

Perbandingan rate kode konvolusi dan aplikasinya pada cdma

Nanang Kurniawan¹, Yoedy Moegiharto²

¹Mahasiswa Politeknik Elektronika Negeri Surabaya, Jurusan Teknik Telekomunikasi

²Politeknik Elektronika Negeri Surabaya Institut Teknologi Sepuluh Nopember

Kampus ITS, Surabaya 60111

e-mail : selalu.tersenyum@yahoo.com

Abstrak

Untuk mengirimkan suatu informasi dari pengirim ke penerima terjadi beberapa proses terlebih dahulu terhadap informasi tersebut. Salah satunya adalah menggunakan proses CDMA. Dimana data pada saat akan dikirimkan data akan mengalami proses spreading yang kemudian akan di dispreading pada penerimanya. Pada proyek akhir ini akan dicoba bagaimana jika data sebelum mengalami proses spreading akan mendapatkan proses *Convolutional Code*. Dengan demikian dapat diketahui pengaruh dari *Convolutional Code* saat diaplikasikan pada sistem CDMA.

Pada proyek akhir ini digunakan salah dua tipe dari rate convolutional code untuk rate $\frac{1}{2}$ dan $\frac{1}{3}$. Akan dilakukan analisa perbandingan berdasarkan dari kinerja dari rate $\frac{1}{2}$ dan $\frac{1}{3}$ dimana kinerjanya akan di bandingkan berdasarkan nilai BER yang lebih baik menurut nilai E_b/N_0 yang di nyatakan dalam dB.

Dengan membandingkan kode rate $\frac{1}{2}$ dan $\frac{1}{3}$ maka dapat diketahui kinerjanya pada sistem CDMA dimana untuk nilai BER 10^{-3} pada rate $\frac{1}{2}$ membutuhkan E_b/N_0 sebesar sekitar 11 dB sedangkan untuk rate $\frac{1}{3}$ membutuhkan E_b/N_0 sebesar sekitar 10 dB. Sehingga dapat dikatakan untuk rate $\frac{1}{3}$ dapat menghasilkan kinerja yang lebih baik dibandingkan rate $\frac{1}{2}$.

Keyword : BER, Code Convolution, CDMA.

1. Pendahuluan

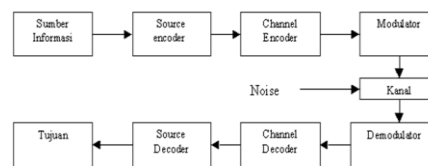
Pada saat ini, sistem komunikasi digital dituntut untuk dapat mentransmisikan data maupun suara dengan kecepatan tinggi dan juga memiliki efisiensi bandwidth yang baik juga, serta memiliki performansi yang

handal. Akan tetapi ada tantangan tersendiri di samping kapasitas *bandwidth* yang terbatas dengan tetap mempertahankan kestabilan BER yang diijinkan oleh sistem tersebut, juga adanya pengaruh *noise* dan *interferensi* terhadap bit-bit yang dikirim sehingga menimbulkan error.

Teknik pengkodean kanal dibutuhkan untuk mendapatkan nilai QOS yang diharapkan. Teknik pengkodean kanal yang baik adalah teknik pengkodean kanal yang menghasilkan BER (*Bit Error Rate*) yang rendah. *Convolutional encoder* adalah salah satu bagian dari teknik pengkodean yang dapat membantu kerja pengkodean kanal agar dapat bekerja maksimal.

2. Teori Penunjang

Untuk mengirimkan suatu informasi dari pengirim ke penerima, dilakukan beberapa proses dahulu terhadap informasi tersebut. Salah satunya adalah proses channel coding. Tahapan proses channel coding dalam sistem transmisi dapat dilihat dalam diagram blok sistem transmisi berikut



Gambar 1. Diagram Blok Sistem Transmisi

Channel coding berfungsi untuk menjaga informasi atau data digital dari error yang mungkin terjadi selama proses transmisi dengan cara menambahkan bit redundansi (tambahan) ke dalam data yang akan

dikirimkan. Channel code yang digunakan untuk mendeteksi error disebut error detection codes, sedangkan yang juga mampu untuk mengoreksi kesalahan tersebut disebut error correction code.

