



UNIVERSITAS INDONESIA

**PEMODELAN DAN SIMULASI RANGKAIAN
ENKODER TEKNIK KOMPRESI SUARA VSELP**

SKRIPSI

LUKAS WOWOR

0405030494

**SKRIPSI INI DIAJUKAN UNTUK MELENGKAPI SEBAGIAN
PERSYARATAN MENJADI SARJANA TEKNIK**

**DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS INDONESIA
DEPOK
JUNI 2009**

PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Lukas Wowor
NPM : 0405030494
Tanda Tangan :
Tanggal : 6 Juli 2009



HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : Lukas Wowor

NPM : 0405030494

Program Studi : Teknik Elektro

Judul Skripsi : Pemodelan dan Simulasi Rangkaian Enkoder Teknik Kompresi
Suara VSELP

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Dr. Ir. Arman Djohan D, M.Eng ()

Penguji : Dr. Ir. Abdul Muis , MSc. ()

Penguji : Ir. Purnomo Sidhi Priambodo ,MSc.,Phd. ()

Ditetapkan di : Depok

Tanggal : 6 Juli 2009

UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, sehingga saya dapat menyelesaikan seminar ini. Penulisan seminar ini dilakukan dalam rangka memenuhi salah satu syarat mata kuliah untuk menyelesaikan program studi di Departemen Teknik Elektro Universitas Indonesia.

Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan seminar ini, sangatlah sulit bagi saya untuk menyelesaikan seminar ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

- (1) Dr. Ir. Arman Djohan Diponegoro selaku dosen pembimbing kedua saya yang juga telah meluangkan waktu untuk memberikan saran dan bimbingan dalam penyelesaian seminar ini;
- (2) orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral; dan
- (3) sahabat dan teman – teman yang telah banyak membantu saya dalam menyelesaikan seminar ini.

Akhir kata, saya berharap seminar ini dapat membawa manfaat bagi pengembangan ilmu selanjutnya.

Depok, 6 Juli 2009

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Lukas Wowor
NPM : 0405030494
Program Studi : Teknik Elektro
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis karya : Skripsi

demikian demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

Pemodelan dan Simulasi Rangkaian Enkoder Teknik Kompresi Suara VSELP

berserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 6 Juli 2009

Yang menyatakan

(Lukas Wowor)

ABSTRAK

Nama : Lukas Wowor
Program Studi : Teknik Elektro
Judul : Pemodelan dan Simulasi Rangkaian Enkoder Pada Teknik Kompresi Suara VSELP

Skripsi ini membahas mengenai pemodelan dan simulasi rangkaian enkoder pada teknik pengkodean suara VSELP. VSELP merupakan suatu teknik pengkodean suara atau kompresi pada komunikasi selular yang memberikan tingkat kompresi sebesar delapan kilo bit perdetik. Bagian Enkoder VSELP melakukan kompresi dengan metode *Linear Prediction Coding* dan memodelkan nilai eksitasi dengan kode buku adaptif dan kode buku eksitasi. Bagian Enkoder ini akan disimulasikan dan dibuktikan tingkat kompresi suaranya. Simulasi dilakukan dengan secara offline yang berarti suara direkam terlebih dahulu dan kemudian diproses secara terpisah. Simulasi tersebut akan menghasilkan sebuah bentuk dokumen dari data asli yang akan memberikan tingkat kompresi sebesar delapan kilo bit perdetik.

Kata kunci:

Enkoder, *Linear Prediction Coding*, eksitasi, VSELP

ABSTRACT

Name : Lukas Wowor
Study Program : Electrical Engineering
Title : Modeling and Simulation of Encoder Circuit of Voice
Compression Technique VSELP

This final project discusses designing and simulating encoder part of VSELP sound compression. Vector Sum Excitation Linear Prediction is one of the technique of sound compression or sound coding on cellular communication that give compression into eight kilo bit per second. VSELP Encoder compressed sound data with linear prediction coding and model the excitation with adaptive codebook and excitation codebook This Encoder part will be simulated to prove the compression level of it. Simulation done in offline condition whereas the data recorded and processed separately. Simulation will give a new file from the origin data that give the eight kilo bit persecond compression.

Key Note:

Encoder, Linear Prediction Coding, Excitation.

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN ORISINALITAS	ii
PENGESAHAN	iii
HALAMAN PERNYATAAN PUBLIKASI	iv
UCAPAN TERIMA KASIH	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	ix
DAFTAR TABEL	x
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan Penulisan	2
1.3 Pembatasan Masalah	2
1.4 Metode Penelitian.....	2
1.5 Sistematika Penelitian	2
BAB 2 PRINSIP DASAR PENGKODEAN VSELP	4
2.1 Pemodelan Suara Manusia.....	4
2.2 Linear Prediction Coding.....	7
2.2.1 Dasar Linear Prediction Coding	7
2.2.2 Metode Pencarian Koefisien All-Pole Filter	9
2.3 Prediksi Lag Fundamental Frekuensi	11
2.4 Eksitasi Suara.....	12
2.5 Dasar Pengkodean Enkoder VSELP	13
2.5.1 Pengkodean Enkoder VSELP.....	13
2.5.2 Struktur dan Metode Pengkodean Enkoder VSELP	14
2.6 Tabel Kode Adaptif dan Tabel Kode Eksitasi	17
BAB 3 ALGORITMA RANGKAIAN ENKODER VSELP	18
3.1 Analisa Jangka Pendek	19
3.1.1 Koefisien Refleksi dan Energi Bingkai.....	19

3.1.2 Interpolasi Parameter Analisa Jangka Pendek.....	20
3.2 Analisa Jangka Panjang	22
3.2.1 Filter Pembobotan Signal Suara	22
3.2.2 Pencarian Lag atau Periode Frekuensi Fundamental.....	23
3.2.3 Filter State	24
3.2.4 Pencarian Parameter Kode Buku Eksitasi.....	24
3.2.5 Transformasi Penguatan ke Dalam GSP0P1	27
3.2.6 Pembaharuan Nilai Kode Buku Adaptif	32
3.3 Proses Penyatuan Parameter Bingkai	30
BAB 4 SIMULASI & HASIL PROSES RANGKAIAN ENKODERVSELP	32
4.1 Simulasi Rangkaian Enkoder.....	32
4.2 Hasil Simulasi	33
4.3 Waktu Pemrosesan Simulasi Rangkaian Enkoder VSELP.....	34
4.4 Analisa Proses Rangkaian Enkoder.....	35
BAB 5 KESIMPULAN	37
DAFTAR REFERENSI	38
LAMPIRAN	39

DAFTAR GAMBAR

Gambar 2.1 Proses Produksi Suara Manusia	5
Gambar 2.2 Pemodelan Produksi Suara Manusia	6
Gambar 2.3 Proses Pemodelan Linear Suara Manusia	7
Gambar 2.4 Hasil Pemodelan Linear Suara Manusia	9
Gambar 2.5 Metode Algoritma FLAT	10
Gambar 2.6 Lag atau Fundamental Frekuensi	12
Gambar 2.7 Proses LPC	13
Gambar 2.8 Blok Diagram Enkoder VSELP	15
Gambar 3.1 Proses Enkoder VSELP	18
Gambar 3.2 Proses Analisa Jangka Pendek	19
Gambar 3.3 Proses Analisa Jangka Panjang	22
Gambar 3.4 Proses Pemilihan Indeks Kode Buku	25
Gambar 3.5 Proses Pembaharuan Kode Buku Adaptif	30
Gambar 3.6 Struktur Data Kompresi VSELP	30
Gambar 4.1 Sampel Suara 'Saya' dengan basis waktu	32
Gambar 4.2 Hasil Suara Sintesa Sederhana	34
Gambar 4.3 Sintesa suara dengan Levinson Durbin	36

DAFTAR TABEL

Tabel 2.1 Struktur Data Kompresi VSELP 16

Tabel 2.2 Susunan Indeks VSELP 16



BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan kemajuan dalam teknologi telekomunikasi selular membuat komunikasi menjadi lebih mudah untuk dilakukan. Pengguna teknologi selular selalu bertambah dengan kelipatan logaritmik. Pertumbuhan tersebut harus diimbangi dengan kemampuan operator atau penyedia layanan selular untuk memberikan kualitas layanan yang terbaik. Pengoptimalan jaringan dan kapasitas jaringan harus dapat melayani kebutuhan pelanggan dalam berkomunikasi.[1],[2],[3]

Frekuensi adalah sumber daya yang terbatas karena penggunaanya yang dibatasi oleh aturan pemerintah. Alokasi lebar pita frekuensi yang diperoleh oleh setiap operator terbatas sehingga permasalahan akan terjadi. Operator akan kesulitan untuk mencukupi kebutuhan komunikasi pelanggan yang bertumbuh dengan tetap.

Penggunaan pita frekuensi oleh operator sebagian besar dialokasikan untuk kebutuhan komunikasi suara. Kualitas suara yang baik berada pada 64kbps sampai 256 kbps. Jumlah pita lebar yang dipergunakan akan sangat sedikit untuk jumlah pengguna pada waktu yang bersamaan. Untuk mengatasi permasalahan tersebut, kompresi harus dilakukan pada data suara untuk memberikan kapasitas kanal yang lebih besar.

Teknik pengkodean suara VSELP (Vector Sum Excitation Linear Prediction) merupakan salah satu metode kompresi suara pada jaringan selular. VSELP merupakan pengembangan teknologi kompresi suara yang dipergunakan pada IS-54 di Amerika Utara. VSELP melakukan kompresi data dari 128kbps menjadi 8kbps sehingga kapasitas kanal dapat dipergunakan 16 kali lebih banyak.[4][5][6]

1.2 Tujuan Penulisan

Skripsi ini bertujuan untuk melakukan perancangan rangkaian enkoder dari pengkodean VSELP dan membahas setiap proses yang dilakukan rangkaian enkoder serta mensimulasikannya.

1.3 Batasan Masalah

Batasan masalah pada skripsi ini adalah menjelaskan dasar teknik pengkodean VSELP dan proses rangkaian enkoder serta mensimulasikan tingkat kompresi VSELP dengan tidak memperhatikan kualitas suara sintesa.

1.4 Metodologi Penelitian

Metodologi penelitian yang dipergunakan yaitu:

1. Studi literatur melalui buku, jurnal, dan tesis yang membahas mengenai *speech coding* terutama VSELP (*Vector Sum Excited Linear Prediction*).
2. Melakukan penelitian perancangan encoder VSELP dari algoritma VSELP 8kbps.
3. Melakukan simulasi encoder VSELP pada SIMULINK MATLAB

1.5 Sistematika Penulisan

BAB I

PENDAHULUAN

Menjelaskan tentang latar belakang dan dasar pemikiran pengkodean suara dengan laju bit rendah, tujuan penulisan, batasan masalah dalam pembahasan tugas seminar ini, beserta metodologi penelitian yang digunakan dan sistematika penulisan makalah seminar ini.

BAB II

PRINSIP DASAR PENGKODEAN VECTOR SUM EXCITATION LINEAR PREDICTION

Menjelaskan tentang dasar teori dan teknik – teknik dasar dalam pengkodean suara umumnya dan juga proses pembentukan suara manusia.

BAB III

ALGORITMA RANGKAIAN ENKODER VSELP

Menjelaskan tentang keseluruhan algoritma proses yang dilakukan oleh rangkaian enkoder VSELP.

BAB IV

SIMULASI DAN HASIL PROSES RANGKAIAN ENKODER VSELP

Menjelaskan hasil dari proses simulasi rangkaian enkoder VSELP serta analisa terhadapnya.

BAB IV

KESIMPULAN



BAB II

PRINSIP DASAR PENGKODEAN VECTOR SUM EXCITATION LINEAR PREDICTION

Vector Sum Excitation Linear Prediction (VSELP) adalah teknik pengkodean suara analisa-sintesis (*Hybrid Coding*) pada telekomunikasi selular yang mengkompresi data suara menjadi delapan kilo bit perdetik. VSELP merupakan standar IS-54 yaitu standar kompresi atau pengkodean suara di Amerika Utara. Teknik pengkodean tersebut merupakan teknik pengkodean analisis-sintesis suara yang menganalisa data suara dan kemudian menghasilkan suara sintesis dengan metode *linear prediction* dan dengan tiga tabel kode untuk suara.

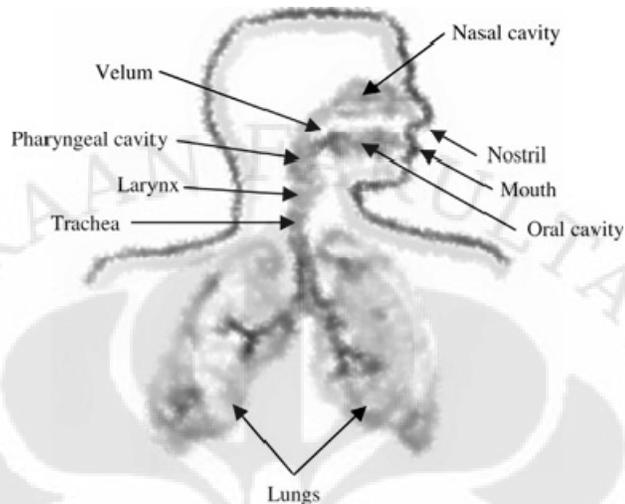
Teknik kompresi dan pengkodean suara didasarkan kepada dasar pemodelan bentuk dari pembentukan suara oleh pita suara manusia. Pemodelan produksi suara manusia dilakukan secara linear. Pemodelan tersebut adalah linear prediction yang merupakan dasar dari setiap kompresi suara. VSELP berdasar kepada LPC sebagai filter dari masukan yang merupakan kombinasi dari tiga kode tabel yang merepresentasikan data suara masukan.

