

BAB I PENDAHULUAN

1.1 Latar Belakang

Migrasi sistem penyiaran yang saat ini masih menggunakan sistem analog, menuju ke sistem digital sudah merupakan suatu keharusan. Sistem analog mempunyai banyak kelemahan diantaranya ketahanan sinyal terhadap gangguan (*noise*) yang tidak terlalu baik serta tidak efisien terhadap spektrum frekuensi, dimana pada sistem analog membutuhkan *frequency guard* antar pemancar yang beroperasi agar tidak saling mengganggu.

Sedangkan pada sistem digital, sinyal lebih tahan terhadap gangguan (*noise*) serta lebih efisien terhadap spektrum frekuensi. Karena dengan sistem digital, beberapa pemancar dapat menggunakan sebuah spektrum frekuensi bersama-sama, yang disebut dengan jaringan SFN (*Single Frequency Network*). Selain itu kelebihan sistem digital adalah mempunyai kualitas gambar lebih baik yang diterima oleh pengguna, baik pengguna tetap (*fixed user*) maupun pengguna bergerak (*mobile user*) dan pada sistem digital juga menawarkan program siaran yang banyak dan variatif.

Dalam penerapan sistem digital di Indonesia, maka diperlukan adanya suatu penelitian untuk dapat menentukan parameter-parameter sistem yang sesuai dengan kondisi lingkungan di Indonesia. Sebab setiap daerah di Indonesia mempunyai karakteristik kanal yang berbeda-beda. Maka dalam skripsi ini akan dibahas mengenai evaluasi kinerja TV digital yaitu dengan melakukan pengukuran *path loss*, kualitas cakupan dan *delay spread* yang hasilnya kemudian dibandingkan dengan hasil pemodelan kemudian dianalisis untuk mengetahui kualitas kinerja dari TV digital pada lingkungan *urban* di Jakarta. Skripsi ini juga berkaitan dengan skripsi yang berjudul "***Standardisasi Parameter Set-Top Box DVB-T untuk Implementasi TV Digital di Indonesia***". Yang membahas tentang standar parameter penerima siaran DVB-T dan menetapkan spesifikasi teknis untuk standarisasi set-top-box. Dan berkaitan pula dengan skripsi yang berjudul "***Karakteristik Propagasi Radio Untuk Sistem Penyiaran TV Digital Terrestrial Pada Lingkungan Urban***". Yang membahas tentang propagasi radio penyiaran televisi digital.

1.2 Rumusan Masalah

Dengan latar belakang seperti yang telah diuraikan, maka rumusan masalah penelitian ditekankan pada:

1. Bagaimana mengkaji kualitas penyiaran system DVB-T dilihat dari gambar dan suara, pada kondisi *Line of Sight* (LOS) dan *Non- Line of Sight* (NLOS)?
2. Bagaimana pengaruh penggunaan *guard interval* 1/16 dan 1/32 dengan nilai *code rate* $\frac{1}{2}$ dan $\frac{3}{4}$?
3. Berapa jarak jangkauan penerimaan sistem DVB-T ?

1.3 Batasan Masalah

Berdasarkan perumusan masalah di atas maka pembahasan dibatasi pada :

- a. Pengukuran kinerja dari TV digital Terrestrial dilakukan pada lingkungan *urban* di Jakarta dengan mengukur *path loss*, kualitas cakupan dan *delay spread*.
- b. Parameter sistem DVB-T yang digunakan adalah Modulasi 16 QAM, *code rate* $\frac{1}{2}$ dan $\frac{3}{4}$, *Guard Interval* 1/16 dan 1/32, Mode 8k (durasi simbol 896 μ s), Standar Video menggunakan MPEG-2.
- c. Frekuensi kerja sistem DVB-T menggunakan 578 MHz.
- d. Pada pemodelan sistem dilakukan dengan model *baseband*, dan tidak mensimulasikan *pilot carrier*.
- e. Menggunakan *software* matlab 7.0.1 dalam pemodelan sistem.

1.4 Tujuan

Tujuan dari penulisan skripsi ini adalah untuk melakukan analisis dari data hasil pengukuran dan pemodelan sistem untuk mengetahui serta menguji kinerja sistem pada lingkungan *urban* di Jakarta. Dan mengusulkan suatu rekomendasi yang dapat dijadikan standar untuk implementasi sistem TV digital dan *set top box*.

1.5 Sistematika Penulisan

Bab I. Pendahuluan

Pada bab ini akan dibahas latar belakang, rumusan masalah, pembatasan masalah, tujuan, dan sistematika penulisan.

Bab II. Teori Penunjang

Pada bab ini akan dibahas mengenai teori-teori yang terkait konsep dasar televisi digital, penyiaran televisi digital tersterial DVB-T, pengkodean dengan *reed-solomon* dan sistem modulasi dengan OFDM.

Bab III. Metodologi Pengukuran dan Pemodelan Sistem

Pada bab ini akan dibahas mengenai jenis alat ukur dan pengukuran yang akan dijadikan dasar analisis serta pemodelan sistem penyiaran televisi digital teresterial (DVB-T).

Bab IV. Pembahasan

Pada bab ini akan dibahas mengenai analisis data hasil pengukuran dan pemodelan sistem pada DVB-T.

Bab V. Penutup

Pada bab ini akan dibahas mengenai kesimpulan dari keseluruhan isi skripsi yang menjawab mengenai rumusan masalah, selain itu juga memuat saran yang bisa digunakan untuk mengembangkan sistem pertelevisian di Indonesia demi kenyamanan masyarakat.

BAB II TINJAUAN PUSTAKA

Penyiaran TV digital secara umum dapat didefinisikan sebagai pengambilan atau penyimpanan gambar dan suara secara digital, yang kemudian ditransmisikan juga secara digital, dan diterima pada sisi pengguna dalam format digital.

Pada awalnya, sistem penyiaran TV digital yang dilakukan di negara-negara maju dan di Indonesia dilakukan melalui media komunikasi satelit dan kabel. Perkembangan teknologi, terutama yang dilakukan di Eropa dan Jepang telah berhasil menemukan metode untuk melakukan penyiaran TV teresterial yang berarti sistem transmisi melalui gelombang radio dari darat ke darat.

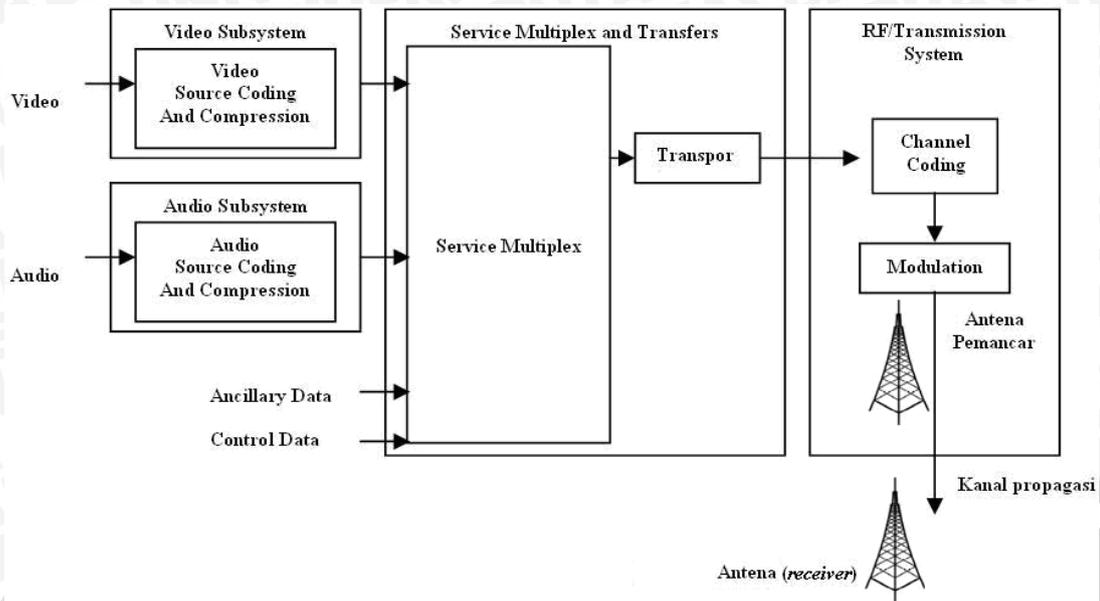
Indonesia saat ini sedang dalam proses migrasi sistem penyiaran TV dari sistem analog yang saat ini dipergunakan menuju ke sistem digital. Migrasi tersebut dilakukan dengan dasar bahwa sistem digital memiliki kinerja yang lebih baik dibandingkan dengan sistem analog. Sistem TV digital menawarkan kualitas gambar yang lebih baik yang diterima oleh pengguna, baik pengguna tetap (*fixed user*) maupun pengguna bergerak (*mobile user*). Selain itu sistem digital juga menawarkan program siaran yang lebih banyak dan variatif, sebab sistem digital dapat mengefisienkan *bandwidth* yang ada.

2.1 Standar Teknologi Televisi Digital Teresterial

Dalam sistem penyiaran TV digital, terdapat 2 bagian standarisasi yang digunakan, yaitu bagian *source coding* dan *multiplexing* serta bagian *channel coding* dan *transmission*. Untuk standarisasi yang pertama, yaitu standarisasi *source coding* dan *multiplexing*, sebagian besar menggunakan standar *Moving Picture Experts Group-2* (MPEG-2). Sedangkan standar yang kedua, yaitu bagian *channel coding* dan *transmission*, terdapat sejumlah standar TV digital yang berkembang pesat saat ini. Standar tersebut yaitu:

- *Advanced Television System Committee* (ATSC) yang berkembang di Amerika Serikat (USA)
- *Digital Video Broadcasting Terrestrial* (DVB-T) yang berkembang di Eropa
- *Integrated Service Digital Broadcasting* (ISDB-T) dari Jepang
- *Terrestrial Digital Multimedia Broadcasting* (T-DMB) dari Korea
- *Digital Multimedia Broadcasting Terrestrial* (DMB-T)

Gambar 2.1 Berikut menunjukkan skema sistem transmisi siaran TV digital.



