

Pengembangan Universal Audio Scrambler Menggunakan Teknik *Orthogonal Frequency Division Multiplexing* (OFDM)

¹M. Budiman, ²Riyanto, ³Y. Fitri, ⁴A. Adnan

Program Studi Fisika – Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Bandung, Jalan Ganesa 10 Bandung, 40132

¹*mamanbudiman@yahoo.com*

²*riyanto.fisika@gmail.com*

³*yulia_fitri18@yahoo.com*

⁴*ias_aca@yahoo.com*

Received date : 20 Desember 2010

Accepted date : 25 Februari 2011

Abstrak

Telah didesain sebuah alat universal audio scrambler menggunakan teknik OFDM yaitu suatu teknik modulasi yang menggunakan banyak frekuensi (*multicarrier*) yang saling tegak lurus (*orthogonal*). Teknik ini dilakukan dengan menggunakan transformasi Fourier balik (*Invers Fast Fourier Transform/IFFT*) yang mengubah domain frekuensi menjadi domain waktu pada bagian pemancar (transmitter) sedangkan pada penerima menggunakan transformasi Fourier (*Fast Fourier Transform/FFT*) untuk mendapatkan kembali sinyal dalam domain frekuensi. Frekuensi output dari transmitter didesain berada pada daerah frekuensi audio dengan *bandwidth* 2,4 kHz sehingga dapat diterapkan pada berbagai jenis alat komunikasi. Alat ini menggunakan kompresi suara CVSD dengan rasio 1:8 dan mapping 16PSK untuk memperkecil jumlah bit data. Kecepatan transfer data (*data rate*) alat ini adalah 8 kbps. Desain juga diuji performanya terhadap variasi *noise* pada kanal transmisi dan terhadap frekuensi offset akibat efek *doppler*. Hasilnya, desain bekerja baik pada SNR di atas 60 dengan toleransi frekuensi offset sebesar 5 Hz.

Kata kunci: *audio scrambler, OFDM, IFFT/FFT, 16PSK, CVSD.*

Abstract

Universal audio scrambler has been designed using OFDM technique, a modulation technique which uses multi-frequency (multi-carrier) that is orthogonal to each other. This technique is performed using Invers Fast Fourier Transformation (IFFT), which changes the frequency domain into time domain at the transmitter, while the receiver using Fast Fourier Transformation (FFT) to get the signal back in the frequency domain. The output frequency of the transmitter is designed to be in the audio frequency area (2.4 kHz bandwidth), so that it can be applied to various types of communication tools. This tool uses CVSD voice compression with 1:8 ratios and 16PSK mapping to minimize the number of data bits. Data transfer speed (data rate) for this tool is 8 kbps. The design was also tested its performance against the variation of noise in transmission channels and the frequency offset due to Doppler effect. As a result, the design works well at SNR above 60 with a tolerance offset frequency of 5 Hz.

Keywords: *audio scrambler, OFDM, IFFT/FFT, 16PSK, CVSD.*

1 Pendahuluan

Telekomunikasi berkaitan erat dengan proses pertukaran informasi, data dan konten multimedia. Salah satu media telekomunikasi yang sudah dipakai masyarakat luas adalah telepon atau komunikasi radio. Pada pembicaraan melalui telepon atau komunikasi radio tersebut terjadi pertukaran informasi yang terkadang bersifat rahasia misalnya pada bidang militer, pemerintahan dan bisnis. Untuk itu diperlukan sistem pengamanan pembicaraan yang bebas dari penyadapan, sehingga informasi pembicaraan terlindungi dari pihak yang