2.1 Pemodelan Suara Manusia

Bentuk gelombang suara adalah sebuah tekanan gelombang suara yang berasal dari pergerakan struktur anatomi yang dibuat sistem produksi suara manusia. Suara (*speech*) secara dasar dihasilkan sebagai gelombang akustik yang dipancarkan dari lubang hidung (*nasal*) dan mulut ketika udara dikeluarkan dari paru – paru dengan hasil aliran udara mendesak karena adanya desakan di dalam tubuh. Hal ini berguna untuk menginterpretasikan produksi suara dalam hal penyaringan akustik.

Tiga rongga utama dari sistem produksi suara yaitu nasal, oral dan pharyngeal yang membentuk filter akustik utama. Filter dieksitasi oleh udara dari paru – paru dan diisi pada keluaran utama filter tersebut dengan tahanan yang berkaitan dengan bibir. Filter tersebut yang kita modelkan yang akan

dipergunakan untuk analisa dan sintesis suara. Filter tersebut disebut prediksi linear atau *linear prediction* seperti pada Gambar 2.1.



Gambar 2.1 Proses Produksi Suara Manusia [1]

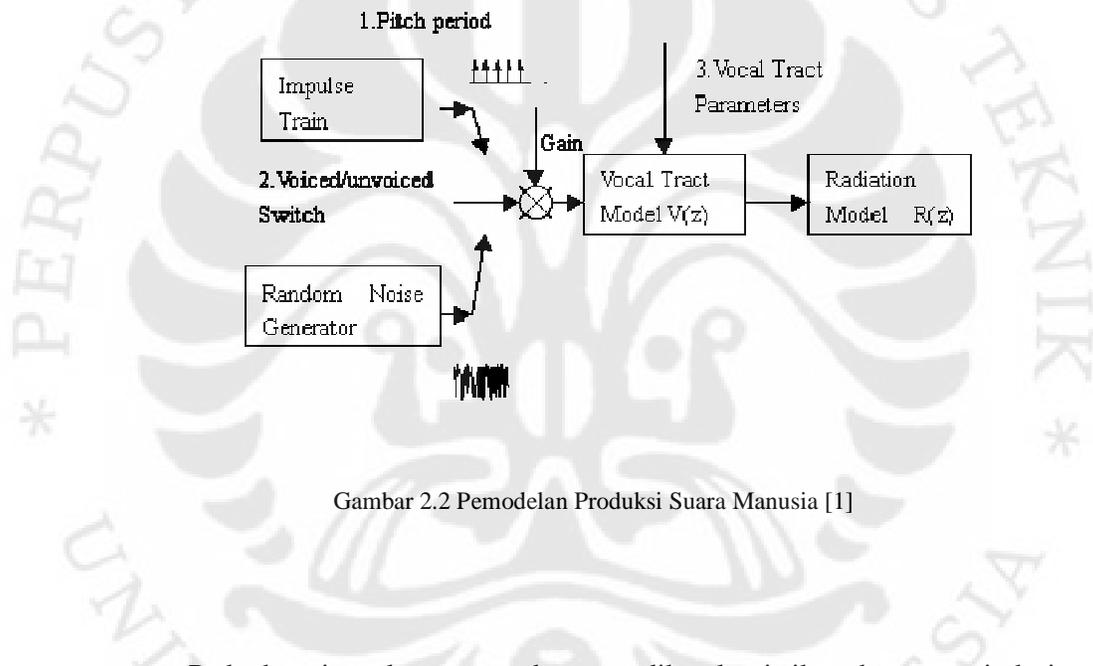
Rongga suara (*vocal tract*) merujuk pada *pharyngeal* dan rongga mulut (*oral cavity*) yang dikelompokkan bersama – sama. *Nasal tract* dimulai dari bagian *velum* dan berakhir pada lubang hidung (*nostrils*). Ketika *velum* merendah, *nasal tract* berpasangan secara akustik dengan rongga suara untuk menghasilkan suara nasal.[8]

Wujud dan bentuk dari *vocal* dan *nasal tract* berubah kontinyu terhadap waktu, menciptakan sebuah filter akustik dengan respon frekuensi terhadap variasi waktu. Saat udara dari paru – paru berjalan melalui *tracts*, spektrum frekuensi dibentuk oleh selektivitas frekuensi dari *tract – tract* ini. Frekuensi resonansi dari pipa rongga suara disebut frekuensi formant (*formant frequency*) atau formant, yang tergantung pada bentuk dan dimensi dari rongga suara.[8]

Di dalam laring (*larynx*), terdapat salah satu komponen paling penting dari sistem produksi suara yaitu pita suara (*vocal cords*). Pita suara adalah sepasang pita elastis otot dan membran muskos yang membuka dan menutup dengan cepat

selama produksi suara. Kecepatan pita dalam membuka dan menutup unik untuk masing – masing individu dengan suara tertentu.

Saat berbicara, sinyal suara dapat diklasifikasikan menjadi ucapan bersuara (*voiced*) dan ucapan tidak bersuara (*unvoiced*) seperti pada Gambar 2.2. Ucapan bersuara dihasilkan ketika pita suara bergetar sehingga aliran udara dari paru – paru terinterupsi secara periodik, menciptakan deretan pulsa untuk mengeksitasi *vocal tract*. Dengan pita suara yang tetap (tidak bergetar), terjadi turbulensi yang diciptakan dengan adanya aliran udara yang melewati rongga suara yang terdesak sehingga menghasilkan ucapan tak bersuara.



Gambar 2.2 Pemodelan Produksi Suara Manusia [1]

Pada domain waktu, ucapan bersuara dikarakteristikan dengan periodesitas yang kuat yang muncul di dalam sinyal, dengan frekuensi dasar yang disebut sebagai frekuensi puncak (*pitch frekuensi*) atau *pitch*. Untuk laki – laki, kisaran *pitch* berada pada 50 – 250 Hz, sedangkan untuk wanita kisarannya pada interval 120 – 500 Hz. Ucapan tak bersuara, sebaliknya tidak menunjukkan adanya periodesitas dan pada dasarnya acak secara alami. [1][2][8]

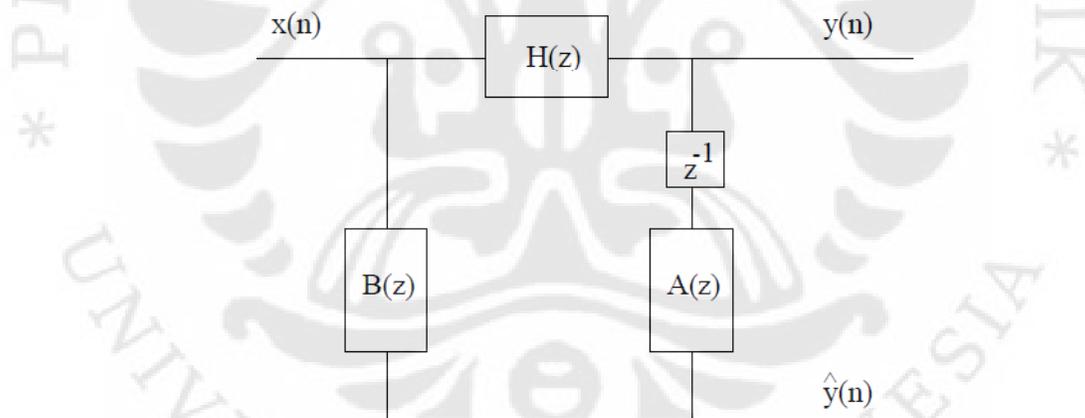
2.2 Linear Prediction Coding

2.2.1 Dasar Linear Prediction Coding

Linear Prediction Coding (LPC) merupakan bagian yang penting dalam pemrosesan signal suara. LPC memodelkan trek vocal dengan sebuah all-pole filter yang meprediksi suara dengan kombinasi dari input dan keluaran sebelumnya secara rekursif.

$$\hat{y} = \sum_{k=1}^p a(k)y(n-p) + \sum_{k=0}^N b(k)x(n-N) \quad (1)$$

Nilai keluaran dari prediksi \hat{y} merupakan nilai prediksi atau estimasi terhadap keluaran sebenarnya seperti pada Gambar 2.3.[8] Idenya adalah dengan mengetahui nilai masukan $x(n)$ dan keluaran $y(n)$ kita dapat mengetahui sifat dan karakteristik dari system kita $H(z)$. Permasalahannya adalah bagaimana kita dapat mengetahui koefisien $a(k)$ dan $b(k)$ sehingga kita dapat memperoleh keluaran estimasi yang baik dan dengan kesalahan estimasi yang kecil terhadap nilai dari keluaran yang sebenarnya.



Gambar 2.3 Proses Pemodelan Linear Suara Manusia [1]

Untuk mendapatkan koefisien dari filter estimasi suara manusia dipergunakan metode auto regressive dengan alasan :

- a. Masukan dari sistem tidak diketahui.

- b. Komputasi yang mudah untuk menentukan $a(k)$.
- c. Vokal trek dari manusia secara teori merupakan filter all-pole.
- d. *All pole* dari model yang stabil dapat merepresetasikan respon amplitudo dari setiap gelombang suara.

Sistem all pole dengan transfer fungsi :

$$\frac{g}{A(z)} \quad (2.1)$$

dengan $A(z) = 1 + a_1z^{-1} + \dots + a_nz^{-p}$ dan g merupakan gain.

$$\frac{Y(z)}{X(z)} = \frac{g}{A(z)},$$

$$Y(z)A(z) = gX(z) \quad (2.2)$$

Inverse dari persamaan dua adalah :

$$y(n) + \sum_{i=1}^p a_i y(n-i) = gx(n),$$

Sehingga

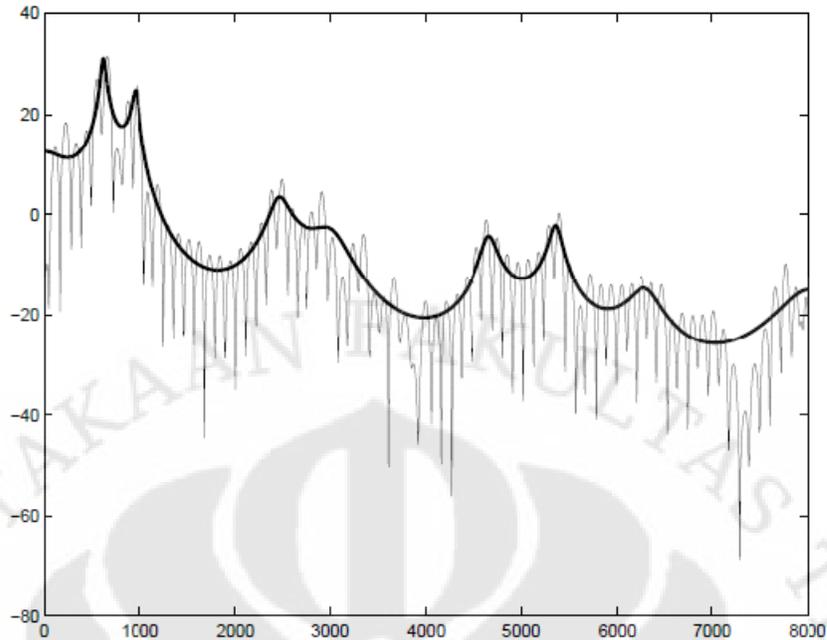
$$y(n) = gx(n) - \sum_{i=1}^p a_i y(n-i) \quad (2.3)$$

Pada kenyataannya, masukan dari sistem tidak pernah diketahui dan kita memfokuskan diri pada all-pole filter. Sehingga pemodelan hanya bergantung kepada keluaran sebelumnya dan nilai estimasi tersebut dapat memodelkan dengan baik. Sehingga persamaan nilai estimasi dari signal suara adalah

$$\hat{y} = - \sum_{k=0}^p a(k)y(n-p) \quad (2.4)$$

Orde dari pemodelan all pole filter dari sistem kita adalah sepuluh pada frekuensi cuplik sebesar 8000 Hz. Orde sepuluh dari sistem kita akan dapat memberikan model yang baik untuk mengestimasi suara manusia.

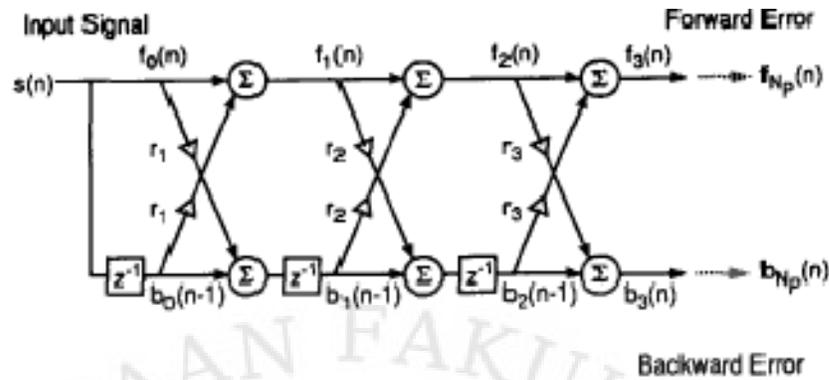
LPC dengan sepuluh koefisien adalah standar yang dipergunakan oleh Departemen Pertahanan Amerika untuk dipergunakan dalam kompresi suara bit rendah dengan standar FS1015. LPC 10 adalah standar bagi algoritma kompresi yang lain untuk mempergunakan sepuluh koefisien untuk analisa jangka pendek dan dapat memodelkan sinyal suara seperti pada Gambar 2.4.