Gambar 2.1 Sistem Transmisi Siaran TV Digital Terrestrial

Sumber : <http://isotc.iso.org/livelink/livelink/fetch/...>

Sistem ATSC menggunakan teknik modulasi VSB (*Vestigial Side band*). Dari berbagai riset komparatif, sistem ATSC lebih sesuai untuk penerima TV yang tidak bergerak (*fixed*). Memang sejak awal sistem ini didesain untuk menghantarkan sinyal HDTV. Sedangkan DVB-T dan ISDB-T menggunakan teknik OFDM (*Orthogonal Frequency Division Multiplexing*), yang dapat memberikan kekebalan atau imunitas terhadap efek yang disebabkan oleh lintasan jamak (*multipath*) serta juga dapat menjangkau pengguna bergerak (*mobile receiver*).

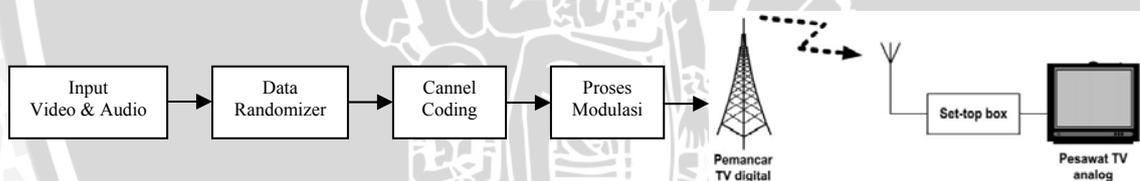
Standar yang menggunakan OFDM memiliki karakteristik yang berbeda-beda, berdasarkan penggunaan sistem pengkodean kanal (*channel coding*) yang digunakan. *Channel coding* tersebut juga menentukan *payload* (kapasitas kanal) sistem.

Standar-standar diatas dan beberapa variannya telah diadopsi oleh beberapa negara. Seperti DVB yang diadopsi oleh seluruh negara di Eropa serta Australia dan beberapa negara di Asia. ATSC di adopsi oleh Amerika Utara, sejumlah negara di Amerika Selatan dan beberapa negara di Asia.

Untuk sistem penerimaan TV digital, biasa dibedakan menjadi 2 macam, yaitu siaran TV SDTV (*Standart Definition Tele-Vision*) dan HDTV (*High Definition Tele-Vision*). Siaran SDTV merupakan siaran TV digital yang dapat diterima menggunakan pesawat televisi biasa yang telah dilengkapi oleh *set top box*. Sedangkan HDTV merupakan siaran televisi digital yang hanya dapat diterima dengan menggunakan pesawat penerima khusus, yang telah dilengkapi oleh *set top box* di dalamnya. HDTV mempunyai kualitas gambar yang lebih baik dibandingkan dengan SDTV.

2.2 Sistem *Digital Video Broadcast Terrestrial (DVB-T)*

DVB-T adalah suatu sistem penyiaran televisi digital yang dilakukan dari darat ke darat, tanpa menggunakan satelit. Sistem penyiaran teresterial memiliki kanal propagasi yang rentan akan gangguan, terutama yang disebabkan oleh efek lintas jamak (*multipath*). Oleh karena itu, DVB-T menggunakan sistem OFDM (*Orthogonal Frequency Division Multiplexing*) yang tahan terhadap gangguan tersebut. Selain itu, DVB-T juga menggunakan sistem *concatenated code*, yaitu penggabungan 2 jenis pengkodean kanal.



Gambar 2.2 Model Propagasi Sistem Penyiaran TV Digital Teresterial

Sumber: Teknologi Komunikasi Nirkabel, Gamantyo Hendranto, 2008, hal 15

Sistem DVB-T menggunakan 2 macam pengkodean yang dapat dibedakan menjadi pengkodean luar (*outer coding*) dan pengkodean dalam (*inner coding*). Selain itu, DVB-T juga dilengkapi dengan *outer interleaver* dan *inner interleaver*. *Outer coding* dan *outer interleaver* juga digunakan pada sistem DVB-C (*Digital Video Broadcast Cable*) dan DVB-S (*Digital Video Broadcast Satellite*).

DVB-T memiliki 2 jenis mode operasi yang dibedakan berdasarkan dengan jumlah *subcarrier* yang digunakan. Terdapat mode 2k dan mode 8k. Panjang durasi simbol mode 8k adalah 896 μ s dan 224 μ s untuk mode 2k. Mode 8k

memiliki 6817 *subcarrier*, 6048 diantaranya digunakan untuk memodulasi tiap simbol secara independen, sedangkan *subcarrier* yang lain digunakan untuk persinyalan (*signalling*) dan sinkronisasi. Sedangkan mode 2k memiliki total 1705 *subcarrier*, dan seperti pada mode 8k, hanya 1512 *subcarrier* yang berfungsi memodulasi simbol, *subcarrier* yang lain dipergunakan untuk proses sinkronisasi dan persinyalan (*signalling*).

Tabel 2.1 dibawah berikut akan memberikan penjelasan mengenai sistem 2k dan 8k tersebut.

Tabel 2.1 Parameter OFDM untuk Mode 8k dan 2k

Parameter	Mode 8k	Mode 2k
Banyaknya carrier	6817	1705
K min	0	0
K max	6816	1704
Durasi Tu	896 μ s	224 μ s

Sumber: Euopian Standart Telecommunication Institute (ESTI)
TR 101 190 v 1.2.1, Hal 13

Untuk menghindari interferensi sinyal akibat efek lintas jamak (*multipath*), sistem DVB-T mengaplikasikan *guard interval* yang disisipkan pada setiap simbol OFDM. Panjang dari *guard interval* bergantung pada area kerja sistem. Penggunaan *guard interval* juga harus dipertimbangkan. Sebab memperbesar nilai *guard interval* tanpa memperpanjang durasi sinyal secara keseluruhan akan mengakibatkan penurunan kapasitas nilai *bit rate*. Sedangkan memperbesar nilai *guard interval* serta durasi sinyal secara keseluruhan tidak akan mempengaruhi kapaitas kanal, tetapi hal tersebut akan mengakibatkan pemrosesan sinyal akan lebih sulit karena banyaknya jumlah *carrier* yang terjadi akibat durasi simbol yang diperbesar. Panjang *guard interval* pada sistem 2k dan 8k dapat dilihat pada tabel 2.2 dibawah.

Tabel 2.2 Nilai *Guard Interval* Mode 8k dan 2k

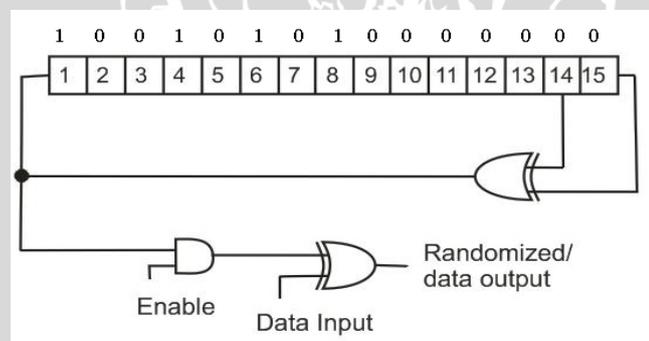
Panjang Interval	Panjang <i>guard interval</i>	
	8k mode	2k mode
1/4	224 μ s	56 μ s
1/8	112 μ s	28 μ s
1/16	56 μ s	14 μ s
1/32	28 μ s	7 μ s

Sumber: Euopian Standart Telecommunication Institute (ESTI)
TR 101 190 v 1.2.1, Hal 18

2.3 Data Randomizer

Pada sistem DVB-T, dilakukan pengacakan data pada sisi pemancar. Pengacakan data dilakukan oleh suatu generator *pseudorandom* untuk menghasilkan output berupa deretan biner yang memiliki karakter *pseudorandom*. Sinyal *pseudorandom* bukan merupakan sinyal random yang sebenarnya, yaitu sinyal yang tidak dapat diprediksikan. Namun sinyal *pseudorandom* bersifat deterministik, yaitu sinyal periodik yang diketahui oleh *transmitter* dan *receiver*. Sinyal ini juga disebut *pseudonoise* atau *pseudorandom* karena meskipun termasuk sinyal deterministik, sinyal ini memiliki karakter statistik seperti *white noise*. Sehingga biasanya dianggap sebagai sinyal random.

Pengacakan data oleh *data randomizer* dilakukan untuk menjamin tidak terjadinya perubahan nilai bit-bit secara tajam dari informasi yang berasal dari frame-frame video sehingga sistem digital dapat memproses frame video tersebut. Pengacakan data juga dilakukan sebagai salah satu bentuk peningkatan kinerja sistem, yaitu dengan mencegah terjadinya kerusakan data yang parah apabila terjadi *error burst*. Pada sistem DVB-T, proses *randomization* dilakukan oleh *Pseudo Random Binary Sequence* (PRBS) yang bergenerator polinomial $1+X^{14}+X^{15}$, seperti terlihat pada Gambar 2.3 dibawah ini:



Gambar 2.3 Proses Randomisasi

Sumber: Ibid, 2001 hal 26

2.4 Pengkodean Kanal

Pengkodean kanal atau *cannel coding* merupakan pengkodean data digital yang berguna untuk melindungi data dari *error* dengan memberikan bit-bit *redundancy* atau bit-bit tambahan pada data yang akan dikirim sehingga dapat meningkatkan kinerja sistem. Pengkodean kanal yang digunakan untuk mendeteksi *error* disebut *error detection code* (kode pendeteksi eror),

sedangkan kode yang mampu mendeteksi dan mengoreksi *error* disebut *error correction code* (kode pengoreksi *error*).

Pengkodean kanal bekerja pada data digital dengan mengkodekan sumber informasi ke dalam urutan kode untuk ditransmisikan melalui kanal. Pada dasarnya ada tiga jenis pengkodean untuk *error correction* dan *error detection* yaitu *block codes* (kode blok), *convolutional codes* (kode konvolusional), dan *turbo codes* (kode turbo).