2.1 Kode Konvolusi

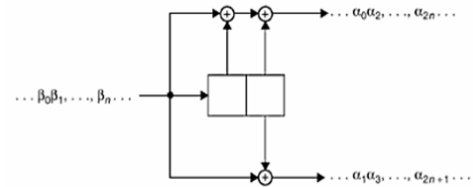
Kode konvolusi adalah jenis kode yang memiliki perbedaan mendasar dari block code dimana urutan bit informasi tidak dikelompokkan dalam blok-blok yang berbeda sebelum dikodekan. Proses yang terjadi adalah bit informasi sebagai masukan secara kontinyu di mapping kedalam urutan bit output encoder.

Proses encoding yang diterapkan melakukan operasi biner yang khusus pada bit-bit masukan, sehingga bit-bit sumber dikatakan di konvolusi untuk menghasilkan deretan bit keluaran. Hal ini juga menyebabkan bit-bit keluaran tidak hanya bergantung pada bit-bit yang di proses, namun juga bergantung pada deretan sebelumnya dari bit-bit sumber dengan demikian membutuhkan suatu bentuk dari memori. Dalam prakteknya bentuk memori ini dapat berupa shift register dengan panjang tertentu yang kita kenal dengan nama constraint length teknik ini mampu meningkatkan coding gain yang lebih besar di bandingkan dengan jika digunakan block coding dengan kompleksitas yang sama.

Pada pengkodean konvolusi ini, tiap codeword selain bergantung pada message yang bersesuaian juga tergantung pada m blok message sebelumnya. Satu set codeword dengan k input, n output dan tingkat memori m disebut dengan kode konvolusi (n,k,m). Rate kode didefinisikan $R = k/n$. Karena pengkodean konvolusi memiliki tingkat memori maka harus diimplementasikan dengan rangkaian logika sekuensial.

Kode konvolusi ini dihasilkan dengan cara melewatkan urutan bit informasi melalui sejumlah tingkat shift register. Pada umumnya shift register terdiri dari $N(k \text{ bit})$ tingkat dan m generator polinomial seperti gambar dibawah. Data masukan digeser sepanjang k bit shift register pada satu kali waktu.

Jumlah bit keluaran untuk tiap k bit masukan adalah n bit, dengan kode rate $R_c = k/n$. Parameter N disebut panjang constraint dan menunjukkan jumlah bit data masukan



Gambar 2. state rate = 1/2 dan memory m = 2

Beberapa istilah dalam convolutional code adalah sebagai berikut:

Generator Matriks

Untuk mempresentasikan code convolutional pada matlab di gunakan matriks generator (G). Matriks generator di definisikan dari n koneksi vektor dilambangkan $g_1, g_2, \dots, g_n \dots (2)$, dimana untuk setiap n-modulo-2 adders. Setiap vektor berdimensi K dan koneksi dari shift register dengan modulo-2 adder.

Contoh ilustrasi untuk (3,1,3) encoder konvolusi

$$g_1 = [1 \ 0 \ 0], g_2 = [1 \ 0 \ 1], g_3 = [1 \ 1 \ 1] \dots (2.1)$$

maka generator matriks G nya untuk kode konvolusi

$$G = [g_1 \ g_2 \ g_3] \dots (2.2)$$

Untuk mendapatkan codeword U setiap register iterasi i diletakan pada row vektor S_i maka generator matriks G,

$$U_i = S_i \cdot G \dots (2.3)$$

Maka pada contoh untuk (3,1,3) kode konvolusi register 1 0 0 untuk iterasi $i = 1$. codeword di tuliskan

$$U_1 = [1 \ 0 \ 0][G] = [1 \ 1 \ 1] \dots (2.4)$$

Generator Polinomial

Berfungsi untuk membangkitkan polinomial sesuai dengan urutan bit data masukan.

Logic Table

Tabel kebenaran berfungsi untuk menunjukkan keluaran dari encoder jika diberi urutan bit masukan

State Diagram

State diagram digunakan untuk mempresentasikan proses pengkodean yang berbentuk diagram sederhana. Diagram ini akan menunjukkan kemungkinan keadaan dan transisinya dari satu keadaan ke keadaan lainnya.