Gambar 2.4 Hasil Pemodelan Linear Suara Manusia [1]

2.2.2 Metode Pencarian Koefisien All-Pole Filter

Pencarian koefisien dari filter all pole $A(z)$ dapat dilakukan dengan berbagai metode yaitu Levinson Durbin dan Leorux-Gueguen serta *Fixed Point Covariance Latice Algorithm* (FLAT). Pada Pengkodean VSELP, algoritma FLAT dipergunakan untuk mencari koefisien filter dari LPC. Struktur FLAT seperti pada Gambar 2.5 mencari koefisien refleksi yang kemudian akan mencari nilai koefisien filter(α) dengan konversi. Hal ini dilakukan karena koefisien refleksi mempunyai rentang yang tidak lebar sehingga ketika koefisien refleksi dikuantisasi, kesalahan kuantisasi akan semakin kecil. Koefisien refleksi juga lebih stabil karena rata-rata tidak melebihi satu nilai besarannya (syarat kestabilan filter diskrit, *pole* lebih kecil sama dengan satu).



Gambar 2.5 Metode Algoritma FLAT [10]

Pencarian koefisien LPC dengan metode FLAT adalah mencari koefisien refleksi dengan masukan signal suara dan dengan keluaran *forward prediction error* atau prediksi error.[5][13] Definisikan $b_j(n)$ sebagai kesalahan mundur dari filter dan $f_j(n)$ sebagai kesalahan maju dari filter.

Autokorelasi dari $f_j(n)$:

$$F_j(i, k) = \sum_{n=np}^{na-1} f_j(n-i) f_j(n-k) \quad (2.5)$$

Autokorelasi dari $b_j(n-1)$:

$$B_j(i, k) = \sum_{n=np}^{na-1} b_j(n-i-1) b_j(n-k-1) \quad (2.6)$$

Autokorelasi dari (5) dan (6) :

$$C_j(i, k) = \sum_{n=np}^{na-1} f_j(n-i) b_j(n-k-1) \quad (2.7)$$

pembaharuan pada autokorelasi pada kesalahan maju adalah

$$F_j(i, k) = F_{j-1}(i, k) + r_j (C_{j-1}(i, k) + C_{j-1}(k, i)) + r_j^2 B_{j-1}(i, k) \quad (2.8)$$

pembaharuan pada autokorelasi pada kesalahan mundur adalah

$$B_j(i, k) = B_{j-1}(i+1, k+1) + r_j (C_{j-1}(i+1, k+1) + C_{j-1}(k+1, i+1)) + r_j^2 F_{j-1}(i+1, k+1) \quad (2.9)$$

pembaharuan nilai autokorelasi kesalahan maju dan mundur adalah

$$C_j(i, k) = C_{j-1}(i, k+1) + r_j (B_{j-1}(i, k+1) + F_{j-1}(i, k+1)) + r_j^2 C_{j-1}(k+1, i) \quad (2.10)$$

lalu r_j adalah koefisien refleksi untuk j tingkat filter maka nilai dari kesalahan maju dan kesalahan mundur adalah

$$r_j = -2 \frac{C_{j-1}(0,0) + C_{j-1}(Np-j, Np-j)}{F_{j-1}(0,0) + B_{j-1}(0,0) + F_{j-1}(Np-j, Np-j) + B_{j-1}(Np-j, Np-j)} \quad (2.11)$$

Algoritma FLAT adalah sebagai berikut:

1. Hitung kovariansi dari signal suara

$$\varphi(i, k) = \sum_{n=np}^{na-1} s(n-i)s(n-k) \quad (2.12)$$

Untuk $0 \leq i, k \leq Np$

2. Kemudian hitung F_0, C_0, B_0

$$F_0(i, k) = \varphi(i, k) \quad (2.13)$$

$$B_0(i, k) = \varphi(i+1, k+1) \quad (2.14)$$

$$C_0(i, k) = \varphi(i, k+1) \quad (2.15)$$

Untuk $0 \leq i, k \leq Np$

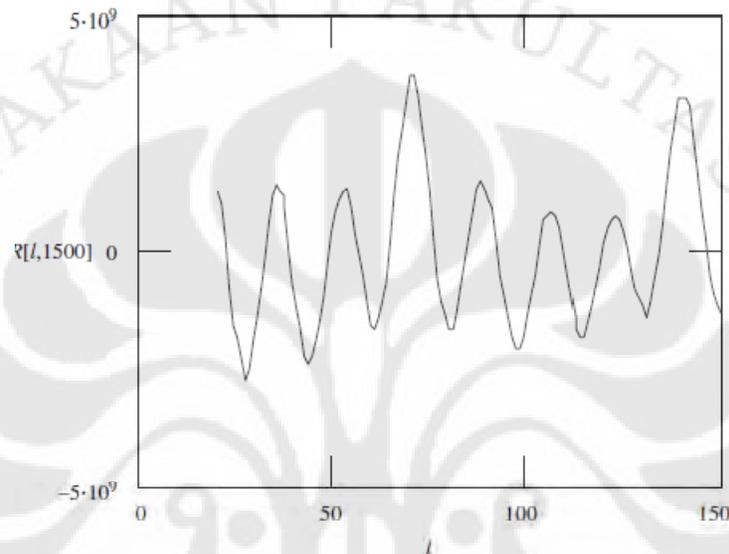
3. Kemudian hitung r_j kemudian perbaharui data F_j, B_j, C_j untuk $0 \leq i, k \leq Np-j-1$
4. Lakukan melakukan looping proses 3 sebanyak 10 kali

2.3 Prediksi Lag Fundamental Frekuensi

Salah satu parameter yang penting dalam analisa, sintesis, dan pengkodean suara adalah estimasi fundamental frekuensi atau pitch dari bunyi suara. Fundamental frekuensi berkenaan langsung dengan pembicara dan memberikan warna suara bagi pembicara. Suara tercipta ketika udara mengalir dari paru-paru diinterupsi oleh getaran pita suara secara periodik. Periode dimana pita suara terbuka adalah fundamental frekuensi atau pitch. Lag adalah rentang waktu dimana fundamental frekuensi berulang.

Pria memiliki fundamental frekuensi yang berada pada rentang 50 sampai 250 Hz sedangkan wanita berada pada rentang 120 dan 500 Hz [2]. Pada domain waktu, pria berada pada rentang 4 sampai 20 ms sedangkan wanita berada pada rentang waktu 2 sampai 8 ms. Fundamental frekuensi harus dianalisa per bingkai dengan bingkai sebelumnya dan dapat dilakukan hanya untuk bunyi suara saja karena karakteristik suara yang periodik sedangkan bunyi tidak bersuara tidak. Bingkai analisa suara biasanya berkisar diantara 160-180 data suara perbingkai.

Fundamental frekuensi pada umumnya berada pada rentang 2.5 ms sampai 18.3 ms sehingga untuk analisa *pitch lag* pada bingkai suara dapat dikodekan pada rentang waktu tersebut atau pada data 20 sampai 147 [2]. Hal tersebut sudah dapat memberikan prediksi yang mendekati sempurna untuk analisa-sintesis suara. Pada Gambar 2.6, kita melihat bahwa fundamental frekuensi adalah 70-72 dan berulang lagi pada kisaran 140-145 sehingga lag adalah 70-72.



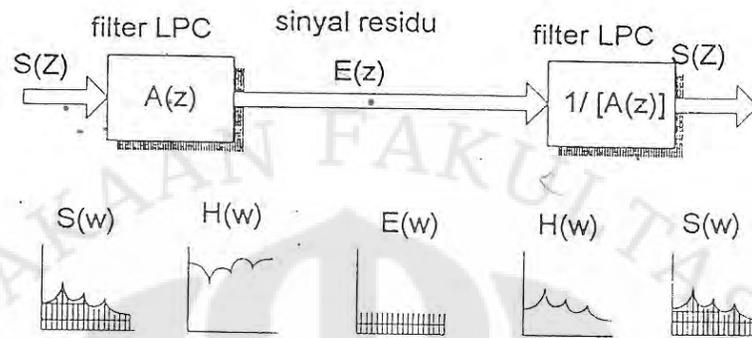
Gambar 2.6 Lag atau Fundamental Frekuensi [1]

Pada semua teknik kompresi suara, nilai lag akan memberikan warna pada suara sintesis pembicara. Lag merupakan bagian yang penting karena merupakan letak dari fundamental frekuensi suara yang akan memberikan hasil keluaran suara sintesis yang sesuai dengan suara yang diestimasi.

2.4 Eksitasi Suara

Pada pengkodean hybrid, nilai dari residu (eksitasi) dianalisa dengan cara mencocokkan data suara persubbingkai dengan kombinasi tabel kode. Tabel kode tersebut akan memberikan nilai estimasi yang cocok dengan nilai eksitasi suara yang asli. Nilai estimasi tersebut jika dimasukkan kembali kedalam filter LPC

maka akan memberikan nilai suara sintesis yang mendekati suara asli [7][8][9]. Terlihat pada Gambar 2.7, sinyal residu akan menghasilkan suara sintesis yang mendekati nilai dari sinyal suara sebenarnya.



Gambar 2.7 Proses LPC [2]

2.5 Dasar Pengkodean Enkoder VSELP

2.5.1 Pengkodean Enkoder VSELP

VSELP merupakan teknik pengkodean analisa-sintesis yang merupakan pengkodean yang kompleks akan tetapi dapat memberikan kompresi yang baik serta suara sintesis yang baik. VSELP melakukan kompresi suara hingga delapan kbps yang memberikan kapasitas dari jaringan untuk menampung data lebih banyak lagi.

Pengkodean suara VSELP didisain untuk dapat mencapai tujuan – tujuan sebagai berikut:

- lebih efisien dalam indeks tabel kode yang diinginkan
- laju bit yang rendah
- kebutuhan ruang penyimpanan tabel kode yang lebih kecil
- lebih tahan (*robust*) terhadap *channel errors*
- memiliki tingkat kualitas suara sebaik mungkin

VSELP melakukan dua analisa yaitu analisa jangka pendek dan analisa jangka panjang. VSELP melakukan pengkodean berdasarkan bingkai suara

dimana satu bingkai terdiri dari 160 data suara yang terbagi kedalam empat subbingkai suara dengan masing-masing subbingkai berisi 40 data suara. Analisa jangka pendek menganalisa satu bingkai dan melakukan pencarian koefisien LPC dan energi bingkai sedangkan analisa jangka panjang menganalisa persubbingkai dan melakukan pencarian tabel kode eksitasi dan gain serta nilai dari lag.

VSELP mempunyai tiga tabel kode yaitu satu tabel kode adaptif dan dua tabel eksitasi. Satu tabel adaptive nilainya berubah dan diperbaharui setiap analisa persubbingkai untuk melakukan analisa lag sedangkan dua tabel eksitasi adalah tetap nilainya dan dipergunakan untuk menganalisa eksitasi dari suara persubbingkai.

VSELP menganalisa suara dengan basis bingkai dengan sampel suara 160 data perbingkai. Satu bingkai memberikan parameter sebanyak 160 data perbingkai (20ms) sehingga akan memberikan kompresi sebesar 8Kbps.

2.5.2 Struktur dan Metode Pengkodean Encoder VSELP

Struktur pengkodean VSELP berdasarkan kepada tiga proses besar yaitu analisa jangka panjang, jangka pendek, serta pemrosesan data hasil kompresi. Ketiga stuktur tersebut dilakukan secara bertahap dari analisa jangka pendek, jangka panjang, dan pemrosesan data kompresi. Pada Gambar 2.8 adalah keseluruhan proses enkoder dari VSELP.

Analisa jangka pendek dilakukan pada bagian encoder untuk mencari koefisien all pole filter dengan metode FLAT. Metode tersebut seperti dijelaskan pada subbab 2.2.2 dilakukan untuk mencari sepuluh koefisien refleksi dengan N_a berjumlah 170 data dan N_p berjumlah 10. Analisa jangka pendek juga melakukan perhitungan dari energi bingkai. Setelah didapat parameter-parameter jangka pendek, parameter tersebut akan diinterpolasi untuk mendapatkan nilai parameter yang stabil untuk dipergunakan pada analisa jangka panjang.

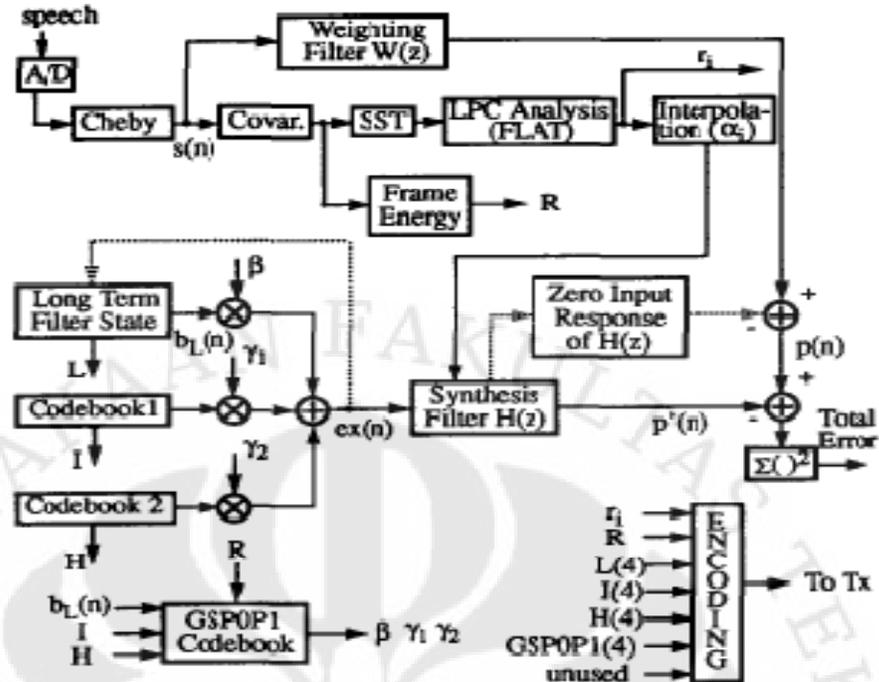


Fig. 1 VSELP encoder block diagram

Gambar 2.8 Blok Diagram Enkoder VSELP [4]

Analisa jangka panjang adalah proses untuk menganalisa nilai residual atau eksitasi secara rangkaian tertutup untuk mendapatkan nilai dari tabel kode yang cocok untuk menghasilkan kembali nilai eksitasi. Analisa dilakukan persubbingkai yang berisi 40 data untuk dianalisa. Nilai eksitasi tersebut kemudian akan masuk kedalam filter untuk menghasilkan suara sintesis yang mendekati suara aslinya. Pada analisa jangka panjang, parameter-parameter yang didapat adalah nilai indeks lag dari kode tabel adaptif, nilai indeks kedua tabel kode eksitasi, serta nilai gain dari setiap subbingkai.