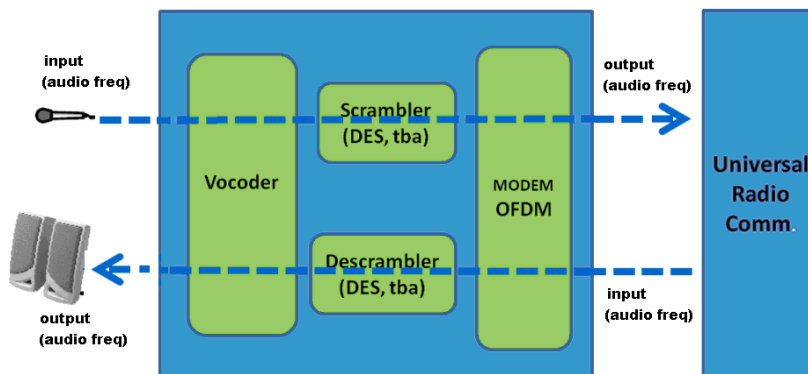
tidak berhak menerima. Salah satu teknik pengamanan yang digunakan adalah pengacakan sinyal suara (*scrambling*) dengan mengubah urutan data suara [1]. Kelemahan metode ini adalah mempunyai waktu proses yang lebih besar dari waktu sampling sehingga tidak dapat diterapkan untuk sistem real time.

Pada penelitian ini, proses *scrambling* dilakukan secara digital. Sinyal suara diubah kedalam sinyal digital/biner(bit) kemudian data biner ini diacak dengan algoritma *scrambler*. Data hasil *scrambler* selanjutnya dimodulasi dengan frekuensi tertentu. Agar dapat diaplikasikan untuk semua sistem telekomunikasi (*universal*) maka modulasi dilakukan pada frekuensi audio. Penggunaan frekuensi audio mengakibatkan terbatasnya laju pengiriman data (*data rate*) karena frekuensinya rendah. Masalah ini diatasi dengan modulasi yang menggunakan banyak frekuensi yaitu teknik OFDM (*Orthogonal Frequency Division Multiplexing*) [2]. Teknik OFDM telah banyak digunakan pada sistem telekomunikasi berkecepatan tinggi dan telah dijadikan standar *Digital Audio Broadcast* (DAB) dan *Digital Video Broadcast* (DVB). Selain itu, teknik ini juga digunakan pada *High Bit-rate Digital Subscriber Lines* (HDSL), *Very High Speed Digital Subscriber Lines* (VHDSL), *High Definition Television* (HDTV), *Worldwide Interoperability for Microwave Access* (WiMAX) dan komunikasi radio[3].

Kompresi sinyal suara juga dilakukan untuk memperkecil jumlah bit data yang harus ditransmisikan. Dalam desain *universal audio scrambler* ini digunakan kompresi CVSD (*Continuously Variable Delta Slope Modulation*) . Kompresi jenis ini dapat mengubah *bitrate* suara dari 64kbps menjadi 8 kbps.

Audio scrambler ini diimplementasikan pada OMAP-L137 *Evaluation Modul* dengan *Digital Signal Processor* (DSP) tipe TMS320C6747 *floating point* produk dari *Texas Instrument* [4]. Sistem ini dimodelkan dan disimulasikan menggunakan *software* MATLAB versi 7.9 dan *digenerate* ke dalam bahasa C menggunakan *software* CCS (*Code Composer Studio*) versi 3.3. Selanjutnya paper ini akan membahas lebih detail perancangan sistem dari *audio scrambler* menggunakan teknik OFDM dan uji performa pada kanal transmisi.

2 Perancangan Sistem dan Diskusi



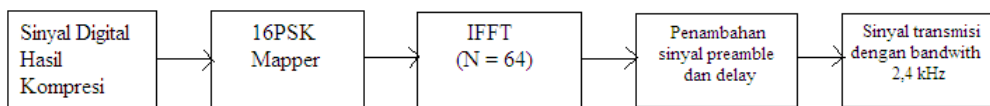
Gambar 1 Diagram blok Universal Audio Scrambler

Desain Audio Scrambler terdiri atas tiga bagian utama yaitu *vocoder*, *scrambler* dan modem OFDM. *Vocoder* berfungsi untuk proses kompresi sinyal audio (*audio compress*), selanjutnya diacak oleh *scrambler* untuk kemudian di modulasi oleh modem OFDM.