Trellis Diagram

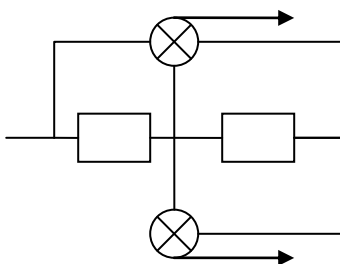
Merupakan bentuk penyederhanaan dari diagram pohon yang merupakan representasi dari keluaran

Tree Diagram

Menunjukkan struktur encoder dalam bentuk diagram pohon dengan cabang-cabangnya menunjukkan variasi keadaan dan keluaran yang mungkin terjadi

Algoritma pembentukan kode konvolusi

Algoritma pembentukan kode konvolusi dapat menggunakan operasi diagram state. Pada kode konvolusi (n,k,m) tiap k bit data yang masuk pada satu satuan waktu, keluarannya menjadi n bit pada satu satuan waktu. Pembentukan encoder kode konvolusi diperlihatkan oleh Gambar 4



Gambar 3. gambar convolutional encoder $(2,1,2)$

Dengan mengacu pada pengkodean kode konvolusi $(2,1,2)$ Gambar 4, proses dimulai dengan membuat state mula dan mark code

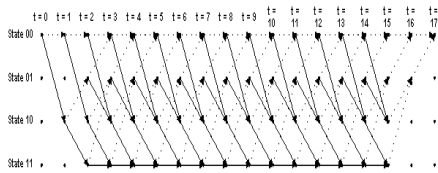
dengan nilai 00 kemudian diberi inputan 0 sehingga output dari encoder tersebut adalah 00 dan state nya berubah menjadi 00, saat state 00 dan input bernilai 1 maka output dari encoder tersebut adalah 10 dan state dari encoder tersebut berubah menjadi 10. Saat encoder tersebut berada pada state 01 dan input bernilai 0 maka output dari encoder tersebut adalah 11 dan state dari encoder tersebut menjadi 00. Kemudian saat state dari encoder tersebut adalah 01 dan di beri input 1 maka output dari encoder tersebut adalah 01 dan state dari encoder tersebut berubah menjadi 10 begitu selanjutnya saat berada pada state 10 dan 11 saat diberi input 0 dan 1 maka akan muncul output dan juga state selanjutnya yang terdapat pada table 1. Tabel 1 memperlihatkan tabel dari state kode konvolusi $(2,1,2)$

State Kini	Input	
	0	1
	Output/State Transisi	Output/State Transisi
00	00/00	10/10
01	11/00	01/10
10	11/01	01/11
11	00/01	10/11

Tabel 1. State dari kode konvolusi $(2,1,2)$

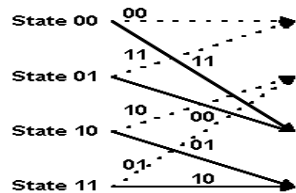
2.2 Algoritma Viterbi

Salah satu teknik *forward error correction* untuk kode konvolusi adalah algoritma viterbi. *Convolutional code* dengan viterbi decoder adalah sebuah FEC teknik yang sesuai dengan sebuah sinyal yang berada dalam saluran additive white gaussian noise (AWGN). Cara kerja dari algoritma viterbi sendiri adalah dengan mencari jarak minimum terkecil yang memiliki nilai akumulasi error matrix yang terkecil. Algoritma viterbi biasanya menggunakan trellis diagram untuk penggambarannya.



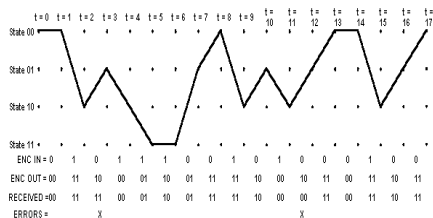
Gambar 4. gambar trellis diagram untuk convolutional encoder (2.1.3)

Untuk mendapatkan nilai akumulasi error matrix yang terkecil dari output encoder kode konvolusi diperlukan data tentang state output dari enkoder kode konvolusi tersebut . Gambar 5.2 akan menggambarkan tentang hubungan state selanjutnya dan output pada encoder kode konvolusi jika diberi input bernilai 1 atau jika input bernilai 0. Pada gambar tersebut jika input bernilai 1 maka akan digambarkan dengan garis lurus dan tebal, jika input bernilai 0 maka digambarkan dengan garis putus-putus.