Pemrosesan data hasil pengkodean adalah hal yang terakhir dilakukan untuk mendapatkan data yang tepat dan sesuai untuk didekodekan kembali pada sisi decoder. Berikut merupakan struktur pengkodean data :

Tabel 2.1 Struktur Data Kompresi VSELP [4]

Parameter	Bit/subbingkai (5ms)	Bit/bingkai (20 ms)
10 koefisien LPC(10)		38
Energi bingkai (R(0))		5
Lag L (4)	7	28
Indeks tabel kode I,H(8)	7+7	56
GSP0P1(4)	8	32
Tidak terpakai		1
TOTAL	29	160

Jumlah bit yang dipakai untuk setiap parameter merupakan nilai data indeks dari kuantisasi tabel setiap parameter. Setiap parameter dikuantisasi sesuai dengan banyaknya bit kuantisasi indeks. Untuk N bit berarti ada 2 pangkat N data kuantisasi setiap parameter. Setiap data tabel kuantisasi didapatkan dari standar IS54 yang dipakai pada standar telekomunikasi Amerika Utara.

Susunan koefisien-koefisien yang akan dikodekan seperti pada tabel 3.2 memberikan Gambaran tentang struktur parameter VSELP. Setiap bingkai per 20 ms akan memberikan 160 parameter sehingga akan memberikan 8000 parameter per detik.

Tabel 2.2 Susunan Indeks VSELP [9]

Nama Parameter	Jumlah bit	Keterangan
R0	5	Energi bingkai
LPC 1	6	Koefisien refleksi 1
LPC 2	5	Koefisien refleksi 2
LPC 3	5	Koefisien refleksi 3
LPC 4	4	Koefisien refleksi 4
LPC 5	4	Koefisien refleksi 5
LPC 6	3	Koefisien refleksi 6
LPC 7	3	Koefisien refleksi 7
LPC 8	3	Koefisien refleksi 8
LPC 9	3	Koefisien refleksi 9
LPC 10	2	Koefisien refleksi 10
Lag_1	1	Lag subbingkai 1
Lag_2	1	Lag subbingkai 2

Lag_3	1	Lag subbingkai 3
Lag_4	1	Lag subbingkai 4
CODE 1_1	7	Indeks tabel kode eksitasi 1 subframe 1
CODE 1_2	7	Indeks tabel kode eksitasi 1 subframe 2
CODE 1_3	7	Indeks tabel kode eksitasi 1 subframe 3
CODE 1_4	7	Indeks tabel kode eksitasi 1 subframe 4
CODE 2_1	7	Indeks tabel kode eksitasi 2 subframe 1
CODE 2_2	7	Indeks tabel kode eksitasi 2 subframe 2
CODE 2_3	7	Indeks tabel kode eksitasi 2 subframe 3
CODE 2_4	7	Indeks tabel kode eksitasi 2 subframe 4
GSP0P1 1	8	Indeks tabel penguatan subframe 1
GSP0P1 2	8	Indeks tabel penguatan subframe 2
GSP0P1 3	8	Indeks tabel penguatan subframe 3
GSP0P1 4	8	Indeks tabel penguatan subframe 4

2.6 Tabel Kode Adaptif dan Tabel Kode Eksitasi

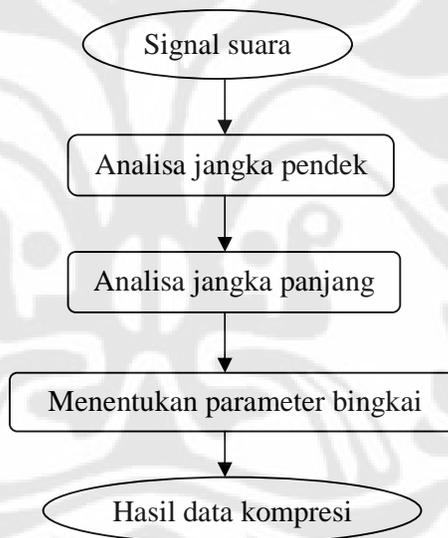
Tabel kode adaptif dan dua tabel kode eksitasi adalah nilai yang akan menjadi basis vektor untuk dapat menganalisa pada rangkaian tertutup pencarian nilai eksitasi. Tabel Adaptif merupakan tabel yang diperbaharui setiap kali perhitungan untuk satu subframe. Tabel eksitasi terdiri dari dua tabel eksitasi dan setiap tabel eksitasi terdiri 7 basis vektor yang memberikan 2^m vektor basis. Setiap vektor basis terdiri dari 40 data yang akan dipergunakan untuk mengestimasi nilai eksitasi. Tabel kode eksitasi telah didapatkan dari standar IS54 yang terdapat pada dokumen TIA/EIA.[7]

BAB III

PERANCANGAN PROSES RANGKAIAN ENKODER VSELP

Perancangan rangkaian encoder VSELP harus mengimplementasikan proses analisa dan sintesa sinyal suara dan melakukan pencarian parameter-parameter dari kompresi suara VSELP. Rangkaian enkoder akan melakukan proses pencarian parameter dengan melakukan proses analisa jangka pendek dan proses jangka panjang. Rancangan enkoder VSELP harus bersesuaian dengan algoritma proses yang dilakukan oleh rangkaian enkoder VSELP.

Pada Gambar 3.1 Algoritma proses rangkaian enkoder VSELP :

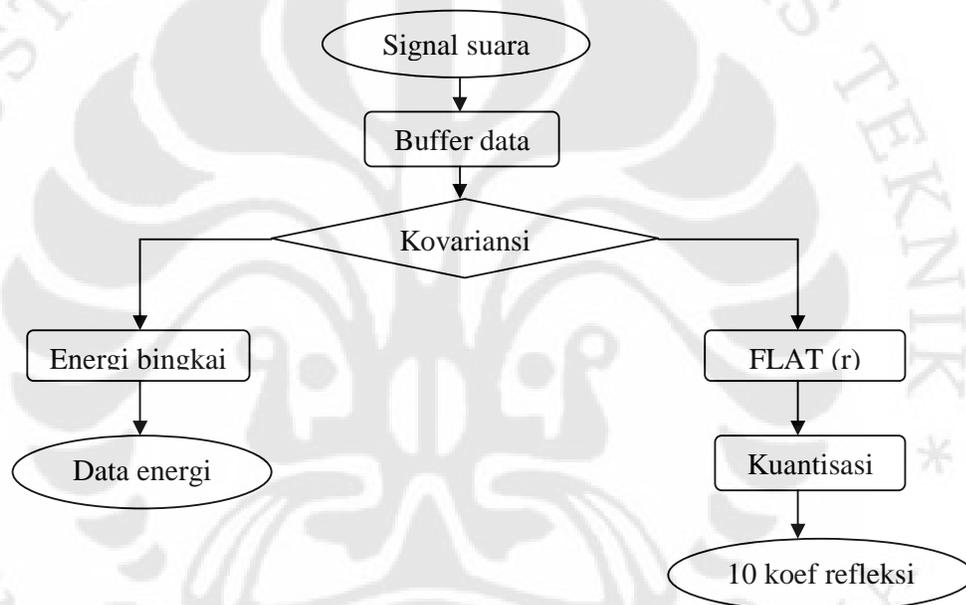


Gambar 3.1 Proses Enkoder VSELP

Algoritma rancangan enkoder secara garis besar melakukan tiga proses yaitu analisa jangka panjang dan analisa jangka pendek serta pemrosesan data-data perbingkai untuk mendapatkan urutan data yang tepat.

3.1 Analisa Jangka Pendek

Analisa jangka pendek dilakukan sebagai awal dari proses pencarian parameter all pole filter yang akan dipergunakan pada proses jangka panjang. Proses tersebut tersusun seperti pada Gambar 3.2. Proses pencarian all pole filter dilakukan dengan metode FLAT kemudian akan didapat koefisien refleksi yang akan diubah menjadi koefisien filter all pole. Pada analisa jangka pendek, analisa dilakukan perbingkai dengan 170 data yang diperbaharui 160 data. Proses analisa jangka pendek juga akan mendapatkan nilai energi bingkai.



Gambar 3.2 Proses Analisa Jangka Pendek

3.1.1 Koefisien Refleksi dan Energi Bingkai

Proses analisa jangka pendek dilakukan dengan memproses perbingkai data. Proses dilakukan dengan membuffer atau mengelompokkan data sebanyak 170 dengan *overlapped* atau diperbaharui 160 data dengan sisa 10 data lama. Hal tersebut dilakukan untuk perhitungan kovariansi seperti pada persamaan 2.12 dengan nilai N_a 170 dan n_p (nilai predictor) sebanyak 10. Kovariansi akan

diperoleh 11x11 data yang kemudian akan diproses dengan metode FLAT yang akan menghasilkan sepuluh koefisien refleksi.

Pada pemrosesan analisa jangka pendek, nilai energi bingkai dapat diperoleh dengan persamaan berikut :

$$R = \frac{\varphi(0,0)+\varphi(10,10)}{320} \quad (3.1)$$

Nilai energi bingkai ini merupakan nilai dari energi bingkai rata-rata yang akan memberikan Gambaran energi dari bingkai tersebut. Nilai energi bingkai akan memberikan parameter energi untuk menghitung parameter gain dari subbingkai.

Koefisien refleksi (r_i)dikuantisasi menggunakan tabel kode yang didisain untuk setiap koefisien refleksi. Alokasi bit untuk pengkuantisasian koefisien refleksi adalah:

r_1 dikuantisasi 64 tingkat	(6 bit)
r_2 dikuantisasi 32 tingkat	(5 bit)
r_3 dikuantisasi 32 tingkat	(5 bit)
r_4 dikuantisasi 16 tingkat	(4 bit)
r_5 dikuantisasi 16 tingkat	(4 bit)
r_6 dikuantisasi 8 tingkat	(3 bit)
r_7 dikuantisasi 8 tingkat	(3 bit)
r_8 dikuantisasi 8 tingkat	(3 bit)
r_9 dikuantisasi 8 tingkat	(3 bit)
r_{10} dikuantisasi 4 tingkat	(2 bit)

Nilai untuk tiap koefisien refleksi r_i diwakilkan dengan nilai r_i terdekat yang terdapat dalam tabel kode koefisien refleksi, sedangkan untuk parameter energi bingkai dikuantisasi 32 tingkat (5 bit) yang berkisar dari minimum -64 dB sampai dengan maksimum -4dB dengan interval 2dB.[7]

3.1.2 Interpolasi Parameter Analisa Jangka Pendek

Nilai energi bingkai dan koefisien refleksi akan dipergunakan untuk analisa jangka panjang. Nilai dari keduanya akan diinterpolasi agar menghasilkan nilai yang stabil untuk proses analisa subframe. Setiap energi bingkai dan koefisien filter *all pole* akan diinterpolasi sebanyak empat kali untuk masing-

masing subframe. Koefisien refleksi yang terkuantisasi dikonversi kedalam bentuk alpha untuk menjadi koefisien filter *all pole* sebelum dilakukan interpolasi.