Sistem DVB-T menggunakan 2 buah jenis *coding* pada sistem *channel coding*-nya, yaitu reed-solomon sebagai *outer code* dan kode konvolusional sebagai *inner code*.

2.4.1 Reed-solomon Coding

Sistem DVB-T menggunakan reed-solomon sebagai *outer coder* atau kode terluar dari sistem. Kode reed-solomon termasuk kode blok untuk mengoreksi *error*. Kode blok merupakan kode yang mampu mendeteksi dan mengoreksi *error* suatu blok data dalam jumlah tertentu. Kode blok dapat digunakan untuk meningkatkan kinerja sistem komunikasi ketika alternatif yang lain, misalnya peralatan yang tidak mungkin untuk diterapkan, seperti penambah daya pancar atau penggunaan demodulator yang kompleks.

Kode *reed-solomon* (RS) adalah *subclass* dari kode *Bose-Chadhuri-Hocquenghem* (BCH) non-biner. Pada kode RS, setiap blok dipandang sebagai *k* simbol yang setiap simbolnya mempunyai *m* bit. Kode RS dapat ditulis dengan persamaan berikut:

$$RS(n, k) = RS(2^m - 1, 2^m - 1 - 2f) \quad (2.1)$$

Yang berarti untuk *k* simbol, dimana dalam 1 simbol terdapat *m* bit data biner, akan dikodekan menjadi *n* simbol, dan kemampuan koreksinya seperti pada persamaan berikut:

$$f = \frac{(n - k)}{2} \quad (2.2)$$

2.4.1.1 Encoder Reed-Solomon

Bentuk umum dari kode RS (*Reed-solomon*) dapat ditulis dengan persamaan berikut:

$$(n, k) = (2^m - 1, 2^m - 1 - (2f)) \tag{2.3}$$

Dimana $n-k=2t$ adalah banyaknya simbol *parity* dan t merupakan kemampuan untuk mengoreksi simbol *error*. Sedangkan bentuk generator polinomial dari RS dapat ditulis dengan persamaan berikut:

$$g(x) = g_0 + g_1 + g_2x^2 + \dots + g_{2t-1}x^{2t-1} + g_{2t}x^{2t} \tag{2.4}$$

Derajat dari generator polinomial sama dengan banyaknya simbol *parity* yang ditambahkan yaitu $2t$. Sehingga akar dari generator polinomial, yang ditambahkan dengan a , mempunyai derajat yang sama. Akar-akar dari $g(x)$ dapat dituliskan sebagai $a, a^2, a^3, \dots, a^{2t}$. Sebagai contoh, kode RS (7,3) dengan $n = 7$ dan $k = 3$, mempunyai kemampuan untuk mengoreksi simbol *error* $2t=n-k = 4$, maka kode tersebut mempunyai 4 akar. Generator polinomialnya dapat ditulis sebagai persamaan berikut:

$$\begin{aligned} g(x) &= (x - a)(x - a^2)(x - a^3)(x - a^4) \\ &= (x^2 - (a - a^2)x + a^2)(x^2 - (a^3 + a^4)x - a^7) \\ &= (x^2 - a^4x + a^3)(x^2 - a^6x + a^7) \\ &= x^4 - (a^4 - a^3)x^3 + (a^3 + a^{10} + a^6)x^2 - (a^4 - a^9)x + a^3 \\ &= x^4 - a^3x^3 + a^0x^2 - a^1x + a^4 \end{aligned} \tag{2.5}$$

Berdasarkan derajat polinomialnya dari rendah ke tinggi dan dalam *field* biner $+1 = -1$, maka generator $g(x)$ dapat dituliskan pada persamaan berikut:

$$g(x) = a^4 + a^1x + a^0x^2 + a^3x^3 + x^4 \tag{2.6}$$

Untuk pembentukan polinomial *codeword* dilakukan dengan persamaan berikut:

$$U(x) = p(x) + x^{n-k}m(x) \tag{2.7}$$

Dimana $U(x) = \text{codeword}$ yang dibentuk

$m(x) =$ simbol informasi yang akan dikodekan

$p(x) =$ simbol *parity* yang didapat dari sisa pembagian antara perkalian simbol informasi yang akan dikodekan dan polinomial x^{n-k} dengan generator polinomial $g(x)$ yang secara matematis $p(x)$ dapat dituliskan dengan persamaan dibawah.

$$p(x) = x^{n-k}m(x) \text{ mod } g(x) \tag{2.8}$$



2.4.1.2 Dekoder Reed-Solomon

Pada dekoder reed-solomon, *codeword* yang diterima mempunyai persamaan berikut:

$$r(x) = U(x) + e(x) \quad (2.9)$$

Dimana:

$r(x)$ = *codeword* yang diterima

$U(x)$ = *codeword* yang dikirimkan

$e(x)$ = *error*

$r(x)$ akan bernilai sama dengan $U(x)$ pada saat *codeword* yang diterima tidak mengalami *error* ($e(x)=0$).

Langkah pertama yang dilakukan dekoder RS untuk mengkodekan data yang diterima, yaitu menghitung sindrom (S). Sindrom didapat dengan cara memasukkan akar-akar generator polinomialnya ke dalam *codeword* yang diterima. Perhitungan sindrom dapat dituliskan seperti persamaan berikut:

$$S_i = r(a^i) = r(a^i) \quad (2.10)$$

Dimana $i = 1, 2, \dots, n-k$

Jika sindrom yang dihasilkan sama dengan nol maka *codeword* yang diterima tidak mengalami *error*, sedangkan apabila tidak sama dengan nol, maka dapat diartikan bahwa *codeword* yang diterima mengalami *error*, sehingga akan dilakukan langkah berikutnya yaitu:

a) Mencari lokasi *error*

Lokasi *error* yang dilambangkan dengan $\sigma(x)$ mempunyai persamaan *polinomial error locator* seperti persamaan (2.11)

$$\sigma(x) = 1 + \sigma_1 x + \sigma_2 x^2 + \dots + \sigma_v x^v \quad (2.11)$$

Nilai v menunjukkan *error* yang terjadi sampai pada lokasi ke x^v . Nilai-nilai dari $\sigma_1, \sigma_2, \dots, \sigma^v$ dapat dicari menggunakan matriks seperti yang terlihat pada persamaan (2.12), dimana sindrom pertama dilakukan untuk memprediksi sindrom berikutnya.

$$\begin{pmatrix} S_1 & S_2 & S_3 & \dots & S_{t-1} & S_t \\ S_2 & S_3 & S_4 & \dots & S_t & S_{t+1} \\ - & - & - & - & - & - \\ S_{t-1} & S_t & S_{t+1} & \dots & S_{2t-3} & S_{2t-2} \\ S_t & S_{t+1} & S_{t+2} & \dots & S_{2t-2} & S_{2t-1} \end{pmatrix} \begin{pmatrix} \sigma \\ \sigma_{t-1} \\ - \\ - \\ \sigma_2 \\ \sigma_1 \end{pmatrix} = \begin{pmatrix} -S_{t+1} \\ -S_{t+2} \\ - \\ - \\ -S_{2t-2} \\ -S_{2t} \end{pmatrix} \quad (2.12)$$

Setelah nilai-nilai σ pada persamaan (2.3) sudah dapat ditentukan maka terbentuklah persamaan untuk $\sigma(x)$. Lokasi *error* dapat diketahui dengan cara mensubstitusikan akar-akar generator polinomial ke variabel x . Lokasi *error* (β) terjadi pada saat $\sigma(x)=0$

b) Nilai error

Setelah lokasi *error* diketahui, maka *error* dapat ditentukan dengan menggunakan persamaan (2.13)

$$S_i = e_i(\beta) = \sum_{a=1}^t e_a \beta_a \quad (2.13)$$

Dimana $i = 1, 2, \dots, (n-k)/2 = 1, 2, \dots, t$

2.4.2 Kode Konvolusional

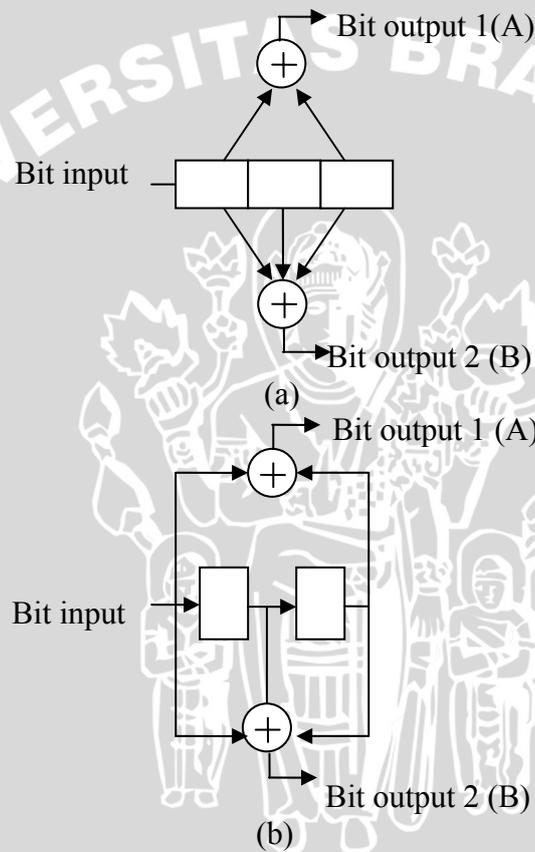
Kode konvolusional merupakan jenis pengkodean yang memetakan setiap bit dari aliran data input. Input data sebanyak k bit akan diubah menjadi output data sebanyak n bit. Rasio dari k/n disebut dengan *code rate*, yang memberi informasi mengenai jumlah bit *redundant* yang ditambahkan. Kode konvolusional tidak memiliki panjang yang tetap seperti kode blok.

2.4.2.1 Enkoder kode konvolusional

Enkoder konvolusional dapat diimplementasikan dengan sejumlah *shift register* dan *adder* modulo-2. Gambar 2.4 (a) menunjukkan rangkaian enkoder kode konvolusional dan Gambar 2.4 (b) merupakan enkoder ekivalennya dengan *rate* $\frac{1}{2}$ dan panjang *shift register* yang disebut sebagai *constraint length* sebanyak 3.