Gambar 5 hubungan input,output,state dan next state pada encoder kode konvolusi

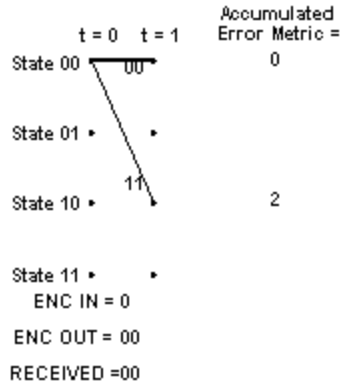
Dengan mengetahui hubungan antara input,output , state dan next state maka akan mengerti bagaimana mencari akumulasi error yang terkecil yang sesuai dengan encodernya.



Gambar 6. hasil algoritma viterbi berdasarkan akumulasi error matrik terkecil

Setiap kali kami menerima sepasang saluran simbol, kita akan menghitung

metrik untuk mengukur "jarak" antara apa yang kita terima dan semua saluran yang mungkin simbol pasangan kita bisa diterima Dimulai dari $t = 0$ sampai $t = 1$, hanya ada dua kemungkinan simbol saluran pasangan kita bisa diterima: 00_2 dan 11_2 Itu karena convolutional encoder itu diinisialisasi ke nol semua state, dan diberi satu bit input = satu atau nol, hanya ada dua state kita bisa transisi dan dua kemungkinan output dari encoder. Mungkin output encoder adalah 00_2 dan 11_2 Metrik yang akan kita gunakan untuk sekarang adalah jarak Hamming antara saluran yang diterima pasangan simbol dan lambang saluran yang mungkin pasang. Hamming jarak yang dihitung dengan menghitung berapa banyak bit yang diterima berbeda antara simbol saluran pasangan dan kemungkinan saluran simbol pasangan Hasil hanya dapat nol, satu, atau dua. The Hamming (atau metrik) nilai-nilai kita menghitung pada setiap kali instan untuk jalur antara state-state pada waktu sebelumnya instan dan state-state pada waktu saat ini disebut cabang instan metrik Untuk pertama kalinya instan, kita akan menyimpan hasil ini sebagai "akumulasi kesalahan metrik" nilai-nilai, berhubungan dengan state Untuk kedua kalinya instan pada, akumulasi kesalahan metrik akan dihitung dengan menambahkan kesalahan akumulasi sebelumnya metrik untuk cabang saat metrik. Pada $t = 1$, kami menerima 00_2 . Satu-satunya saluran mungkin simbol pasangan kita bisa terima adalah 00_2 dan 11_2 The Hamming jarak antara 00_2 dan 00_2 adalah nol. The Hamming 00_2 jarak antara 00_2 dan 11_2 adalah dua Oleh karena itu, nilai metrik cabang cabang dari 00_2 state 00_2 adalah nol, dan untuk cabang dari 00_2 state 10_2 itu dua Karena akumulasi sebelumnya nilai-nilai metrik kesalahan sama dengan nol, akumulasi nilai-nilai metrik 00_2 state 00_2 dan state 10_2 adalah sama dengan nilai-nilai metrik cabang. Kesalahan Akumulasi nilai metrik untuk dua state lainnya yang belum terdefinisi. Angka di bawah ini menggambarkan hasil pada $t = 1$



Perhatikan bahwa garis-garis antara state-state padat pada $t = 1$ dan state pada $t = 0$ menggambarkan penerus pendahuluan antara state pada $t = 1$ dan state pada $t = 0$ masing-masing Informasi ini ditampilkan secara grafis pada gambar, tetapi disimpan secara numerik dalam pelaksanaannya Untuk lebih spesifik, atau mungkin jelas adalah kata yang lebih baik, pada setiap waktu instan t , kita akan menyimpan jumlah negara pendahulu yang menuju ke masing-masing state yang aktif di t .

2.3 CDMA

Code Division Multiple Access (CDMA) adalah sebuah bentuk pemultipleksan (bukan sebuah skema pemodulasian) dan sebuah metode akses secara bersama yang membagi kanal tidak berdasarkan waktu (seperti pada TDMA) atau frekuensi (seperti pada FDMA), namun dengan cara mengkodekan data dengan sebuah kode khusus yang diasosiasikan dengan tiap kanal yang ada dan menggunakan sifat-sifat interferensi konstruktif dari kode-kode khusus itu untuk melakukan pemultipleksan. CDMA juga mengacu pada sistem telepon seluler digital yang menggunakan skema akses secara bersama ini, seperti yang diprakarsai oleh Qualcomm. CDMA adalah sebuah teknologi militer yang digunakan pertama kali pada Perang Dunia II oleh sekutu Inggris untuk menggagalkan usaha Jerman mengganggu transmisi mereka. Sekutu