Konversi yang dilakukan untuk mengubah koefisien r_i ke a_i menggunakan persamaan berikut :

$$a^{(i)}_i = r_i \quad (3.2)$$

$$a^{(i)}_j = a^{(i-1)}_j + r_i a^{(i-1)}_{i-j} \quad (3.3)$$

dimana $i = 1, 2, \dots, p$

$j = 1, 2, \dots, i-1$

Proses interpolasi dilakukan secara linear untuk masing-masing koefisien all pole filter (α_i) untuk mendapatkan parameter setiap subbingkai. Proses interpolasi α_i adalah sebagai berikut:

$$\alpha_i = (0,75)\alpha_{i,(\text{sebelum})} + (0,25)\alpha_{i,(\text{saat ini})} \quad \text{untuk subbingkai 1} \quad (3.4)$$

$$\alpha_i = (0,5)\alpha_{i,(\text{sebelum})} + (0,5)\alpha_{i,(\text{saat ini})} \quad \text{untuk subbingkai 2} \quad (3.5)$$

$$\alpha_i = (0,25)\alpha_{i,(\text{sebelum})} + (0,75)\alpha_{i,(\text{saat ini})} \quad \text{untuk subbingkai 3} \quad (3.6)$$

$$\alpha_i = \alpha_{i,(\text{saat ini})} \quad \text{untuk subbingkai 4} \quad (3.7)$$

Setelah proses interpolasi, dilakukan pengetesan stabilitas terlebih dahulu dengan merubah kembali koefisien prediksi (α_i) menjadi koefisien refleksi (r_i), dimana stabilitas filter dikatakan baik bila nilainya $-1 \leq r_i \leq 1$. Jika salah satu koefisien refleksi (r_i) ada yang tidak stabil ($|r_i| > 1$) maka akan dilakukan penyesuaian dengan ketentuan sebagai berikut, jika:

- $r_{i(1)}$ tidak stabil maka akan digunakan koefisien $r_{i, \text{sebelum}}$
- $r_{i(3)}$ tidak stabil maka akan digunakan koefisien $r_{i, \text{saat ini}}$
- $r_{i(2)}$ tidak stabil maka akan digunakan koefisien $r_{i, \text{sebelum}}$ atau koefisien $r_{i, \text{saat ini}}$ tergantung mana yang lebih besar energi bingkainya, jika sama maka $r_{i(2)}$ akan menggunakan koefisien $r_{i, \text{sebelum}}$

Proses interpolasi juga dilakukan pada parameter lainnya yaitu parameter energi bingkai (R). $R'_q(0)$ didefinisikan sebagai nilai dari bingkai yang telah diproses dan $R_q(0)$ adalah nilai koefisien dari bingkai sebelum diproses, dengan ketentuan

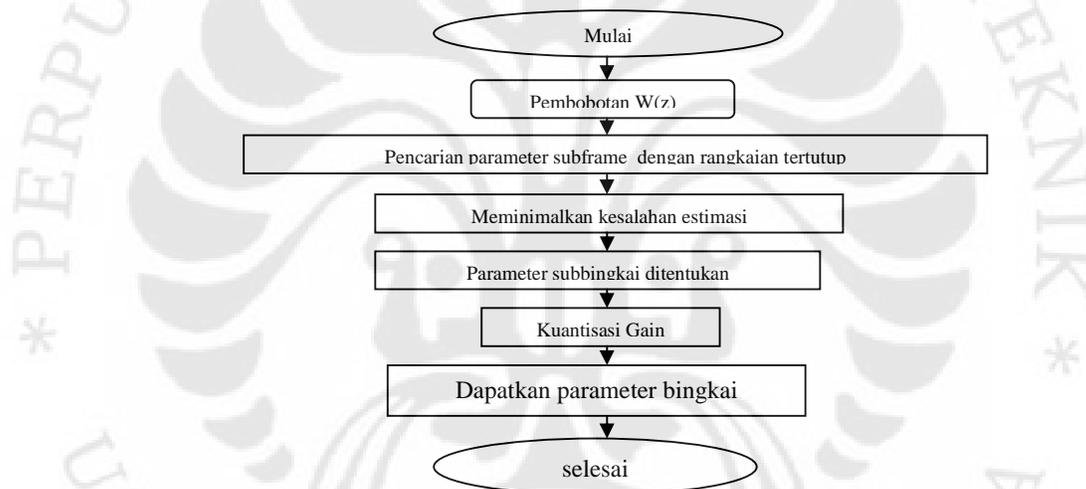
$$R'_q(0) = R_q(0)_{\text{bingkai sebelumnya}} \quad \text{untuk subbingkai 1} \quad (3.8)$$

$$R'_q(0) = R_q(0)_{\text{bingkai saat ini}} \quad \text{untuk subbingkai 3 \& 4} \quad (3.9)$$

$$R'_q(0) = \sqrt{R_q(0)_{\text{bingkai sebelumnya}} \cdot R_q(0)_{\text{bingkai saat ini}}} \quad \text{untuk} \\ \text{subbingkai 2} \quad (3.10)$$

3.2 Analisa Jangka Panjang

Analisa jangka panjang adalah analisa yang paling penting dalam rangkaian enkoder VSELP. Analisa tersebut menganalisa dan mengestimasi nilai dari residu atau eksitasi. Analisa jangka panjang menganalisa data suara persubbingkai yaitu sebanyak 40 data. Setiap subbingkai akan dianalisa dan didapatkan parameter persubbingkai yang akan ditransmisikan. Pada Gambar 3.3 merupakan keseluruhan proses analisa jangka panjang :



Gambar 3.3 Proses Analisa Jangka Panjang

3.2.1 Filter Pembobotan Signal Suara

Signal suara akan diboboti sebelum dilakukan analisa jangka pendek. Karena frekuensi respon dari pendengaran manusia tidak datar, kepresisian kesalahan estimasi pada saat pengkodean pada bagian enkoder dan pada saat pendekodean pada sisi decoder dapat dioptimumkan dengan cara proses pembobotan. Filter memiliki fungsi alih sebagai berikut :

$$W(z) = \frac{1 - \sum_{i=1}^{10} \alpha_i z^{-i}}{1 - \sum_{i=1}^{10} \alpha_i v^i z^{-i}} = H(z)/A(z) \quad (3.11)$$

Filter tersebut diimplementasikan secara *cascade* dengan dua *direct form filter*, *all zero filter* diikuti dengan *all pole filter*. [8][10][11] Filter tersebut mempergunakan koefisien filter interpolasi dengan nilai v adalah parameter pembobotan gangguan yang mengontrol kesalahan yang diperbolehkan pada daerah *formant* atau daerah frekuensi pipa rongga suara. Secara umum nilai v adalah 0.8 telah memberikan efek pengontrolan yang baik. [8]

3.2.2 Pencarian Lag atau Periode Frekuensi Fundamental

Pada perancangan rangkaian enkoder VSELP, parameter lag dapat ditentukan dengan menganalisa $p(n)$. Nilai lag pada dikuantisasi sebanyak 128 tingkat yaitu pada data ke 20 sampai dengan 147 dengan satu lag bernilai nol untuk menandakan ketidakadaan suara. Pencarian lag mempergunakan rangkaian terbuka memberikan tingkat estimasi cukup baik. [12]

Pencarian lag dilakukan dengan memaksimalkan persamaan :

$$E_i = \sum_{n=0}^{N-1} \{ p(n) - \beta_i g_i(n) \}^2 \quad (3.12)$$

dimana E adalah nilai kesalahan estimasi dan $g(n)$ adalah nilai vektor lag dengan β adalah penguatannya. Persamaan (3.12) dapat disederhanakan menjadi :

$$E_i = \sum_{n=0}^{N-1} p(n) - \frac{C_i^2}{G_i} \quad (3.13)$$

Dengan

$$C_i = \sum_{n=0}^{N-1} g_i(n) p(n) \quad (3.14)$$

yang merupakan nilai autokorelasi antara nilai eksitasi persubbingkai dengan vektor respons dari filter *all pole filter* (g_i) dan

$$G_i = \sum_{n=0}^{N-1} g_i(n) g_i(n) \quad (3.15)$$

merupakan autokorelasi dari g_i .

Pencarian lag dapat dilakukan dengan menggunakan rangkaian terbuka dengan memaksimalkan nilai C_i^2/G_i dengan persamaan

$$\frac{[\sum_{n=0}^{N-1} p(n) r^{n-lag}]^2}{\sum_{n=0}^{N-1} r^{2(n-lag)}} \quad (3.16)$$

Nilai $p(n\text{-lag})$ didapat dari pembufferan terhadap $p(n)$ sebanyak 187 data dengan pembaharuan data sebanyak 40 data. Kemudian Nilai beta akan didapatkan dengan persamaan :

$$\beta = C_i/G_i \quad (3.17)$$

Karena kita membatasi nilai beta yang positif, nilai lag yang memiliki C_i yang negatif tidak akan diperhitungkan.

3.2.3 Filter State

Filter state adalah filter yang akan memberikan nilai vektor $bL(n)$ atau vektor analisa jangka panjang sesuai dengan lag pada subframe tersebut. Filter tersebut akan mengeluarkan vektor yang bersesuaian dengan nilai lag dari kode buku adaptif dengan persamaan :

$$bL(n) = r(n - xLag), 0 \leq n \leq 39 \quad (3.18)$$

Dimana x adalah fungsi pembulatan jika nilai lag kurang dari 40 maka x akan bernilai dua dan jika nilai lag lebih dari 40 maka nilai x akan bernilai satu. Nilai tersebut akan memberikan nilai lag akan melebihi nilai dari n sehingga vektor akan selalu berjumlah 40 data.[7]

3.2.4 Pencarian Parameter Kode Buku Eksitasi

Pencarian nilai yang tepat untuk estimasi nilai eksitasi pada VSELP dilakukan dengan mencari nilai vektor kode buku yang tepat. Nilai eksitasi akan diwakilkan dengan dua vektor dari dua kode buku eksitasi. Setiap kode buku terdiri dari tujuh vektor basis (v) yang terdiri dari 40 data yang akan memberikan 128 kombinasi vektor yang akan dicari vektor yang tepat. Vektor yang memberikan nilai kesalahan yang kecil akan ditetapkan sebagai vektor kode eksitasi seperti pada persamaan 3.13.

Kombinasi vektor kode buku ($u_{k,i}$) dilakukan dengan cara mengkombinasikan penjumlahan ketujuh basis vektor dengan perumusan :

$$u_{k,i}(n) = \sum_{m=1}^M \theta_{im} \cdot v_{k,m}(n) \quad (3.19)$$

dengan k adalah kode buku ke- k , i adalah indeks dari banyak kode vector ($1 \leq i \leq 128$), m adalah banyaknya vektor basis ($1 \leq m \leq 7$), θ_{im} adalah parameter yang menentukan kombinasi linier pembentukan vektor kode dengan ketentuan yaitu:

$\theta_{im} = +1$ jika bit m dari codeword $i = 1$

$\theta_{im} = -1$ jika bit m dari codeword $i = 0$

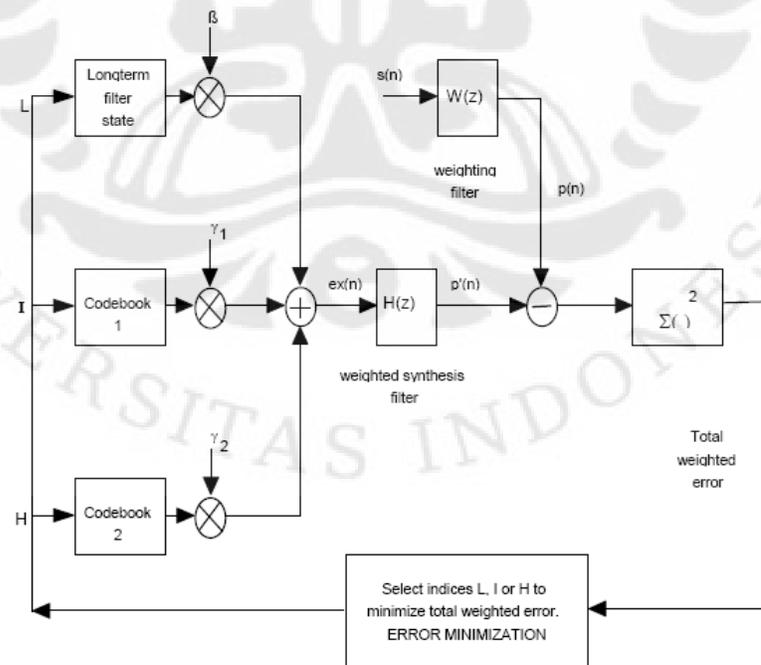
pembentukan vektor kode I berasal dari penjumlahan M basis vektor yang tanda operasinya (+/-) ditentukan oleh nilai bit dalam indeks i [1],[2],[4],[5],[8],[9],[11]. Misalkan untuk indeks 1 (dalam biner 7 bit: 0000001) mempunyai $\theta_{11} = 1$ dan $\theta_{12}=\theta_{13}=\dots=\theta_{17} = -1$, maka besar vektor kode 1 dalam tabel kode pertama dapat dituliskan:

$$u_{1,1}(n) = \sum_{m=1}^M \theta_{im} \cdot v_{1,m}(n) = v_{11}(n) - v_{12}(n) - \dots - v_{17}(n) \quad (3.20)$$

Pencarian kode buku eksitasi yang tepat dilakukan dengan metode rangkaian tertutup dan mencari secara berulang dari kombinasi dari dua kode buku eksitasi dan satu kode buku adaptif sesuai dengan lag yang difilter dengan filter $1/A(z)$ yang memberikan nilai kesalahan estimasi yang kecil seperti pada Gambar 3.4. Persamaan yang harus diminimalkan adalah :

$$E_i = \sum_{n=1}^{N-1} [p(n) - \beta c_0(n) + \gamma_1 c_1(n) + \gamma_2 c_2(n)]^2 \quad (3.21)$$

Pencarian kode buku eksitasi dapat dilakukan masing-masing secara tertutup dengan metode orthogonal *Gram-Schmidt*.



Gambar 3.4 Proses Pemilihan Indeks Kode Buku [4]

Pencarian tabel kode eksitasi dilakukan setelah ditentukan nilai prediksi lag jangka panjang, L . Prosedur pencarian tabel kode secara berurutan dipilih satu vektor kode dari tabel kode eksitasi VSELP pertama dan kemudian memilih satu vektor kode dari tabel kode eksitasi VSELP kedua.

Kemudian untuk memulai pencarian vektor kode diperlukan $q_{k,m}(n)$ yang merupakan *zero state response* dari tiap vector basis $v_{k,m}(n)$ yang berasal dari filter $H(z)$, hal ini diperlukan untuk mendapatkan vector kode terfilter (*filtered codevector*) yang dapat dirumuskan,

$$f_{k,i}(n) = \sum_{m=1}^M \theta_{im} \cdot q_{k,m}(n) \quad (3.22)$$

dimana $M=7$ dan $0 \leq i \leq 2^M - 1$ serta n adalah banyaknya sampel ($N=40$).

Penentuan vektor kode dari tabel kode pertama harus sesuai dengan hasil pemilihan nilai lag, L , sebelumnya. Penentuan vektor kode dari tabel kode kedua harus sesuai dengan hasil pemilihan nilai lag dan *codeword* terpilih dari tabel kode pertama, I . Salah satu teknik yang dapat diterapkan menggunakan prosedur ortogonalisasi untuk memisahkan proses pemilihan vektor tabel kode eksitasi dari komponen eksitasi yang telah ditentukan sebelumnya. Ortogonalisasi dari basis vektor yang terfilter didapatkan dengan mengimplementasikan metode *gram-schmidt*.