Sinyal masukan berupa sinyal suara yang disampling dengan kecepatan 8000 sampel/detik dan ukuran sampel 8 bit, sehingga jumlah bit masukan 64000 bps. Agar jumlah bit masukan yang besar ini dapat dikirim maka digunakan kompresi audio CVSD yang mampu melakukan kompresi data hingga 1/8 kali lipat. Dengan demikian, jumlah bit masukan yang diproses di modem OFDM sebesar 8000 bps.

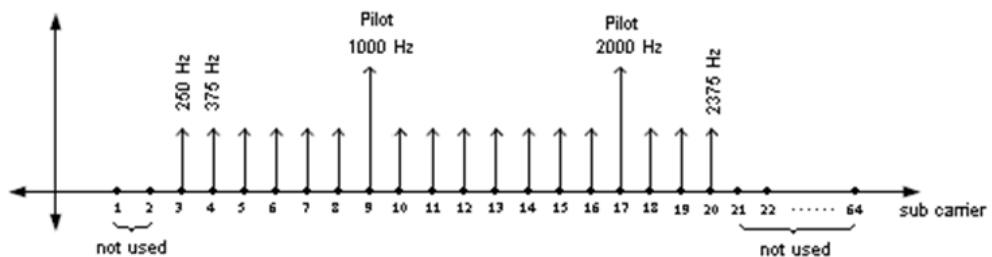
2.1 Transmitter Modem OFDM

Sinyal digital yang masuk pada OFDM modulator akan diubah ke bilangan biner dan dikelompokkan kedalam 64 bit untuk setiap frame data. Data sebanyak 64 bit ini akan mengalami proses mapping menggunakan 16PSK dimana data dibagi kedalam gugus 4 bit sehingga akan terbentuk 16 simbol untuk setiap frame data yang masuk.



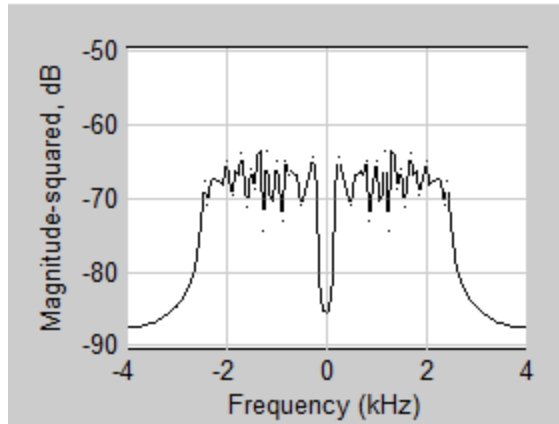
Gambar 2 Transmitter modem OFDM

Output riil dan imajiner dari 16PSK akan dikenakan jumlahnya menjadi 64 melalui penambahan 2 pilot pada data ke-9 dan ke-17 dan zero padding pada data ke-21 hingga data ke-64. Dengan demikian, panjang data akan sesuai dengan panjang IFFT yang digunakan untuk diubah domainnya dari frekuensi ke domain waktu sehingga sinyal dapat ditransmisikan.



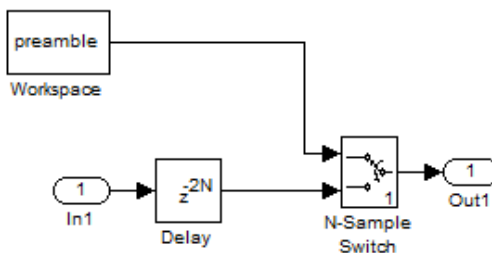
Gambar 3 Pengaturan sub carrier

Dari 64 kaki masukan IFFT, hanya 18 kaki masukan yang digunakan yaitu 16 kaki untuk 16 simbol data dan 2 kaki untuk pilot sebagai acuan untuk carrier recovery. Hal ini sengaja dilakukan agar bandwidth yang ditransmisikan tidak melebihi 2,4 kHz (jika semua kaki IFFT digunakan maka lebar bandwidth transmisi menjadi 8 KHz, melebihi kanal transmisi untuk aplikasi audio).