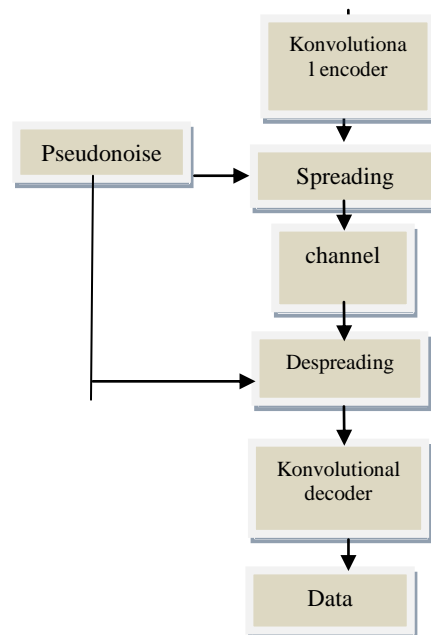
memutuskan untuk mentransmisikan tidak hanya pada satu frekuensi, namun pada beberapa frekuensi, menyulitkan Jerman untuk menangkap sinyal yang lengkap.

Pada CDMA seluruh pelanggan menggunakan spektrum frekuensi yang sama namun dibedakan pada tiap kode uniknya yang membuat kerahasiaan informasi tiap pelanggannya terjamin. Pembangkitan sinyal pada CDMA dilakukan dengan mengalihkan suatu sinyal informasi dengan suatu pembawa pseudorandom noise sequence yang mengakibatkan sistem ini mempunyai rata-rata transmisi yang jauh lebih besar dari keperluan bandwidth minimum.

3. METODOLOGI

Perancangan Sistem

a



Gambar 7. Blok diagram sistem

Dalam sistem ini sinyal setelah dibangkitkan secara random, data akan masuk kedalam encoder kode konvolusi dengan dua macam rate yaitu $\frac{1}{2}$ dan $\frac{1}{3}$. Lalu data akan di spreading oleh Pseudonoise code yaitu gold. Sebelum

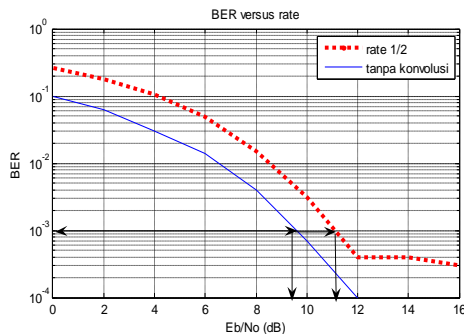
sinyal ditransmisikan melalui kanal sinyal harus dimodulasikan terlebih dahulu. Setelah menjadi sinyal termodulasi data akan dikirimkan melalui kanal dengan nosie AWGN. Dimana nosie AWGN adalah noise yang disebabkan oleh panas. Sehingga data yang dikirimkan terdapat perubahan data yang dikirim atau dengan kata lain data menjadi error

Pada sisi penerima akan terjadi proses sebaliknya yaitu data akan diterima oleh demodulasi kemudian data akan di desreading dengan PN code yang sama . Setelah itu data akan dilewatkan ke decoder kode konvolusi untuk mendapatkan data asli

4.Data Hasil Pengukuran

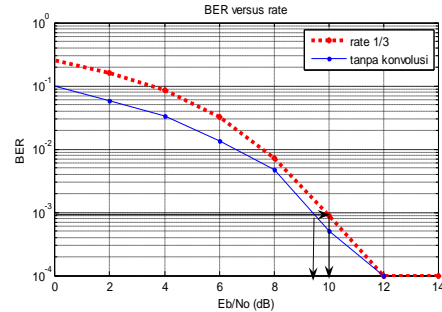
4.1.1 HASIL PROSES CDMA DENGAN MENGGUNAKAN RATE 1/2

Gambar di bawah 4.1 menunjukkan perbandingan antara hasil BER yang dihasilkan oleh proses CDMA dengan menggunakan proses Convolutional code dengan rate $\frac{1}{2}$ yang di gambarkan oleh garis merah dibandingkan dengan BER yang dihasilkan apabila hanya melewati kanal AWGN saja