Setiap basis vektor yang terfilter dari tabel kode pertama, $q_{1,m}(n)$, diortogonalkan dengan $b'_L(n)$, yang merupakan *zero state response* $H(z)$ terhadap vektor prediksi jangka panjang $b_L(n)$.

Didefinisikan:

$$\Gamma = \sum_{n=0}^{N-1} (b'_L(n))^2 \quad (3.23)$$

dan

$$\psi_m = \sum_{n=0}^{N-1} b'_L(n) q_{1,m}(n) \quad (3.24)$$

untuk $1 \leq m \leq M$; maka $q'_{1,m}(n)$, basis vektor terortogonalisasi dapat dihitung dengan:

$$q'_{1,m}(n) = q_{1,m}(n) - \left(\frac{\psi_m}{\Gamma} \right) b'_L(n) \quad (3.25)$$

untuk $1 \leq m \leq M$ dan $0 \leq N \leq n-1$.

Sehingga vektor kode terfilter dan terortogonalisasi dapat diekspresikan dengan:

$$f'_{1,i}(n) = \sum_{m=1}^M \theta_{im} q'_{1,m}(n) \quad (3.26)$$

untuk $1 \leq m \leq M$ dan $0 \leq N \leq n-1$.

Vektor basis terfilter untuk tabel kode kedua diortogonalisasikan dengan $b'_L(n)$ dan $f'_{k,i}(n)$. Hal ini dapat diselesaikan pertama kali dengan mengortogonalisasikan vektor $q'_{2,m}(n)$ terhadap $b'_L(n)$ dengan cara yang sama seperti pada tabel kode pertama. Hasil vektor ini kemudian diortogonalisasi terhadap $f'_{1,l}(n)$ menggunakan prosedur serupa. Karena $f'_{1,l}(n)$ adalah ortogonal dari $b'_L(n)$, hasil vektor $q'_{2,m}(n)$, akan diortogonal kepada $b'_L(n)$ dan $f'_{1,l}(n)$.

Selanjutnya vektor kode terfilter dan terortogonalisasi untuk tabel kode kedua dapat diekspresikan dengan:

$$f'_{2,i}(n) = \sum_{m=1}^M \theta_{im} q'_{2,m}(n) \quad (3.27)$$

untuk $0 \leq i \leq 2^M - 1$, $0 \leq n \leq N - 1$.

Selanjutnya untuk mendapatkan codeword i dengan menyederhanakan persamaan,

$$E'_{k,i} = \sum_{n=0}^{N-1} (p(n) - \gamma'_k f'_{k,i}(n))^2 \quad (3.28)$$

dimana $k=1$ untuk tabel kode pertama dan $k=2$ untuk tabel kode kedua dan dimana γ_k optimal untuk setiap vektor kode i . Setelah kita mendapatkan basis vektor terfilter dan terortogonalisasi, prosedur pencarian tabel kode selanjutnya identik. Vektor yang terpilih adalah yang memberikan nilai C_i^2/G_i yang maksimal. Nilai penguatan dari masing-masing vektor akan didapatkan dengan persamaan 3.17.

3.2.5 Transformasi Penguatan ke Dalam GS, P0, P1

Nilai dari penguatan masing-masing kode buku baik adaptif maupun eksitasi akan dikuantisasi dengan transformasi ke dalam bentuk GS,P0,P1. Nilai penguatan tersebut akan dikuantisasi sebanyak 256 tingkat kuantisasi dengan metode pengkuantisasian vektor.[7] Hal tersebut dilakukan untuk memberikan nilai kesalahan estimasi yang kecil. Penguantisasian satu persatu tidak dapat dilakukan karena keterikatan ketiga kode buku sebagai vektor kombinasi.

Hubungan antara β , γ_1 , dan γ_2 dengan GS, P0, P1 dapat dilihat dari rumus di bawah ini:

$$\beta = \sqrt{\frac{RS.GS.P0}{R_x(0)}} \quad (3.29)$$

$$\gamma_1 = \sqrt{\frac{RS.GS.P1}{R_x(1)}} \quad (3.30)$$

$$\gamma_2 = \sqrt{\frac{RS.GS(1-P0-P1)}{R_x(2)}} \quad (3.31)$$

Nilai autokorelasi vektor kode buku ke-k adalah

$$R_x(k) = \sum_{n=0}^{N-1} c^2 k(n) \quad (3.32)$$

Nilai estimasi energi residu adalah :

$$RS = NR^2 q(0) \prod_{i=1}^{np} (1 - r_i)^2 \quad (3.33)$$

Nilai GS adalah :

$$R = GS RS \quad (3.34)$$

Vektor kuantisasi dari nilai penguatan dilakukan dengan transformasi ke bentuk GS,P0,P1. Vektor kuantisasi tersebut dilakukan dengan meminimalkan persamaan berikut :

$$E = Rpp - \sqrt{GSP0} - b\sqrt{GSP1} - c\sqrt{GS(1-P0-P1)} + dGS\sqrt{P0P1} + eGS\sqrt{P0(1-P0-P1)} + fGSP0 + hGSP1 + iGS(1-P0-P1) \quad (3.35)$$

Dimana :

$$a = 2Rpc(0) \sqrt{\frac{RS}{R_x(0)}} \quad (3.36)$$

$$b = 2Rpc(1) \sqrt{\frac{RS}{R_x(1)}} \quad (3.37)$$

$$c = 2Rpc(2) \sqrt{\frac{RS}{R_x(2)}} \quad (3.38)$$

$$d = \frac{2Rcc(0,1)RS}{\sqrt{R_x(0)R_x(1)}} \quad (3.39)$$

$$e = \frac{2Rcc(0,2)RS}{\sqrt{R_x(0)R_x(2)}} \quad (3.40)$$

$$f = \frac{2Rcc(1,2)RS}{\sqrt{R_x(1)R_x(2)}} \quad (3.41)$$

$$g = \frac{Rcc(0,0)RS}{R_x(0)} \quad (3.42)$$

$$h = \frac{Rcc(1,1)RS}{R_x(1)} \quad (3.43)$$

$$i = \frac{R_{cc(2,2)RS}}{R_{x(2)}} \quad (3.44)$$

Nilai energi eksitasi residu adalah

$$R_{pp} = \sum_{n=0}^{N-1} p(n)p(n) \quad (3.45)$$

Nilai korelasi berseberangan antara nilai residu dengan vektor tiap kode buku yang merupakan respon filter $H(z)$

$$R_{pc} = \sum_{n=0}^{N-1} p(n)c^k(n) ; k=0,1,2 \quad (3.46)$$

Nilai korelasi berseberangan antara vektor kode buku respon terhadap filter $H(z)$:

$$R_{cc} = \sum_{n=0}^{N-1} c^k(n)c^j(n) ; k,j=0,1,2 \quad (3.47)$$

Nilai setiap vektor $GS, P0, P1$ akan dimasukkan kedalam persamaan 3.35 dan dipilih dari ke 256 vektor penguatan yang memberikan nilai error yang kecil.

3.2.6 Pembaharuan Nilai Kode Buku Adaptif

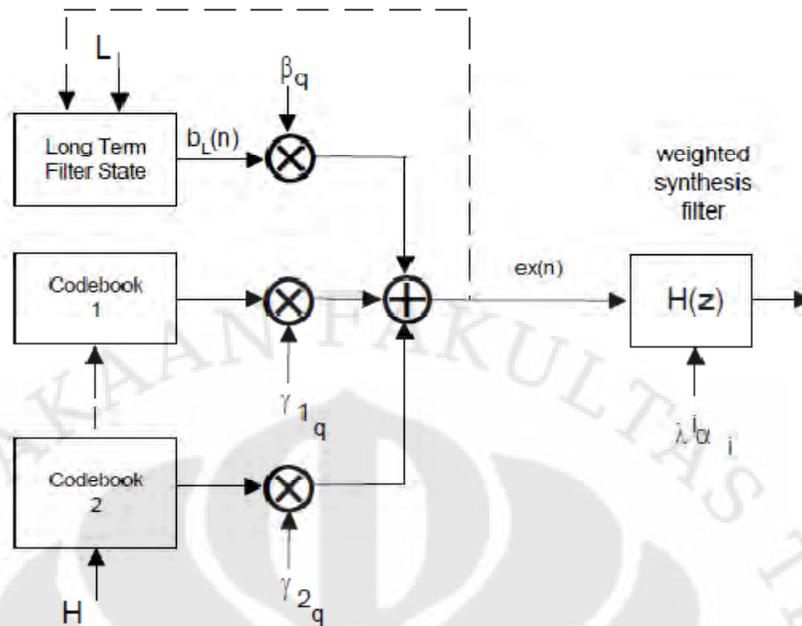
Kode buku adaptif diperbaharui setiap kali proses pencarian kode buku eksitasi. Kode buku adaptif terdiri dari 187 data dengan 40 data adalah data sekarang dan 147 data adalah data keluaran sebelumnya. Proses pembaharuan data oleh vektor $ex(n)$ yang merupakan vektor kombinasi kode buku dengan penguatan yang terkuantisasi dengan persamaan :

$$ex(n) = \beta q bL(n) + \gamma_1 q c_1 + \gamma_2 q c_2 \quad (3.48)$$

dimana c_1 dan c_2 adalah kode buku eksitasi, $bL(n)$ adalah vektor filter state , dan dengan masing-masing penguatan yang terkuantisasi. Pembaharuan kode buku adaptif akan dilakukan seperti pada Gambar 3.5 dengan persamaan :

$$r(n) = r(n + 40) ; -146 < n < -41 \quad (3.49)$$

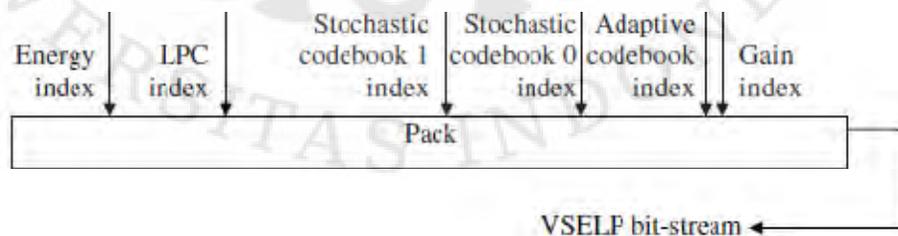
$$r(n) = ex(n + 40) ; -40 < n < -1 \quad (3.50)$$



Gambar 3.5 Proses Pembaharuan Kode Buku Adaptif [10]

3.2.7 Proses Penyatuan Parameter Bingkai

Proses penyatuan parameter bingkai dilakukan untuk setiap pemrosesan setiap bingkai. Parameter-Parameter tiap bingkai akan disusun sesuai dengan ketentuan yang berlaku. Parameter tiap bingkai disusun berdasarkan urutan proses yang dilakukan. Tiap bingkai memiliki 27 parameter yang akan disatukan dalam bingkai parameter seperti pada Gambar 3.6.[2]



Gambar 3.6 Struktur Data Kompresi VSELP [1]

Parameter-parameter tiap bingkai merupakan indeks dari kuantisasi atau indeks kode buku yang disimpan dalam memori dari enkoder dan decoder. Data kompresi VSELP adalah data informasi indeks yang akan merepresentasikan signal suara yang diestimasi atau dikompresi. Setiap parameter bingkai terdiri dari 160 bit data yang akan memberikan data kompresi suara sebanyak 8000 data perdetik atau 8 kilo bit data suara perdetik dengan waktu sampling 8000Hz.

Parameter-parameter VSELP dirancang hanya untuk melakukan kompresi pada suara manusia saja. Sehingga kompresi hanya dilakukan untuk memberikan kualitas komunikasi yang baik saja bukan untuk memberikan kualitas suara yang baik sekali.



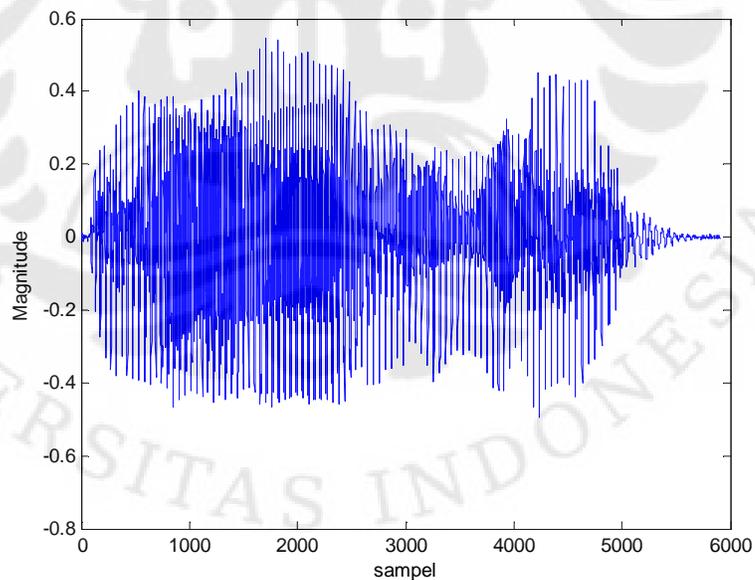
BAB IV

SIMULASI DAN HASIL PROSES RANGKAIAN ENKODER VSELP

Proses Encoder VSELP pada simulasi yang dilakukan didasarkan pada algoritma pemrosesan yang dilakukan oleh enkoder VSELP. Proses simulasi dilakukan dengan berbantu perangkat lunak SIMULINK oleh MATLAB. Proses Simulasi dilakukan untuk memberikan Gambaran serta membuktikan tingkat kompresi yang dilakukan oleh algoritma pengkompresian VSELP.

4.1 Simulasi Rangkaian Enkoder

Simulasi rangkaian enkoder VSELP berbantu peranti lunak Simulink dilakukan secara *offline* atau dilakukan pengkompresian tidak secara kontinu. Simulasi dilakukan dengan terlebih dahulu merekam suara dan melakukan satu persatu kompresi bukan dilakukan dengan data suara yang kontinu dan langsung dikompresi. Sumber suara dengan kata 'saya' seperti pada Gambar 4.1.