Enkoder ini mempunyai satu data input dan dua buah data output A_i dan B_i yang akan membentuk urutan data output secara bergantian $\{ A_i B_i A_i B_i \dots \}$. Setiap pasang bit output $\{ A_i B_i \}$ tergantung pada tiga bit input, yaitu bit input saat ini ada dua bit input sebelumnya yang berada di *shift register*. Garis penghubung dari *shift register* ke *adder* modulo-2 tergantung dari generator polinomialnya (generator vektor). Sebagai contoh pada gambar 2.4 (a), generator vektornya adalah $\{101, 111\}$. Bit 1 pada generator vektor menunjukkan garis penghubung dari *shift register* menuju *adder* modulo-2.



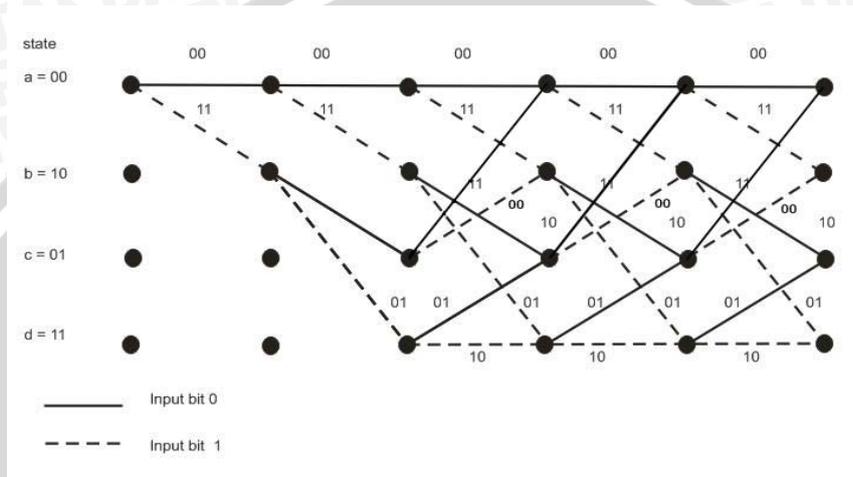
Gambar 2.4 (a) Rangkaian enkoder kode konvolusional, dan (b) Rangkaian enkoder ekivalen dengan *rate* $\frac{1}{2}$ dan *constraint length* 3

Sumber: Digital Communication, Fundamental and Application, 2nd edition, Prentice Hall Inc, 2001. Hal 387

Pada gambar 2.4 (b) yang merupakan rangkaian ekivalen dari rangkaian enkoder pada Gambar 2.4 (a) mempunyai empat macam *state* yaitu *state a*, *b*, *c*, *d*, dimana *state a* = 00, *state b* = 10, *state c* = 01 dan *state d* = 11.



Untuk proses pengkodeannya digunakan diagram *trellis* yang ditunjukkan pada Gambar 2.5, dimana garis putus-putus digunakan untuk jalur bit input "1" dan garis yang tidak putus-putus digunakan untuk jalur bit input "0". Node-node yang terdapat pada diagram *trellis* menunjukkan *state* dari enkoder, dimana baris pertama menunjukkan *state* a = 00, baris kedua dan seterusnya menunjukkan *state* b = 10, *state* c = 01, dan *state* d = 11.

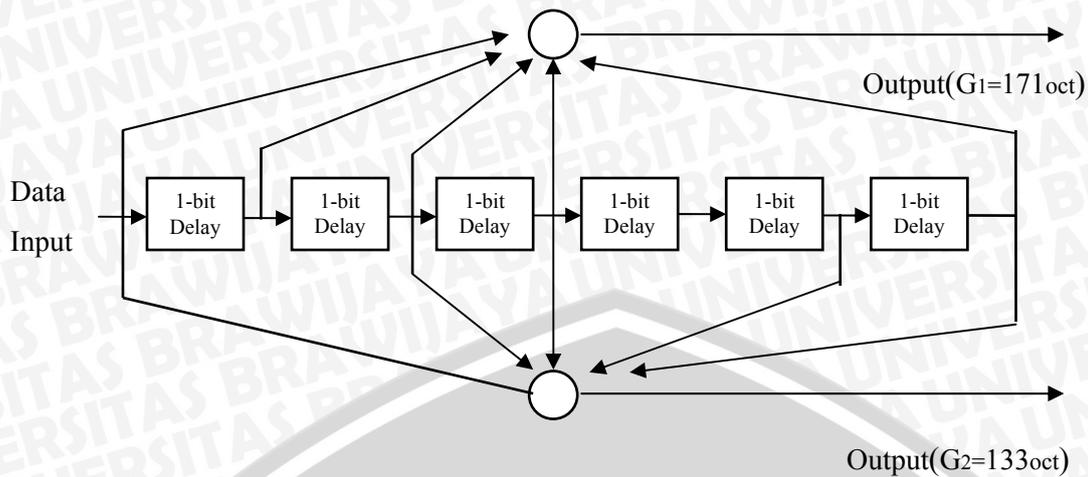


Gambar 2.5 Diagram *trellis* enkoder (*rate* $\frac{1}{2}$ dan *constraint length* 3)

Sumber: Ibid, 2001 hal 394

Dengan menggunakan diagram *trellis* pada Gambar 2.5, apabila bit input yang masuk ke enkoder 11011, maka bit 1 \rightarrow 11, 1 \rightarrow 01, 0 \rightarrow 01, 1 \rightarrow 00, dan 1 \rightarrow 01, sehingga bit-bit output menjadi 1101010001, dimana bit 1 akan melewati jalur dengan garis putus-putus dan bit 0 akan melewati jalur dengan garis tidak putus-putus.

Dalam sistem DVB-T, enkoder kode konvolusional menggunakan rangkaian enkoder dengan *constraint length* 7 dan generator polinomial untuk masing-masing output adalah $1111001 = 171_{\text{oct}}$ untuk generator pertama dan $1011011 = 133_{\text{oct}}$ untuk generator kedua.



Gambar 2.6 Blok diagram enkoder konvolusional dengan *constraint length* 7

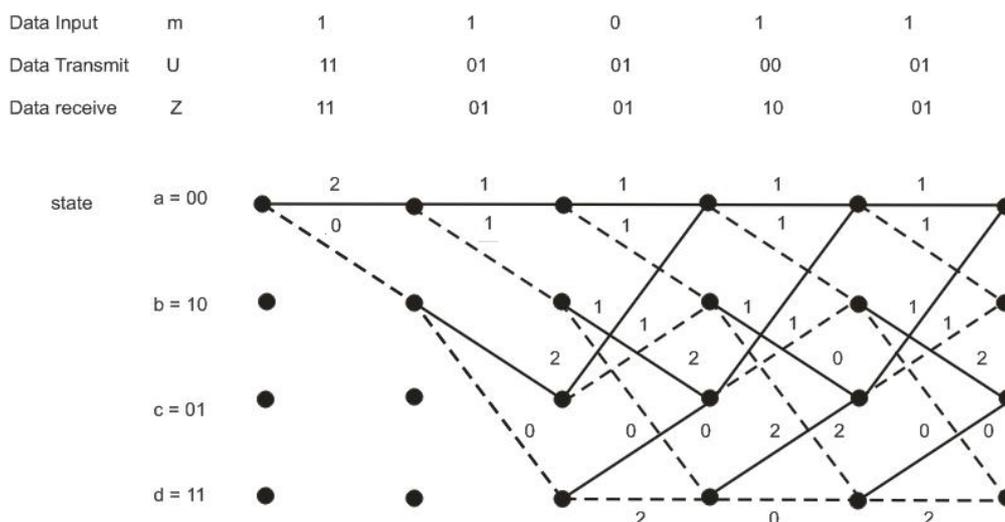
Sumber: Ibid, 2001 hal 394

2.4.2.2 Dekoder Kode Konvolusional

Proses *decoding* dari kode konvolusional menggunakan algoritma Viterbi. Algoritma Viterbi merupakan metode yang sering digunakan, karena metode ini merupakan metode yang paling efektif untuk mendapatkan estimasi *maximum likelihood* yang optimal dari kode konvolusi. Algoritma Viterbi pertama kali dikenalkan oleh A.J. Viterbi.

Pada algoritma Viterbi, pengkodean informasi yang diterima dilakukan dengan cara mencari jalur yang memiliki jumlah *Hamming distance* terkecil. Dalam pemilihan jalur yang akan dilewati, apabila dijumpai adanya 2 jalur yang memiliki *state* yang sama, maka jalur yang akan digunakan adalah jalur dengan *Hamming distance* terkecil. Jalur yang terpilih tersebut disebut dengan *surviving path*.

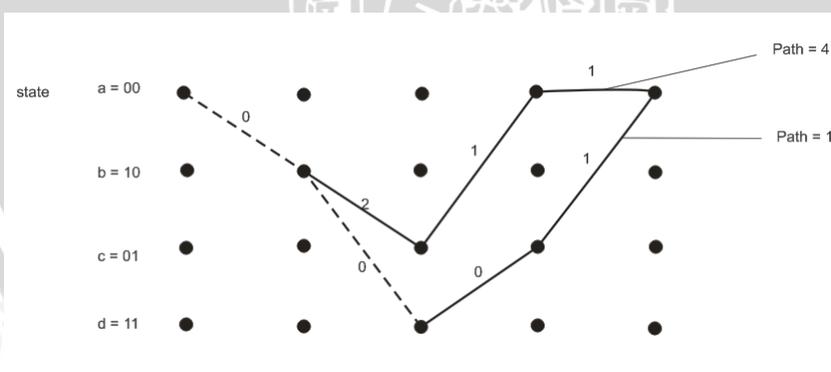
Dekoder konvolusional ini menggunakan diagram *trellis* seperti pada dekoder, tetapi label pada tiap jalurnya saja yang berbeda. Pada dekoder, label yang diberikan pada tiap jalur berupa *Hamming distance*. Adapun diagram *trellis* untuk dekoder konvolusional dapat dilihat pada Gambar 2.7 berikut.