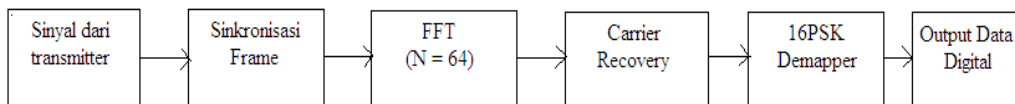
Gambar 4 Bandwidth transmisi 2.4 kHz

Sebelum ditransmisikan, di depan data akan ditambahkan sinyal *preamble* dan *delay* sebanyak 2 *frame* sebagai data sunyi. Hal ini dibutuhkan oleh *receiver* untuk mendeteksi data yang dikirimkan oleh transmitter. Dalam desain ini, *Preamble* yang digunakan berupa sinyal sinusoidal 500 Hz sepanjang 1 *frame* (64 sampel data).



Gambar 5 Penambahan *preamble* dan sinyal sunyi

2.2 Receiver Modem OFDM

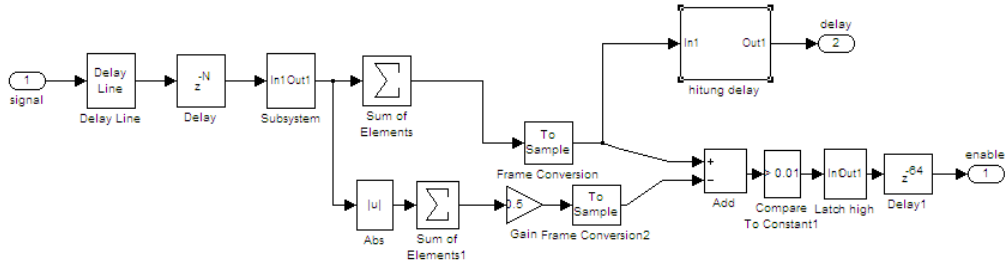


Gambar 6 Receiver OFDM

Proses yang terjadi pada *receiver* adalah kebalikan proses pada *transmitter*. Bedanya, terdapat proses *sinkronisasi frame* untuk mendeteksi *preamble* dan menghitung besar *delay* serta proses *carrier recovery* untuk memulihkan *frekuensi offset* pada data yang diterima.

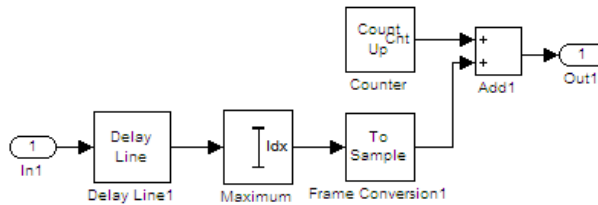
2.3 Sinkronisasi Frame

Proses sinkronisasi frame terdiri dari dua bagian yaitu proses pendeteksi *preamble* dan proses penghitung *delay*.



Gambar 7 Sinkronisasi frame

Preamble dideteksi dengan cara mengurangi jumlah total *preamble* dengan setengah jumlah total absolutnya. Jika hasilnya lebih besar dari 0.001 maka nilai logika 1 akan dikeluarkan untuk mengaktifkan (*enable*) proses penghitung *delay*.



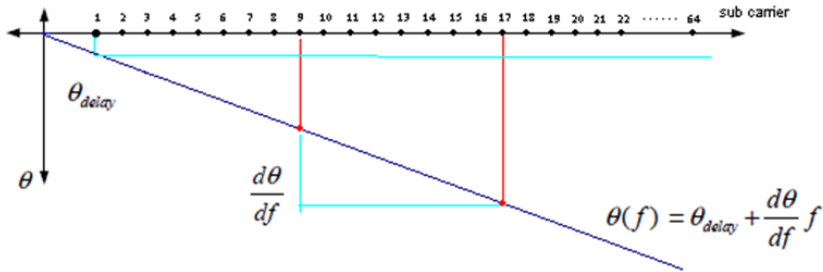
Gambar 8 Penghitung delay

Saat penghitung *delay* aktif, *counter* mulai menghitung sebanyak 64 sampel secara berulang. Sementara posisi maksimum dari jumlah total sinyal *preamble* dideteksi sejak sinyal *enable* aktif. Hasil perhitungan keduanya dijumlahkan sebagai besarnya *delay* sinyal.