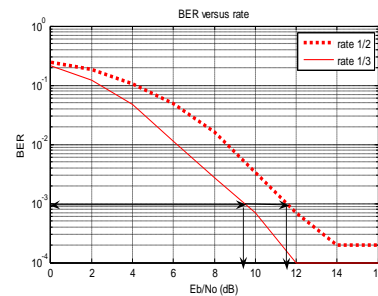
Gambar 8. perbandingan BER pada CDMA dengan menggunakan kanal AWGN dan dengan melewati rate 1/2

4.1.2 HASIL PROSES CDMA DENGAN MENGGUNAKAN RATE 1/3



Gambar 9. perbandingan BER pada CDMA dengan menggunakan kanal AWGN dan dengan melewati rate 1/2

4.1.3 HASIL PERBANDINGAN RATE 1/2 DAN 1/3



Gambar 10. perbandingan BER pada rate 1/2 dan rate 1/3

4.2 HASIL DAN ANALISA DATA

Dari hasil simulasi dua grafik diatas di tunjukan bahwa garis warna merah adalah bila menggunakan rate $\frac{1}{2}$ pada gambar pertama dan rate $\frac{1}{3}$ pada gambar grafik kedua dan pada kanal AWGN di tunjukan dengan garis berwarna biru

Dimana pada gambar grafik pertama di tunjukan bahwa pada kanal AWGN untuk mencapai nilai BER 10^{-3} di perlukan Eb/No sekitar 10dB sedangkan jika di bandingkan apabila data di masukkan kedalam kode konvolusi dahulu dengan rate $\frac{1}{2}$ ternyata untuk mencapai BER 10^{-3} diperlukan Eb/No lebih dari 12dB. Begitu juga pada gambar grafik kedua untuk mencapai BER 10^{-3} diperlukan Eb/No sekitar 10dB jika menggunakan AWGN sedangkan jika menggunakan kode konvolusi dengan rate $\frac{1}{3}$ untuk mencapai

BER 10^{-3} diperlukan nilai Eb/No sekitar 11dB.

Berdasarkan hasil dari analisa dua gambar grafik diatas dapat dikatakan bahwa apabila data hanya melewati kanal AWGN saja membutuhkan nilai Eb/No yang lebih kecil dibandingkan jika melewati proses Convolutional Code hal ini dikarenakan pada bila melewati kanal AWGN kemungkinan terjadi nya error pada proses pengiriman data lebih kecil dibandingkan jika melewati proses Convolutional Code. Dan terlihat juga bahwa rate $\frac{1}{2}$ dan rate $\frac{1}{3}$ hasil yang di berikan lebih baik apabila menggunakan rate $\frac{1}{3}$ karena dengan rate $\frac{1}{3}$ dapat menghasilkan nilai BER yang lebih baik sehingga SNR yang di hasilkan juga dapat dikatakan lebih baik.

5. KESIMPULAN

Berdasarkan hasil penelitian dan analisa yang telah dilakukan , maka dapat di berikan beberapa kesimpulan:

1. Proses CDMA dengan menggunakan kanal AWGN saja menghasilkan nilai BER yang lebih bagus dibandingkan jika terlebih dahulu di lewatkan pada *Convolutional Code*
2. *Convolutional Code* dengan menggunakan rate $\frac{1}{3}$ menghasilkan nilai BER yang lebih baik di bandingkan jika menggunakan rate $\frac{1}{2}$

Daftar Pustaka :

- [1] Andrew J. Viterbi. (1995) CDMA : Principles of Spread Spectrum Communication (1st edition) Prentice Hall PTR
- [2] Man Young Rhee, “ Error Corecting Coding Theory ”, McGraw-Hill Inc, 1989.
- [3] <http://www> Tutorial on Convolutional Coding with Viterbi Decoding
- [4] Peebles, Peyton. Z, “ Probability, Random Variables, and Random Signal Principles“, McGraw-Hill, Inc, 1993.
- [5] Shu Lin and Costello, Daniel. J.” Error Control Coding : Fundamental and Aplication“, Prentice – Hall Inc, Englewood Cliffs, New Jersey, 1983.
- [6] Simon, Marvin. K, Hinedi, Sami. M and Lindsey. William. C, Digital Communication Techniques Signal Design and Detection “, Prentice - Hall International, Inc, Englewood Cliffs, New Jersey, 1995.
- [7] Chen Wei , RTL Implematation of Viterbi Decoder, Linköpings Universitet, 2006