Gambar 2.7 Diagram *Trellis* pada Dekoder (*rate* $\frac{1}{2}$ dan *constraint length* 3)

Sumber: Ibid, 2001 hal 402

Berdasarkan gambar 2.7, pada saat dekoder menerima data 11010110 yang mengalami *error*, maka jalur-jalur yang mungkin untuk dilewati yaitu seperti yang terlihat pada gambar 2.8. Dari gambar 2.8 tersebut, maka jalur yang akan dipilih adalah jalur yang paling bawah dengan jumlah *Hamming distance* 1 karena mempunyai jumlah *Hamming distance* lebih kecil dari pada jalur yang di atasnya yang memiliki jumlah *Hamming distance* 4.



Gambar 2.8 Jalur yang Dilewati untuk Pengkodean Data

Sumber: Ibid, 2001 hal 403

2.5 Data Interleaver

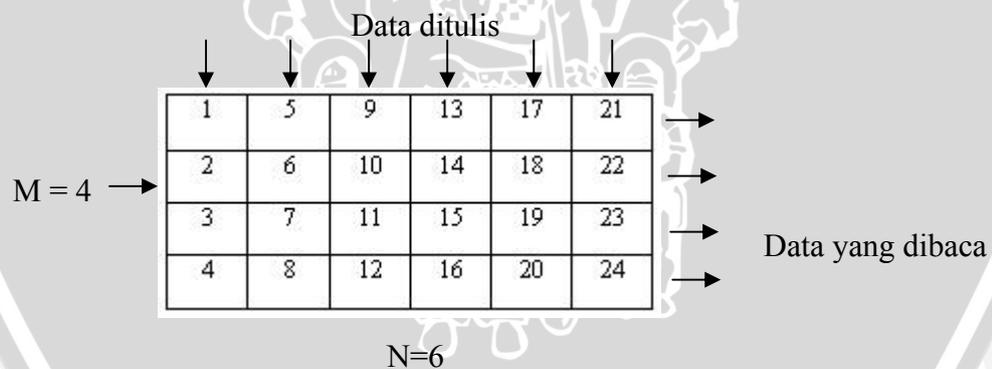
Dalam sistem komunikasi digital, *interleaving* digunakan untuk menghadapi *error burst* dengan cara menyebarkan data informasi dalam waktu tertentu, sehingga data input tidak *corrupt* pada saat yang sama. Disamping

dalam bentuk bit, pengisian *array* bisa juga dalam bentuk simbol. Sistem DVB-T menggunakan dua macam *interleaver*, yaitu *block interleaver* sebagai *inner interleaver* dan *convolutional interleaver* sebagai *outer interleaver*.

2.5.1 Block Interleaver

Prinsip dari *block interleaver* yaitu menerima bit-bit data atau simbol yang berasal dari enkoder, mengubah urutan (permutasi) bit-bit data atau simbol tersebut, kemudian mengumpangkan urutan bit-bit data atau simbol yang baru ke modulator.

Permutasi dilakukan dengan cara menulis bit-bit data atau simbol yang diterima kolom demi kolom dalam bentuk *array* yang berukuran M baris (*depth* M) dan N kolom dan membaca bit-bit data atau simbol tersebut baris demi baris. Sehingga output *block interleaver* akan memisahkan bit-bit yang pada awalnya berdekatan akan terpisah sejauh m bit. Pada *receiver*, *deinterleaver* akan melakukan proses yang berkebalikan dengan *interleaver*. Gambaran bentuk *block interleaver* dengan $M = 4$ baris dan $N = 6$ kolom dapat dilihat pada Gambar 2.9. Output dari *block interleaver* tersebut yaitu 1, 5, 9, 13, 17, 21, 2, 6, ...



Gambar 2.9 Block Interleaver dengan $M = 4$ dan $N = 6$

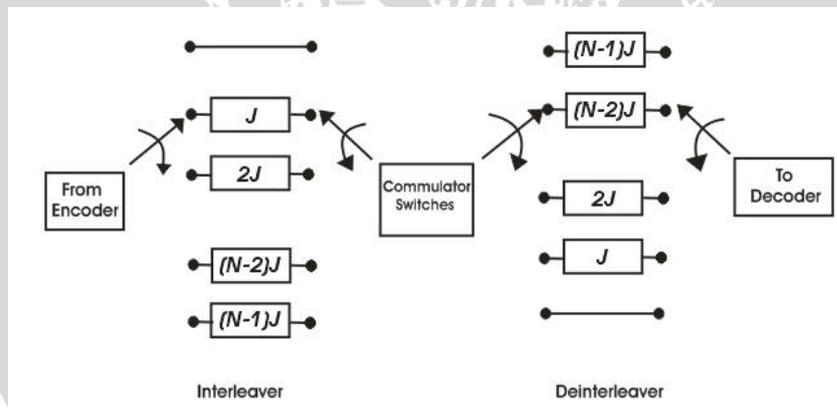
Sumber: Ibid, 2001 hal 464

Pada gambar 2.9 dapat terlihat proses dari *blok interleaver*, dimana data ditulis per kolom dan dibaca per baris.

2.5.2 Interleaver Konvolusional

Bentuk *interleaving* yang lain yaitu *convolutional interleaver*. *Convolutional interleaver* yang diperkenalkan oleh Ramsey dan Forney dengan struktur seperti pada Gambar 2.10. Pada bentuk ini, proses *interleaving* dilakukan dengan menulis setiap input bit atau simbol dalam *shift register* sebanyak J yang memberikan *delay* durasi simbol 0 sampai $N-1$, dan dibaca secara siklik untuk menghasilkan data yang ter-*interleave*.

Kode simbol dipindahkan secara sekuensial ke bank register N , dimana setiap *shift register* mempunyai J simbol lebih banyak dari pada register sebelumnya. Register ke 0 tidak mempunyai *shift register* sehingga data akan langsung dikirim keluar oleh *commutator switch*. Setiap kode simbol yang baru masuk dari enkoder, maka simbol tersebut menuju ke *shift register*, dan kode simbol yang sebelumnya sudah menepati *shift register* tersebut akan digeser atau akan dikeluarkan oleh komutator dari *interleaver* menuju modulator. Setelah register ke $(N-1)$, *commutator switch*. Akan kembali lagi ke register 0 untuk melakukan pemindahan data. Pada penerimaan, *deinterleaver* melakukan operasi yang berkebalikan dengan *interleaver*.

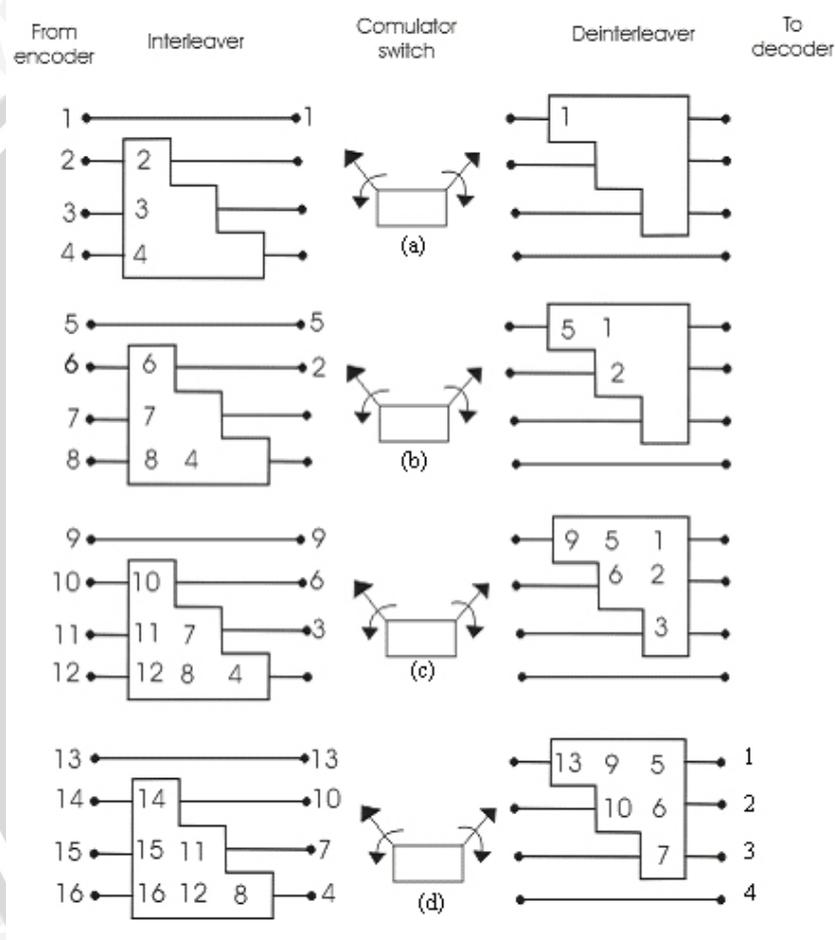


Gambar 2.10 *Convolutional Interleaver* dan *Deinterleaver*

Sumber: Ibid, 2001 hal 466

Untuk lebih jelasnya, proses dari *convolutional interleaver* dapat digambarkan pada Gambar 2.11. Pada Gambar 2.11, apabila simbol 1 sampai 4 masuk, maka simbol 1 saja yang akan langsung dikirimkan, sedangkan simbol-simbol yang lainnya akan berada dalam *shift register* terlebih dahulu, seperti yang terlihat pada Gambar 2.11 (a). Kemudian apabila dimasukan kode-kode

simbol 5 sampai 8, pada Gambar 2.11 (b), maka simbol 5 dan 2 saja yang akan di *switch* oleh komutator sedangkan simbol-simbol yang lain masih berada dalam *shift register*. Gambar 2.11 (c) menunjukkan masuknya simbol-simbol 9 sampai 12 ke interleaver. Pada deinterleaver, dilakukan proses yang berkebalikan. *Shift register* yang berada di deinterleaver akan terisi dan isi dari *shift register* tersebut tidak akan langsung diumpankan ke dekoder simbol-simbol tersebut akan keluar dari *shift register* setelah simbol-simbol berikutnya masuk, yaitu simbol 13 sampai 16, seperti yang terlihat pada Gambar 2.11 (d).