Gambar 4.1 Sampel Suara 'Saya' dengan basis waktu

Proses Simulasi yang dilakukan mempergunakan sumber suara rekaman. Sumber suara tersebut direkam mempergunakan alat perekam dengan format 32 bit kuantisasi PCM dan 8000 Hz. Format data suara tersebut sudah memberikan pemodelan terhadap data suara pada komunikasi GSM.

4.2 Hasil Simulasi

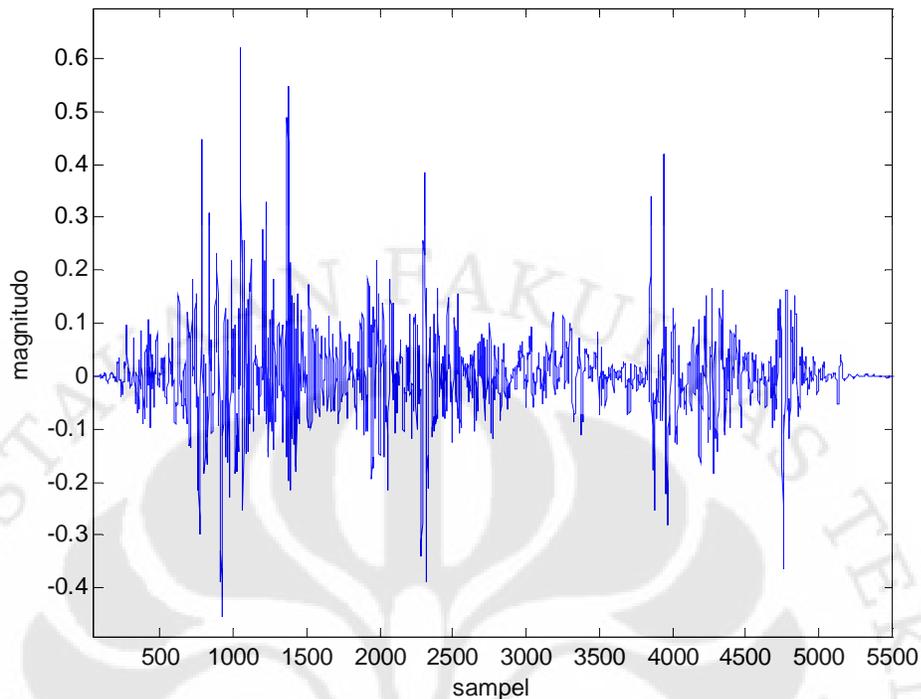
Simulasi rangkaian enkoder dilakukan untuk melakukan pembuktian terhadap tingkat kompresi suara dan melihat proses-proses yang dilakukan oleh rangkaian enkoder serta melihat bagaimana pemodelan yang dilakukan oleh teknik kompresi VSELP. Hasil dari proses simulasi rangkaian enkoder VSELP adalah data kompresi yang merupakan indeks dari kode buku. Dari simulasi kita juga dapat melihat bentuk dari sinyal suara sintesa yang masih kasar.

Proses Simulasi mempergunakan sampel suara 'saya' seperti pada Gambar 4.1 dengan durasi 22 ms. Sampel suara tersebut kemudian dilewatkan kepada proses enkoder VSELP dan didapatkan hasil kompresi yang berupa 27 parameter perbingkai.

Data suara tersebut dengan pencuplikan 8000Hz (8000 sampel perdetik) akan mempunyai 5920 sampel suara dengan 37 bingkai analisa suara.

Data suara 'saya' tersimpan dengan besar data sebesar 24 kilobyte pada file MATLAB. Hasil simulasi menunjukkan bahwa dari 37 bingkai analisa akan didapatkan 37 kali 27 parameter yang akan memberikan 999 parameter yang tersimpan pada MATLAB dengan besar satu kilobyte. Kompresi yang dilakukan mencapai 24 kali dari semula. Hasil Sintesa suara juga terlihat dari hasil simulasi.

Bentuk dari gelombang sintesa suara dari simulasi enkoder dapat dilihat dengan memasukkan data $ex(n)$ ke filter $A(z)$ yang terboboti. Filter tersebut merupakan sebuah proses awal dari rangkaian dekoder. Hasil keluaran dari filter tersebut merupakan Gambaran kasar dari sinyal sintesa suara hasil kompresi oleh rankaian enkoder VSELP.



Gambar 4.2 Hasil Suara Sintesa Sederhana

Hasil suara sintesa dengan algoritma pengkodean VSELP hasil simulasi secara kualitatif tidak dapat memberikan kualitas yang baik untuk teknik pengkompresian. Pada Gambar 4.2 kita dapat melihat dari bentuk gelombang bahwa kecocokan suara pada hasil kompresi dengan suara asli adalah bentuk envelop yang memberikan informasi yang utama pada pengkompresian suara.

4.3 Waktu Pemrosesan Simulasi Rangkaian Enkoder VSELP

Proses simulasi rangkaian enkoder dengan sumber suara 'saya' dengan durasi 22ms dilakukan dengan waktu sekitar 30 detik. Proses rangkaian enkoder untuk memerlukan proses yang panjang karena mempergunakan algoritma yang tidak sederhana sehingga waktu yang dibutuhkan untuk memproses suara lebih lama. Hal tersebut tidak mungkin dapat dilakukan untuk memberikan proses komunikasi yang baik karena keterlambatan penerimaan suara akan terjadi.

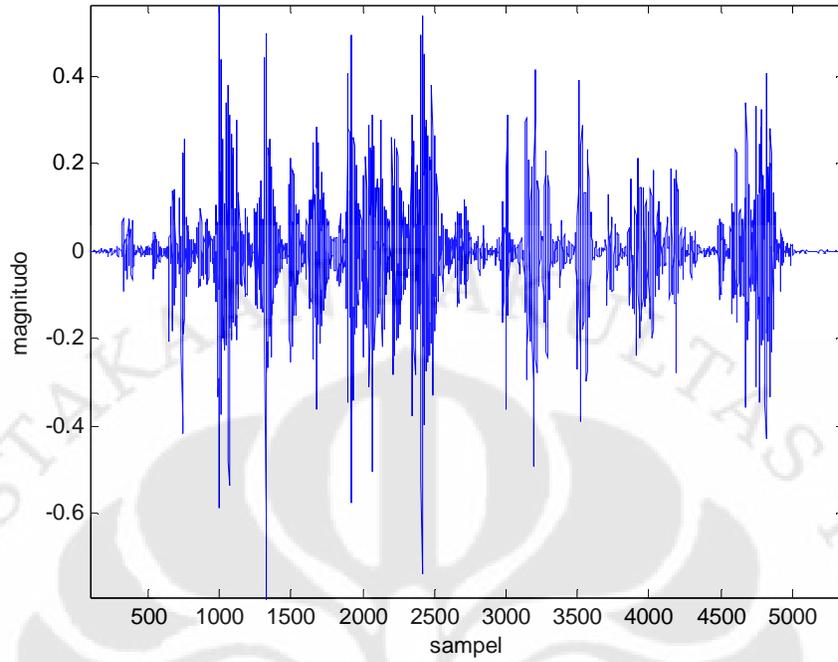
4.4 Analisa Proses Rangkaian Enkoder VSELP

Proses kompresi oleh algoritma VSELP memberikan tingkat kompresi yang baik untuk telekomunikasi. Sumber daya frekuensi yang terbatas dapat teratasi dengan penggunaan kompresi suara VSELP tersebut. Proses kompresi yang mencapai delapan kilo bit perdetik akan mampu memberikan kapasitas yang cukup pemanfaatan kanal suara. Tingkat kompresi juga harus ditingkatkan lagi untuk mendapatkan kompresi suara yang lebih rendah lagi karena kebutuhan masa depan akan kapasitas kanal pasti meningkat dengan bertumbuhnya pengguna selular.

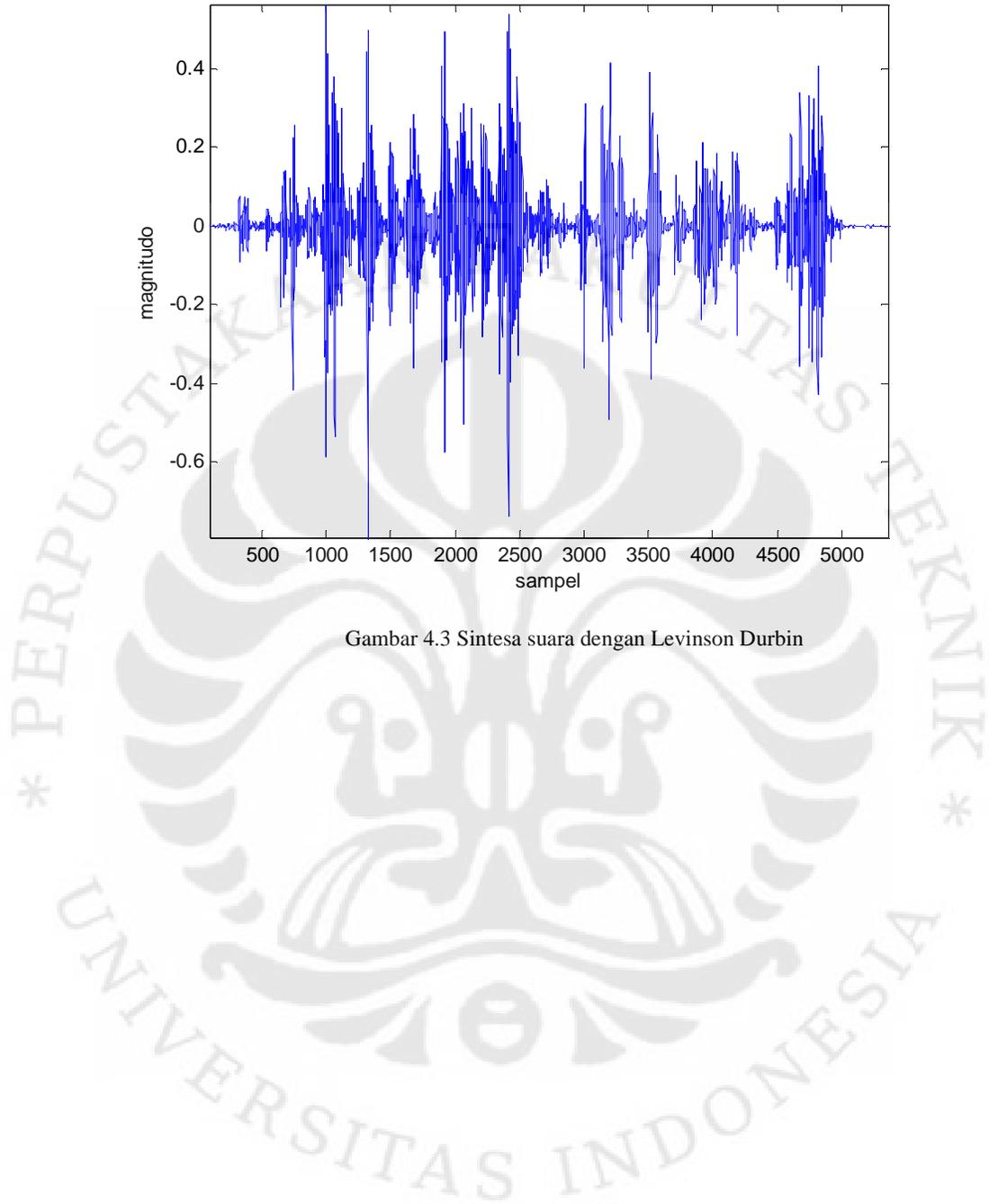
Proses enkoder VSELP juga membutuhkan waktu yang cukup lama terlihat dari simulasi tersebut. Proses komputasi yang panjang harus diatasi agar keterlambatan penerimaan suara tidak terjadi pada rangkaian penerima. Algoritma proses enkoder harus dipersingkat dan kemampuan prosesor yang cepat harus dipergunakan untuk dapat mengatasi masalah tersebut.

Hasil kualitatif pada simulasi menggunakan algoritma FLAT untuk memproses sinyal suara guna mendapatkan koefisien filter LPC kurang memberikan suara yang baik. Percobaan simulasi dilakukan lagi dengan metode Levinson Durbin yang merupakan pemodelan terhadap keluaran sebelumnya saja untuk pemodelan koefisien filter LPC.

Hasil Suara yang dilakukan dengan metode tersebut secara bentuk gelombang belum memberikan kualitas yang baik terlihat pada Gambar 4.3. Akan tetapi telah dapat memberikan kualitas yang lebih baik terhadap hasil suara sintesa. Suara sintesa dengan algoritma Levinson-Durbin lebih baik periodisitasnya dibandingkan suara sintesa dengan algoritma FLAT. Hal tersebut dikarenakan kesalahan kuantisasi pada FLAT dibanding dengan Levinson Durbin lebih besar.



Gambar 4.3 Sintesa suara dengan Levinson Durbin



BAB V

KESIMPULAN

Dari hasil yang dilakukan dapat ditarik kesimpulan bahwa :

1. Tingkat kompresi dari VSELP adalah sebesar delapan kilo bit perdetik.
2. Rangkaian enkoder VSELP melakukan proses komputasi yang kompleks sehingga waktu proses yang dilakukan rangkaian enkoder cukup lama.
3. Proses komputasi VSELP memerlukan prosesor yang mendukung komputasi yang cepat dan sanggup menangani komputasi yang kompleks.
4. VSELP memberikan pemodelan dan estimasi yang baik untuk mengkompresi suara.
5. LPC merupakan bagian terpenting pada bagian enkoder VSELP untuk membentuk estimasi gelombang sinyal suara yang baik.

