal.com/channel_techniques/Channelization_Paper_SDR_forum.pdf, 2002.