2.4 Carrier Recovery

Carrier recovery digunakan untuk mengatasi adanya pergeseran frekuensi *carrier* akibat efek *doppler* maupun adanya perbedaan *clock* antara ADC dan DAC pada *transmitter* dan *receiver*. Besarnya pergeseran frekuensi antar *subcarrier* bertambah secara linier.

Penambahan pilot digunakan sebagai referensi untuk menghitung besarnya pergeseran frekuensi tiap *subcarrier*. Perhitungan pergeseran frekuensi dinyatakan dalam *phase* seperti yang ditunjukkan pada Gambar 9.

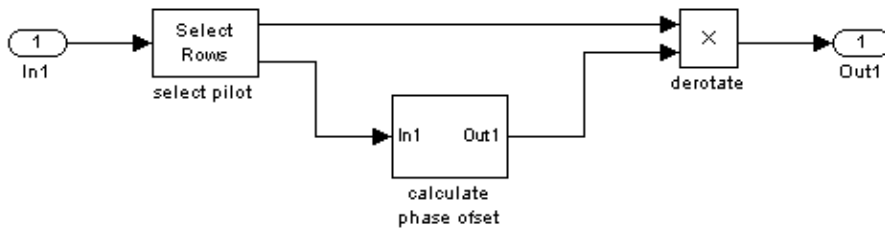


Gambar 9 Pergeseran phase

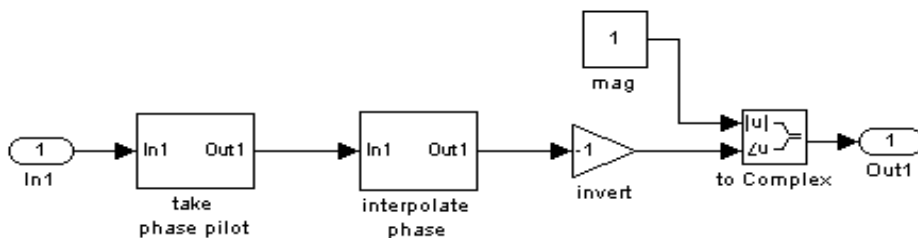
Phase dari pilot pada transmitter telah diset bernilai nol. Setelah dikirim, phase pilot akan berubah akibat adanya kanal transmisi. Perubahan phase inilah yang digunakan untuk mengestimasi pergeseran phase dari tiap subcarrier. Karena pergeseran phase berubah secara linier maka dibutuhkan minimal 2 pilot untuk menginterpolasi titik yang lain (Gambar 12). Hubungan pergeseran phase terhadap frekuensi subcarrier dapat dinyatakan dengan:

$$\theta(f) = \theta_{delay} + \frac{d\theta}{df} f \tag{1}$$

Algoritma yang digunakan untuk menghilangkan efek pergeseran phase adalah dengan mengalikan (merotasikan) tiap subcarrier tersebut dengan pergeseran phase hasil perhitungan.