Gambar 2.11 Convolutional Interleaver dengan $J=1$ dan $N=4$

Sumber: Ibid, 2001 hal 470

2.6 Modulasi Digital

Dalam proses modulasi terdapat modulator dan demodulator. Modulator berguna untuk memetakan deretan informasi biner menjadi bentuk sinyal gelombang yang nantinya dikirim melalui kanal. Pada modulasi *binary*,

modulator secara sederhana memetakan digit biner 0 menjadi sinyal gelombang $s_0(t)$ dan digit biner 1 menjadi sinyal gelombang $s_1(t)$.

Pada dasarnya sistem modulasi digital ada tiga macam, yaitu *Amplitude Shift Keying* (ASK), *Frequency Shift Keying* (FSK), dan *Phase Shift Keying* (PSK). Dalam perkembangan selanjutnya muncul berbagai teknik kombinasi misalnya dengan mengkombinasikan antara modulasi dan fase, maka munculah teknik modulasi *Amplitude Phase Keying* (APK), yang dikenal sebagai *Quadrature Amplitude Modulation* (QAM).

2.6.1 Modulasi M-QAM

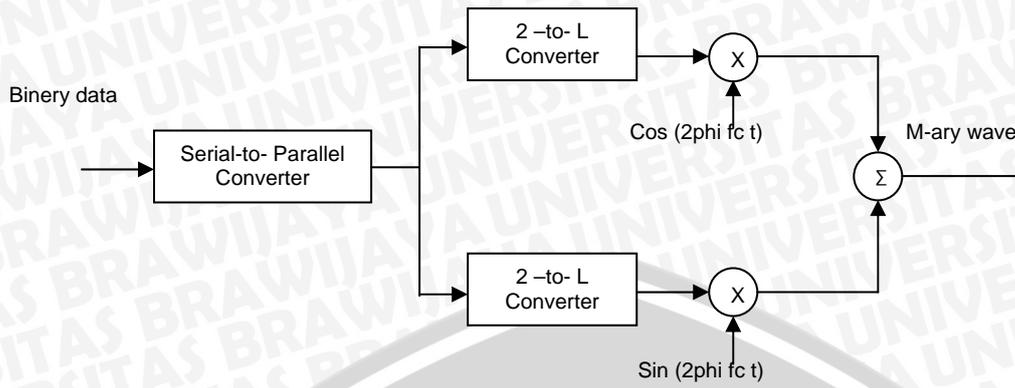
Teknik Modulasi QAM, yang merupakan pengembangan dari ASK (*Amplitude Shift Keying*) dan PSK (*Phase Shift Keying*), adalah suatu bentuk dari modulasi digital dimana sinyal informasi digital diubah amplitudo dan fasenya ke dalam fungsi gelombang pembawa. Secara umum bentuk diagram blok pemancar QAM dapat disusun seperti gambar 2.12. Suatu deretan bit input sebanyak k -bit dipecah oleh *splitter* menjadi dua blok ($k/2$) menjadi bit inphase dan bit *quadrature*.

Bit-bit bernomor ganjil dalam aliaran data dikombinasikan berpasangan untuk memodulasi sinyal *cosinus* (*inphase*) sedang bit-bit bernomor genap untuk memodulasi sinyal *sinus* (*quadrature*). Sinyal *sinus* dan *cosinus* termodulasi membentuk sinyal gabungan sehingga terbentuk modulasi amplitudo dan phase yang secara analitik dapat dinyatakan sebagai persamaan berikut.

$$S_i(t) = \sqrt{\frac{2E_i(t)}{T}} \cos[\omega_c t + \varphi_i(t)] \quad (2.14)$$

$$S_i(t) = A_i(t) \cos[2\pi f_c t + \varphi_i(t)] \quad (2.15)$$

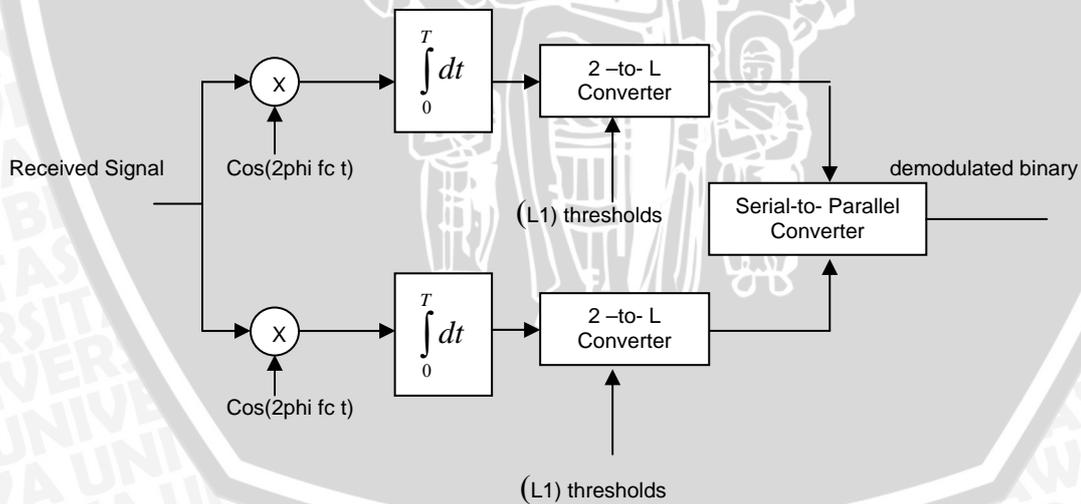
Dimana : $0 \leq t \leq T$ dan $i = 1, 2, \dots, M$; menggambarkan indeks pada *amplitude* dan phase yang mengalami perubahan sebagai fungsi waktu.



Gambar 2.12 Pemancar QAM

Sumber: *Digital Communication*, John Welly&Son.Inc, 1998, Hal.322

Penerima QAM merupakan kebalikan dari pemancar. Setelah sinyal diterima dan dipecah menjadi *inphase* dan *quadrature*, sinyal melalui *product detector* dan menuju *matched filter* untuk dideteksi secara *independen*. Output di-*sample* dan membentuk aliran bit masing-masing lengan (I dan Q) yang siap dikonversi dari paralel ke serial membentuk bit informasi aslinya. Secara umum bentuk diagram blok penerima QAM dapat dilihat pada gambar dibawah:



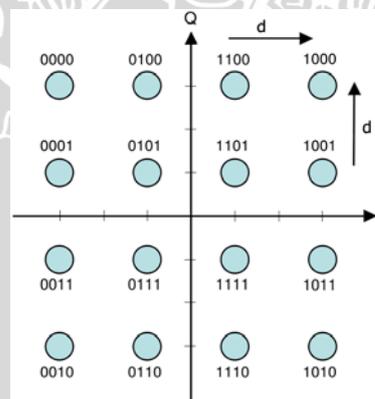
Gambar 2.13 Penerima QAM

Sumber: *Digital Communication*, John Welly&Son.Inc, 1998, Hal.326

2.6.2 Modulasi Baseband 16 QAM

Modulasi berfungsi untuk menyesuaikan karakteristik sinyal agar dapat dilewatkan melalui kanal dan diterima tujuan dengan baik. Pada skripsi ini dipilih 16-QAM (*Quadrature Amplitudo Modulasi*) sebagai bentuk modulasi. 16-QAM banyak digunakan untuk memenuhi pengiriman informasi yang membutuhkan efisiensi *bandwidth* yang tinggi karena 16-QAM dapat memberikan kapasitas yang besar dengan mengirimkan 4 bit melalui 1 simbol sehingga memiliki efisiensi *bandwidth* yang lebih baik dibandingkan *Binary Phase Shift Keying* (BPSK) atau *Quadrature Phase Shift Keying* (QPSK).

Terdapat beberapa macam teknik konstelasi 16-QAM, salah satunya adalah bentuk *rectangular*. Dari suatu bentuk konstelasi, dapat diketahui energi konstelasi dan peluang terjadinya kesalahan pendeteksian bit di penerima. Setiap titik konstelasi pada 16-QAM masing-masing terpisah pada jarak d satuan ditunjukkan seperti pada gambar dibawah yang merupakan konstelasi 16-QAM. Terdapat 2 buah sumbu, yaitu sumbu *inphase* yang merepresentasikan komponen *imajiner*.



Gambar 2.14 Konstelasi Sinyal pada 16 QAM

Sumber: Teknik *Quadrature Amplitude Modulation*, 2005 Hal. 5

Karena sinyal-sinyal ini mempunyai energi serta daya yang bervariasi maka energi yang diperlukan untuk transmisi bisa dievaluasi secara statistik. Untuk M sinyal yang ditransmisikan, energi rata-ratanya adalah:

$$\bar{E} = \frac{1}{M} \sum_{i=1}^M E_i \quad (2.16)$$

Metode pemodelan yang digunakan pada skripsi ini adalah metode *baseband* yang merupakan teknik modulasi *bandpass* yang dibawa ke level *baseband*. Hal ini dilakukan untuk mempermudah proses analisa sinyal pada level *bandpass*. Persamaan sinyal QAM adalah:

$$y(t) = x_1 \cos(2\pi f_c t) - x_2 \sin(2\pi f_c t) \quad (2.17)$$

$$y(t) = \left\{ \text{Re}(x_1(t) + jx_2(t))e^{j2\pi f_c t} \right\} \quad (2.18)$$

$$= \text{Re}\left\{ Y(t)e^{j2\pi f_c t} \right\} \quad (2.19)$$

Dimana:

$$y(t) = \left\{ x_1(t) + jx_2(t) \right\} \quad (2.20)$$

Metode ini dilakukan dengan mempertimbangkan sinyal *passband* dan mengasumsikan sinyal tersebut terjadi pada level *baseband* sehingga tidak lagi mempertimbangkan nilai frekuensi *carrier* dan mempercepat proses simulasi yang berlangsung.