DAFTAR REFERENSI

- [1] Chu, C Wai. 2003. *Speech Coding Algorithms : Foundation and Evolution of Standardized*. New Jersey : John Wiley & Sons Inc
- [2]Saleh, Muhammad.1996. *Implementasi Teknik VSELP Pada Pengkodean Suara Laju Bit Rendah*. Depok: Universitas Indonesia
- [3]Wong, W. T K, R.M. Mack, B.M.G. Cheetham dan X.Q. Sun. 1996. *Low Rate Speech Coding*. Inggris : Liverpool University
- [4]Gerson, Ira A, Mark A Jasiuk. *Vector Excited Linear Prediction (VSELP) Speech Coding at 8 kbps*. Proc. IEEE Int. Conf. Acoustic, Speech and Signal Processing. April 1990
- [5]Gerson, Ira A, Mark A Jasiuk. *Vector Excited Linear Prediction (VSELP) Speech Coding at 4,8 kbps*. Chicago Corporate Result and Development Centre Motorola Inc. 1990
- [6] Li, Wei, Anu Sridhar, Tina Teng. 1999. *Comparison of Speech Coding Algorithms: ADPCM, CELP, and VSELP*. Project for EE 6390, University of Texas
- [7] Le Blanc, W.P dan S.A. Mahmoud. *Structured Codebook Design in CELP*. Carleton University. Ottawa, Kanada
- [8] TIA/EIA Standard 136-420. TDMA Celluler/PCS VSELP. 1999
- [10] Sunwoo, Myung H, Sangil Park dan Kent Terry.1991.*A Real Time Implementation of Key VSELP Routines on a 16-bit DSP Chip*. Motorola, Inc. Austin, Texas

Lampiran 1: Tabel Kode Koefisien Kuantisasi

	LPC1	LPC2	LPC3	LPC4	LPC5
1	-0,9867044	-0,7547349	-0,8606231	-0,4505537	-0,6570091
2	-0,9810331	-0,5826730	-0,8046579	-0,2582354	-0,5339469
3	-0,9762308	-0,4569369	-0,7523137	-0,1276191	-0,4366474
4	-0,9711074	-0,3481136	-0,7056828	-0,0189162	-0,3503954
5	-0,9655631	-0,2492763	-0,6582848	0,0735594	-0,2729992
6	-0,9597059	-0,1585778	-0,6130494	0,1581760	-0,2044571
7	-0,9536623	-0,0772643	-0,5684005	0,2381239	-0,1395732
8	-0,9471911	-0,0050969	-0,5247685	0,3138437	-0,7738069
9	-0,9406522	0,0652718	-0,4832728	0,3864110	-0,1441931
10	-0,9339897	0,1329884	-0,4436872	0,4573097	0,4928109
11	-0,9266146	0,1978288	-0,4048591	0,5282919	0,1149447
12	-0,9190770	0,2600679	-0,3659545	0,5986876	0,1853068
13	-0,9110740	0,3186268	-0,3276270	0,6679506	0,2629969
14	-0,9032389	0,3747462	-0,2901177	0,7368154	0,3558021
15	-0,8951876	0,4288900	-0,2521926	0,8054330	0,4658669
16	-0,8865973	0,4810179	-0,2139451	0,8715001	0,6091661
17	-0,8775992	0,5302551	-0,1760749		
18	-0,8679655	0,5760165	-0,1370347		
19	-0,8578170	0,6193221	-0,0963786		
20	-0,8468435	0,6593915	-0,0557086		
21	-0,8350598	0,6967787	-0,0134266		
22	-0,8232460	0,7315258	0,0291324		
23	-0,8109311	0,7650170	0,0724380		
24	-0,7979615	0,7966733	0,1183170		
25	-0,7842678	0,8267812	0,1668848		
26	-0,7699829	0,8543012	0,2185972		
27	-0,7545527	0,8798118	0,2741918		
28	-0,7377605	0,9037305	0,3353715		
29	-0,7188396	0,9251339	0,4032742		
30	-0,6990101	0,9448361	0,4797808		
31	-0,6768131	0,9636774	0,5761803		
32	-0,6533867	0,9816108	0,6969622		
33	-0,6262213				
34	-0,5971571				
35	-0,5684631				
36	-0,5378258				
37	-0,5058868				
38	-0,4740323				
39	-0,4414438				
40	-0,4054018				
41	-0,3682330				
42	-0,3293100				

43 -0,2894667
 44 -0,2428349
 45 -0,1948891
 46 -0,1466630
 47 -0,0915260
 48 0,0269218
 49 0,0372767
 50 0,1093523
 51 0,1758577
 52 0,2397777
 53 0,3001486
 54 0,3555985
 55 0,4108348
 56 0,4679427
 57 0,5202129
 58 0,5746787
 59 0,6337371
 60 0,6966888
 61 0,7613552
 62 0,8211135
 63 0,8759806
 64 0,9311733

	LPC6	LPC7	LPC8	LPC9	LPC10
1	-0,3351014	-0,5834863	-0,3832901	-0,5373111	-0,1508205
2	-0,1157702	-0,4363778	-0,2111644	-0,3927697	0,0465244
3	0,0326587	-0,3007811	-0,0724298	-0,2594878	0,1958054
4	0,1545226	-0,1750740	0,0490140	-0,1391014	0,3621688
5	0,2685564	-0,0584889	0,1637184	-0,0293514	
6	0,3836386	0,0571851	0,2812095	0,0825513	
7	0,5067841	0,1880215	0,4174391	0,2107033	
8	0,6525010	0,3601036	0,5824320	0,3958447	

Lampiran 2: Tabel Kode Eksitasi 1

	BV 1	BV 2	BV 3	BV 4	BV 5	BV 6	BV 7
1	0,34512	-0,16493	0,12223	-0,20025	-0,64836	0,34142	-0,99150
2	-0,19089	-0,12806	-0,02444	0,30951	-0,01472	0,23058	-0,04214
3	-0,18638	0,20847	0,09521	0,11933	-0,39261	-0,11542	-0,20654
4	-0,28091	0,02897	0,20513	0,25020	-0,03888	0,04632	-0,06140
5	-0,05950	-0,12293	-0,26477	0,07345	0,02080	-0,06644	-0,02154
6	-0,28981	0,09743	-0,04311	0,28950	0,19575	0,08361	0,10041
7	-0,20070	0,05553	-0,27443	0,04940	-0,09889	-0,09363	0,18118
8	-0,26247	0,32947	0,04826	-0,16728	0,38250	-0,02741	0,05004
9	-0,23192	-0,13464	-0,14419	0,17612	0,08439	0,07113	0,25630
10	-0,18100	0,12565	0,17755	0,07342	0,30035	-0,12547	-0,15232
11	0,03789	-0,11684	0,22276	-0,17812	0,17688	0,07751	-0,02675
12	-0,21478	0,00623	0,41995	0,04386	0,22680	0,01797	-0,12967
13	0,12418	0,07953	0,34077	-0,04660	0,38401	0,50265	0,04938
14	-0,12186	0,31938	0,31649	-0,07254	-0,12177	0,07896	0,10620
15	0,02360	-0,27460	0,13951	-0,29067	0,14547	0,42794	0,19914
16	0,27133	-0,01363	-0,20506	-0,06229	-0,05188	0,21770	0,26956
17	0,16327	-0,17662	0,06458	-0,13099	0,11412	0,22893	-0,02002
18	-0,04970	-0,19182	-0,35379	0,07658	-0,02478	0,04618	0,29058
19	0,19028	-0,17463	-0,05985	0,17439	0,09596	0,16568	-0,00313
20	-0,01335	-0,20696	-0,29406	0,18551	0,14327	-0,05426	0,10164
21	0,09224	-0,15702	0,16444	0,35238	-0,05414	0,11283	0,12994
22	-0,05935	-0,26793	-0,09713	0,11465	0,00395	-0,03722	0,23030
23	0,02033	-0,02101	0,23452	0,15899	-0,25869	0,07740	0,29024
24	-0,20718	-0,20320	-0,11410	0,22222	-0,02090	-0,24436	0,18969
25	0,02096	-0,20447	0,32471	-0,01127	-0,21015	-0,18949	0,32532
26	-0,15494	-0,00484	-0,07059	-0,26819	-0,06068	0,14775	0,02585
27	-0,21406	-0,18205	-0,14674	-0,36624	-0,02704	-0,50434	-0,00263
28	-0,23299	-0,07024	-0,10935	-0,37178	0,06947	0,02714	-0,07936
29	-0,03249	-0,05201	0,12904	0,02992	0,13133	-0,13505	-0,09870
30	0,12201	-0,12365	-0,30674	-0,35340	0,06358	-0,03649	-0,16585
31	0,18095	0,10069	0,06986	-0,04227	0,02392	-0,10411	-0,10429
32	0,06743	-0,40631	0,33801	0,06729	0,06551	-0,27498	-0,15930
33	0,11957	0,13148	0,10607	-0,03312	0,05598	-0,05221	0,07002
34	0,39984	-0,00170	0,00421	0,35889	-0,04799	-0,33431	0,00916
35	0,27503	0,38494	0,25291	-0,23170	-0,06142	-0,34611	0,00556
36	0,13968	0,37618	-0,00532	0,11108	-0,31128	-0,03607	-0,02852
37	-0,22649	0,18326	-0,09390	-0,10489	-0,39184	0,08886	0,00439
38	0,24597	0,22552	-0,14084	0,06802	-0,08221	-0,02546	-0,07426
39	-0,14785	0,39523	-0,19451	0,03554	0,00293	0,09428	-0,11090
40	0,13496	0,19653	-0,03808	0,02773	-0,14520	-0,00650	-0,10565

Lampiran 3: Tabel Kode Eksitasi 2

	BV 1	BV 2	BV 3	BV 4	BV 5	BV6	BV 7
1	-1,07847	-0,61713	-0,70374	0,07885	0,04358	0,67365	-0,69421
2	0,32202	0,10425	0,17802	0,16417	-0,05806	0,18363	0,11140
3	0,35721	0,19043	0,13863	0,27696	0,45011	-0,11692	0,60137
4	0,60321	-0,20039	0,01774	0,31391	-0,20821	-0,23851	0,19518
5	0,32913	0,23642	0,31941	-0,18080	-0,35824	-0,03071	0,51339
6	-0,20682	-0,39313	-0,81562	-0,66968	-0,26136	0,36133	0,11329
7	-0,07851	-0,07452	-0,03730	0,23479	-0,67455	0,40771	0,14373
8	0,07766	-0,11137	0,11827	0,44272	0,53828	0,23598	-0,23927
9	-0,55393	-0,32968	-0,42213	0,39966	-0,01478	0,04120	-0,44718
10	-0,44997	0,05506	0,10084	0,54982	-0,05668	-0,18809	0,32583
11	-0,28748	-0,22215	0,34346	-0,18588	0,18989	-0,16328	0,42806
12	0,11916	0,60633	-0,35263	-0,29865	0,35351	-0,06837	0,18975
13	-0,12739	0,19497	-0,23516	0,09813	0,04258	-0,04736	-0,17928
14	0,28625	-0,09999	-0,22036	-0,42056	-0,20107	-0,03099	-0,28041
15	0,22143	-0,24792	-0,00318	0,40107	-0,38876	0,10772	-0,04632
16	0,25922	-0,11129	0,17064	0,03348	-0,32109	0,28602	0,50346
17	-0,19624	0,07109	0,07823	0,25402	0,07393	-0,31818	0,16471
18	-0,05959	-0,03128	-0,50833	0,43090	0,47131	0,22756	0,17461
19	0,17456	-0,00350	-0,25818	0,11130	-0,07108	-0,51571	0,22351
20	-0,00833	0,41162	-0,51180	0,23183	-0,19357	0,01488	-0,45597
21	-0,14028	0,62896	0,26346	-0,27892	0,16608	0,19602	0,30636
22	-0,24199	-0,01263	0,30516	-0,01046	-0,03512	0,56129	-0,40959
23	0,09169	0,35411	0,12222	0,13811	0,14178	-0,21365	-0,26872
24	-0,24956	-0,21920	-0,11149	-0,44804	-0,25010	-0,17230	-0,30716
25	-0,24196	-0,41449	-0,35463	0,08657	-0,15130	-0,34826	0,24646
26	0,16448	0,03219	-0,08889	-0,29839	0,11576	-0,24685	0,11071
27	-0,25597	0,10441	-0,19793	0,19232	0,44935	-0,02466	0,05172
28	0,56466	0,12665	-0,40407	0,22312	0,02846	-0,20396	-0,17795
29	-0,02104	0,47003	0,01048	0,06418	-0,20961	0,55954	-0,44897
30	0,09073	0,20696	0,06411	-0,18149	-0,38936	-0,10731	0,43420
31	-0,32780	0,25220	0,27500	0,24039	-0,23741	-0,03461	-0,37575
32	0,24784	0,05424	0,34009	0,03522	-0,61849	0,05145	0,07283
33	0,29520	-0,16857	-0,02861	0,26413	-0,38516	0,12684	-0,46425
34	0,24857	-0,59575	-0,34430	0,25907	0,46193	-0,26427	0,08351
35	-0,12420	0,14841	-0,40392	-0,01734	0,09302	0,67546	0,23927
36	-0,58027	0,37585	-0,37985	0,14409	-0,35006	0,13809	0,30201
37	-0,36261	0,36992	0,32029	0,58460	0,01615	-0,42508	0,01004
38	-0,03764	0,26253	-0,21012	-0,19695	-0,15594	-0,60288	-0,35488
39	-0,08316	-0,21968	0,42620	-0,22358	-0,18109	-0,18222	-0,29365
40	-0,12957	0,22269	0,33374	-0,30345	-0,16665	-0,29091	-0,09987