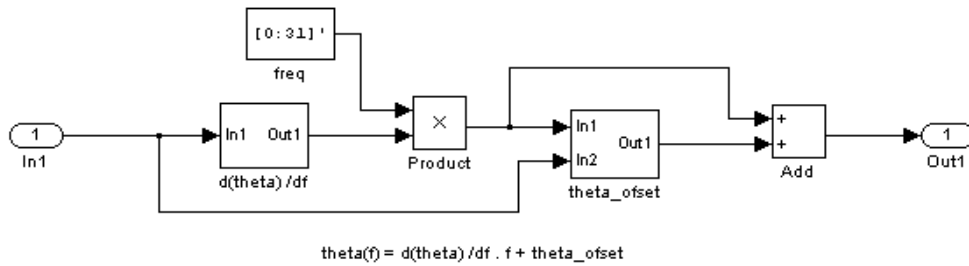


Gambar 10 Carrier recovery



Gambar 11 Proses perhitungan fase offset

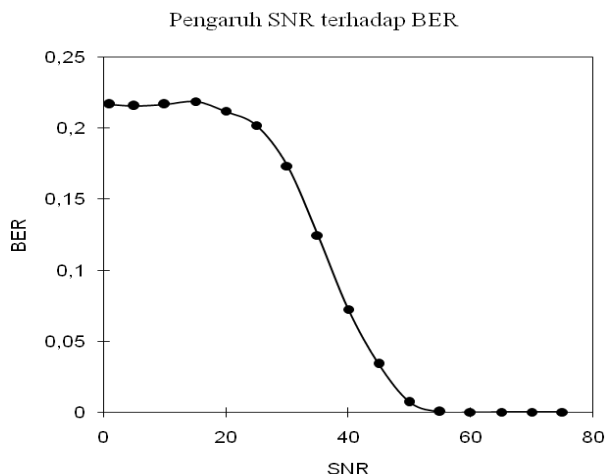
Perhitungan pergeseran *phase* dilakukan dengan mengambil *phase* dari pilot untuk interpolasi tiap *subcarrier*. Selanjutnya pergeseran *phase* dikonversi ke frekuensi menggunakan blok “to Complex” dengan besar amplitudo sama dengan 1. Blok “invert” dimaksudkan untuk membalik *phase* karena akan digunakan untuk meniadakan pergeseran *phase* dari tiap *subcarrier*.



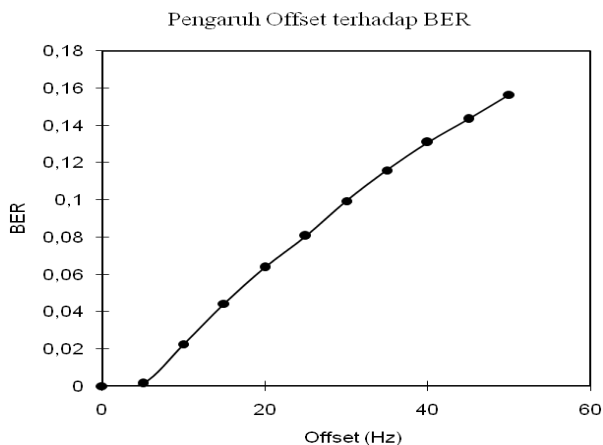
Gambar 12 Proses interpolasi *phase*

2.5 Uji Performa Pada Kanal Tansmisi

Perancangan audio *scrambler* telah diuji terhadap *noise* dan frekuensi *offset* dengan menghitung jumlah *Bit Error Ratio (BER)*. Gambar 13 (a) adalah hasil simulasi untuk menguji performa alat terhadap variasi *noise*. Dari grafik dapat disimpulkan pengaruh *noise* akan hilang pada kondisi SNR lebih besar dari 60. Pengaruh *offset* ditunjukkan pada Gambar 13 (b) dan diperoleh kesimpulan bahwa semakin besar frekuensi *offset* maka errornya akan meningkat.



(a)



(b)

Gambar 13 (a) Pengaruh SNR terhadap BER (b) Pengaruh Offset terhadap BER

3 Kesimpulan

Dari percobaan yang dilakukan diperoleh kesimpulan bahwa teknik OFDM dapat digunakan untuk memodulasi secara digital sinyal suara pada jangkauan frekuensi audio dengan bandwidth 2,4 kHz dan *data rate* 8kbps. Hasil pengujian desain audio *scrambler* dengan simulasi menunjukkan performa yang baik pada kanal dengan SNR diatas 60.

4 Daftar Pustaka

- [1] Lukman, Arif., Gojali, Elli A., *Simulasi Algoritma Dasar Pengacakan Data Audio Menggunakan Matlab*, Jurnal Teknologi Informasi Volume 5 No.1, PPI LIPI, ISSN 14411-3740, Desember 2004.
- [2] Ramdhani, B. *Desain dan Implementasi OFDM pada FPGA*. Universitas Lampung.
- [3] Vercimak, L. & Weyeneth, K., *Software Defined Radio*. Bradley University, 2006.
- [4] Texas Instruments/Spectrum Digital, *TMS320C6747 Technical Reference*, 511345-0001 Rev. A, November 2008.