Rangkaian bit modulasi akan dipetakan ke dalam rangkaian simbol modulasi $S(k)$, dimana k adalah nomor simbol. Jumlah bit per simbol bergantung dari tipe modulasi, untuk 16-QAM jumlah bit tiap simbolnya adalah 4 bit per simbol.

2.7 Orthogonal Frequency Division Multiplexing (OFDM)

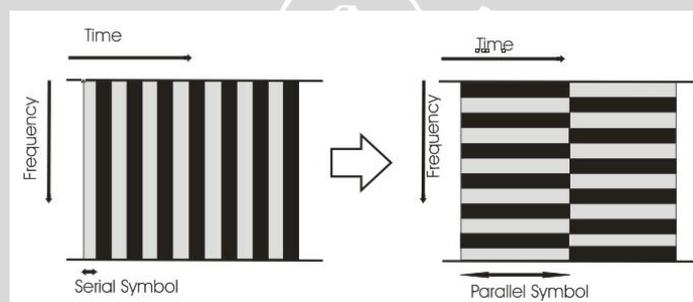
Orthogonal Frequency Division Multiplexing (OFDM) adalah sebuah teknik transmisi yang menggunakan beberapa buah frekuensi (*multicarrier*) yang saling tegak lurus (*orthogonal*). Teknik ini dapat dilakukan dengan membagi *data stream* menjadi beberapa *substream* dengan *data rate* yang lebih rendah dan mentransmisikannya pada *subcarrier* yang saling berdekatan. Pembagian tersebut dapat menggunakan transmisi paralel pada domain frekuensi. Hal itu tidak akan mempengaruhi total *bandwidth* yang dibutuhkan. Pembagian menjadi beberapa *subcarrier* ini dilakukan karena data yang terlalu panjang akan terlalu sensitif terhadap *time incoherence* kanal.

Beberapa keuntungan dari OFDM adalah adanya toleransi terhadap *multipath delay spread* dan efisiensi *spectrum*. Efisiensi *spectrum* dapat dicapai

dengan adanya *overlapping* sinyal pada domain frekuensi. Keuntungan lainnya adalah penggunaan *Inverse Fast Fourier Transform* (IFFT) dan *Fast Fourier Transform* (FFT) pada proses modulasi dan demodulasi yang merupakan proses komputasi yang efisien.

Teknik OFDM digunakan untuk merancang sistem komunikasi dengan *bandwidth* yang lebih efisien, yaitu dengan cara membagi *bandwidth* yang tersedia menjadi beberapa subkanal. Dengan menggunakan teknik ini maka jika semula sebuah sistem transmisi mempunyai *bandwidth* sebesar W , maka jika dibagi kedalam sejumlah N subkanal maka akan didapatkan *bandwidth* untuk setiap subkanal sebesar W/N .

Gambar dibawah menggambarkan proses yang terjadi pada OFDM. Deretan data biner yang dibangkitkan dalam bentuk serial diubah menjadi deretan data biner yang berbentuk paralel.



Gambar 2.15 Konsep *Multicarrier*

Sumber: Theory And Application of OFDM and CDMA, John Welly&Son, 2005, Hal.146

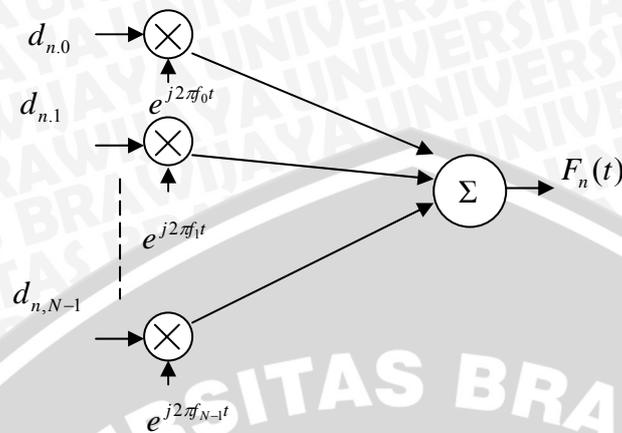
Setelah data diubah dalam bentuk paralel menjadi beberapa *subcarrier* yang saling *orthogonal*, kemudian dilakukan proses modulasi secara *independent* pada masing-masing *subcarrier*.

2.7.1 Sinyal OFDM

OFDM sama dengan FDM, yang membedakannya adalah bahwa N *subcarrier* yang terbentuk bersifat *orthogonal* dengan periode T_s . *Orthogonalitas* ini memenuhi persamaan berikut:

$$f_k = f_0 + \frac{k}{T_s} \text{ untuk } k = 1, 2, \dots, N-1 \quad (2.21)$$

Proses modulasi untuk sistem OFDM yang dilakukan pada pemancar atau *transmitter* dapat digambarkan pada Gambar 2.16 berikut :



Gambar 2.16 Proses Modulasi OFDM (*Transmitter*)

Sumber : Study Teknik OFDM untuk penyiaran TV digital teresterial (DVB-T), Tugas Akhir, Rizka Fadillah, 2006

Sinyal OFDM secara umum dapat direpresentasikan dalam *sum* (penjumlahan) dari N *orthogonal subcarrier* yang terpisah dan termodulasi. Sinyal yang ditransmisikan, secara umum dapat dirumuskan seperti rumus 2.22 berikut ini :

$$s(t) = \sum_{n=-\infty}^{\infty} \sum_{k=0}^{N-1} d_{n,k} g_k(t - nT_s) \quad (2.22)$$

Dimana $g_k(t)$ dengan $k = 0, 1, 2, \dots, N-1$ merupakan 'N' *carrier* dapat dirumus:

$$g_k(t) = e^{j2\pi f_k t}, t \in (0, T_s) \quad (2.23)$$

Pada persamaan (2.22) $d_{n,k}$ merupakan simbol yang modulasi sinyal *carrier* ke- k pada interval ke- n yang mempunyai *periode* T_s . Deretan simbol $d_{n,k}$ diperoleh dengan mengubah data serial dengan periode NT_s menjadi data serial dengan periode $1/T_s$. Hasil dari sinyal yang telah dimodulasi adalah :

$$F_n(t) = \sum_{k=0}^{N-1} d_{n,k} g_k(t - nT_s) \quad (2.24)$$

Dari persamaan 2.24 dapat dilihat bahwa *frame* OFDM $F_n(t)$ yang ke- n terdiri dari 'N' simbol, masing-masing memodulasi salah satu dari 'N' *orthogonal subcarrier*.

Jika persamaan 2.24 disubstitusikan pada persamaan 2.22, diperoleh hasil:

$$s(t) = \sum_{n=-\infty}^{\infty} F_n(t) \quad (2.25)$$

Persamaan 2.25 merupakan persamaan matematis sinyal yang dikirimkan pada proses modulasi menggunakan sistem OFDM.

Pada proses modulasi, sinyal informasi yang telah dibagi menjadi 'N' *subcarrier*, dimodulasi secara *independent*. Hasilnya akan dijumlahkan, baru kemudian ditransmisikan. Pada proses demodulasi, sinyal yang diterima diparalel lagi menjadi 'N' *subcarrier*, kemudian dilakukan proses demodulasi.

Proses ini sama dengan proses demodulasi biasa. Simbol yang didemodulasi dapat diperoleh dengan rumus berikut ini :

$$d'_{n,k} = \frac{1}{T_s} \int_{lT_s}^{(n+1)T_s} s(t) g_k(t) dt \quad (2.26)$$

Sesuai pada gambar 2.16 sinyal yang didemodulasi disamping dengan periode T_s . Tiap sinyal yang masuk pada demodulator tiap waktu n , hasil demodulasinya dapat di rumuskan sebagai berikut :

$$d'_{n,k} = \frac{1}{T_s} \int_{lT_s}^{(n+1)T_s} s(t) g_k(t) dt = d_{n,k} \quad (2.27)$$

Dengan begitu, akan didapat hasil demodulasi tiap *subcarrier*, seperti data yang dikirimkan oleh *transmitter* dengan sempurna dan mengembalikannya menjadi deretan simbol seperti semula.

2.7.2 Implementasi OFDM dengan IFFT dan FFT

Proses modulasi dan demodulasi pada sistem OFDM membutuhkan 'N' modulator dan 'N' demodulator. Jumlah *subcarrier* pada sistem OFDM biasanya mempunyai *orde* ratusan bahkan ribuan. Hal tersebut membuat blok

transmitter dan receiver sangat sulit untuk direalisasikan. Untuk mengatasi masalah tersebut digunakan *Discrete Fourier Transform*.

Jika pada sinyal yang telah dimodulasi seperti pada persamaan 2.24 disampel dengan *rate* ' N/T_s ' akan dihasilkan sinyal diskrit :

$$F_n^m = \sum_{k=0}^{N-1} d_{n,k} g_k(t - nT_s) \text{ untuk } t = (n + \frac{m}{N})T_s \quad (2.28)$$

Persamaan tersebut berlaku untuk nilai $m = 0, 1, \dots, N-1$. Dengan mensubstitusikan persamaan 2.23 ke dalam persamaan 2.28 diatas diperoleh :

$$F_n^m = e^{j2\pi f_0 \frac{m}{N} T_s} \sum_{k=0}^{N-1} d_{n,k} e^{j2\pi \frac{mk}{N}} \quad (2.29)$$

Jika dari persamaan 2.29 diasumsikan nilai dari $f_0 = 0$, maka persamaan di atas menjadi :

$$F_n^m = \sum_{k=0}^{N-1} d_{n,k} e^{j2\pi \frac{mk}{N}} \quad (2.30)$$

Persamaan 2.30 dapat dituliskan dalam bentuk IFFT, sehingga menjadi :

$$F_n^m = N.IFFT(d_{n,k}) \quad (2.31)$$

IFFT dilakukan pada sisi pemancar OFDM untuk merepresentasikan proses modulasi. Oleh karena itu, pada sisi penerima dilakukan FFT agar sinyal informasi kembali seperti semula. Persamaan untuk proses FFT adalah sebagai berikut :

$$FFT(F_n^m) = N.FFT(IFFT(d_{n,k})) = d_{n,k} \quad (2.32)$$

IFFT dan FFT membuat persamaan untuk proses modulasi dan demodulasi menjadi lebih sederhana dan mudah untuk diaplikasikan pada sistem OFDM.

2.7.3 Guard Interval

Pada OFDM, sinyal didesain sedemikian rupa agar saling *orthogonal*, sehingga bila tidak ada *distorsi* pada jalur komunikasi maka setiap *subchannel*

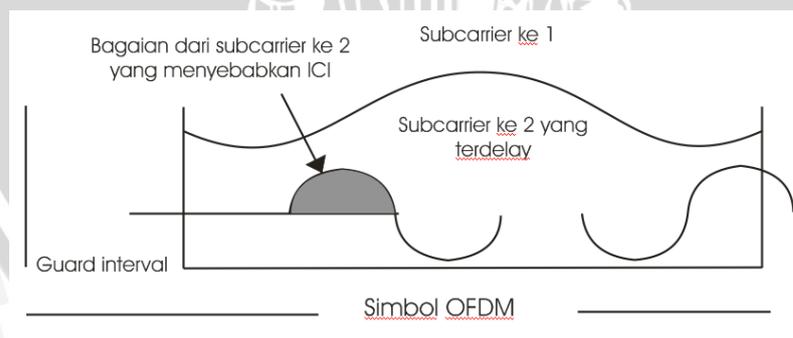
akan dapat dipisahkan oleh penerima menggunakan FFT. Tetapi pada kenyataannya tidak semudah itu, kemungkinan *distorsi* pada jalur komunikasi sangat besar, sehingga *orthogonalitas* sistem tidak dapat dipertahankan yang pada akhirnya akan menyebabkan interferensi antar simbol atau *inter symbol interference* (ISI).

Untuk memecahkan masalah tersebut digunakan *guard interval* yang disisipkan secara periodik pada setiap simbol OFDM, sehingga total periode simbol menjadi:

$$T_{total} = T_{guard} + T_u$$

Dimana T_{total} adalah periode total simbol, T_{guard} adalah periode *guard interval*, dan T_u adalah periode efektif simbol.

Adapun periode dari *guard interval* yang lebih besar dari *delay* yang disebabkan adanya *multipath* bertujuan agar keorthogonalan sistem masih bisa terjaga sehingga simbol yang satu tidak akan tercampur atau menginterferensi simbol yang ada didekatnya. *Guard interval* tersebut bisa tidak mengandung informasi sama sekali. Namun, penyisipan *guard interval* seperti itu dapat menyebabkan timbulnya interferensi antar *carrier* atau *inter carrier interference* (ICI) karena *subcarrier* tersebut tidak lagi saling *orthogonal* antara yang satu dengan yang lainnya. Pengaruh penyisipan *guard interval* yang tidak mengandung informasi sama sekali terhadap keorthogonalan antar *subcarrier* dapat diilustrasikan pada gambar berikut:

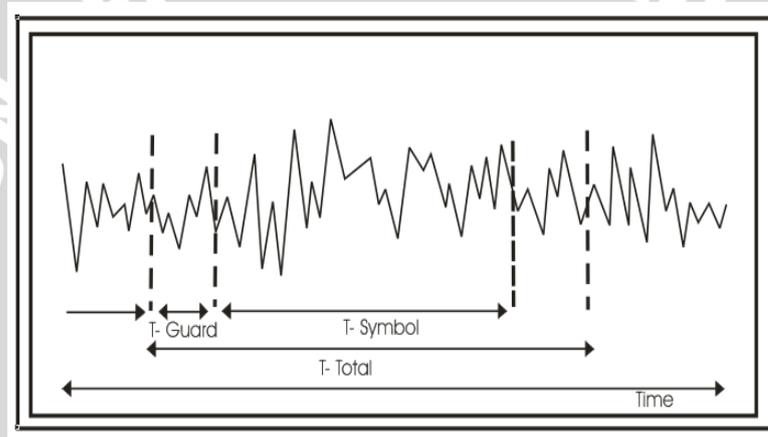


Gambar 2.17 Pengaruh *Multipath* pada *Guard Interval* yang tidak Mengandung Informasi

Sumber : Study Teknik OFDM untuk penyiaran TV digital teresterial (DVB-T), Tugas Akhir, Rizka Fadillah, 2006

Pada gambar diatas dapat dilihat bahwa pada saat penerima OFDM mencoba untuk mendemodulasi *subcarrier* 1, maka penerima akan menemui interferensi yang berasal dari *subcarrier* 2, begitu juga apabila *subcarrier* 2 didemodulasi maka penerima juga akan menemui interferensi yang berasal dari *subcarrier* 1.

Untuk mengatasi masalah ICI tersebut maka *guard interval* berisi hasil salinan dari bagaian akhir simbol yang diletakkan pada bagaian awal simbol (*cyclic prefix*). Sehingga pada akhirnya, *delay* yang disebabkan *multipath* tidak akan mempengaruhi keorthogonalan sistem selama *delay* tersebut memiliki durasi lebih kecil dari *guard interval*. Cara penyisipan *guard interval* dapat diilustrasikan pada gambar dibawah.



Gambar 2.18 Simbol OFDM dengan *Cyclic Prefix*

Sumber : Study Teknik OFDM untuk penyiaran TV digital teresterial (DVB-T), Tugas Akhir, Rizka Fadillah, 2006

2.8 Signal to Noise Ratio

Kinerja atau performa dari suatu sistem komunikasi dapat dilihat dari nilai *signal to noise ratio* (SNR) yang diukur pada daerah *baseband* atau dari nilai *carrier to noise ratio* (CNR) yang diukur pada daerah *passband*.

Perhitungan SNR secara matematis dapat dituliskan sebagai berikut:

$$\frac{S}{N} = \frac{E_b}{N_0} \frac{R_b}{W} \quad (2.33)$$

Dimana:

S = daya sinyal

N = daya noise

E_b = energi bit (dB μ V)

N_0 = kerapatan daya noise (Watt/Hz)

R_b = kecepatan bit (μs)

W = bandwidth yang digunakan (MHz)

Sedangkan untuk perhitungan CNR dari suatu sistem komunikasi dapat dituliskan pada persamaan sebagai berikut:

$$\frac{S}{N} = \frac{E_s R_s}{N_0 W} \quad (2.34)$$

Keterangan:

S = daya sinyal

N = daya *noise*

E_s = energi simbol ($\text{dB}\mu\text{V}$)

R_s = kecepatan simbol (μs)

N_0 = kerapatan daya *noise* (Watt/Hz) W = bandwidth yang digunakan (MHz)

Dengan $E_s = E_b \cdot m$ dan $R_s = R_b / m$, dimana m merupakan banyaknya bit dalam satu simbol, maka:

$$\frac{C}{N} = \frac{E_b \cdot m R_b / m}{N_0 W} = \frac{E_b R_b}{N_0 W} \quad (2.35)$$

Sehingga:

$$\frac{S}{N} \equiv \frac{C}{N}$$

2.9 Kanal Propagasi

Kanal dapat diartikan sebagai lintasan antara penerima dan pemancar. Kondisi kanal tidak pernah mencapai keadaan yang ideal. Keadaan ideal suatu kanal terjadi apabila sinyal yang dikirimkan oleh *transmitter* akan diterima pada *receiver* berupa sinyal tunggal yang merupakan bagaian langsung dari sinyal yang dikirim sehingga dengan sempurna akan direkonstruksi oleh penerima.

Pada kenyataannya, kanal tidak pernah dalam kondisi ideal. Selalu ada faktor luar yang mengganggu sinyal. Pantulan, pembelokan dan hamburan dari sinyal merupakan beberapa macam penyebab sinyal yang diterima oleh penerima tidak sama dengan sinyal yang dikirim oleh pemancar. Disamping itu terdapat pula *noise* yang dapat menyebabkan perubahan karakteristik sinyal yang dikirim. *Noise* yang selalu terjadi pada setiap kanal adalah *Additive White Gaussian Noise* (AWGN).

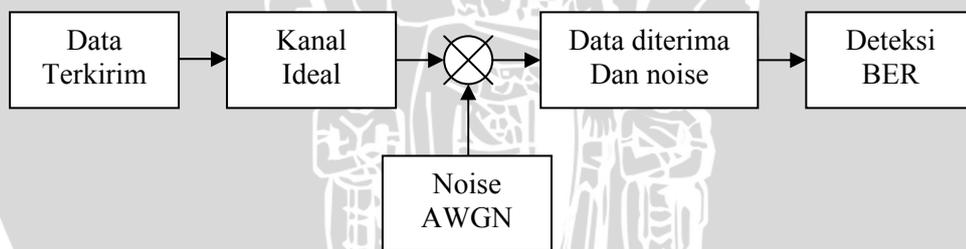
2.9.1 Additive White Gaussian Noise (AWGN).

AWGN adalah *noise* yang terjadi dalam jaringan *wireless* manapun, memiliki sifat-sifat *additive*, *white*, dan *gaussian*. Disebut *additive* karena *noise* yang timbul hanya bersifat menjumlahkan (*adder*) terhadap sinyal, sifat *white* artinya *noise* tidak tergantung dari frekuensi operasi sistem dan memiliki rapat daya yang konstan, dan sifat *gaussian* artinya besarnya tegangan *noise* memiliki rapat peluang terdistribusi *gaussian*. Secara teoritis AWGN sering dimodelkan sebagai derau kanal. Derau ini dapat dideskripsikan sebagai proses acak terdistribusi *gaussian* dengan rata-rata nol. Kanal AWGN merupakan model yang baik untuk digunakan pada kenyataan fisik suatu sistem. Model kanal AWGN dapat dikarakteristikan sebagai berikut:

1. *Noise* $n(t)$ adalah *complex* AWGN *baseband* yang bersifat mengganggu sinyal asli $s(t)$. Akibat *noise* $n(t)$ sinyal yang diterima oleh penerima adalah:

$$r(t) = s(t) + n(t) \quad (2.36)$$

2. AWGN bersifat stasioner dengan *mean* sama dengan nol serta terdistribusi secara acak menurut *gaussian*.



Gambar 2.19 Kanal AWGN

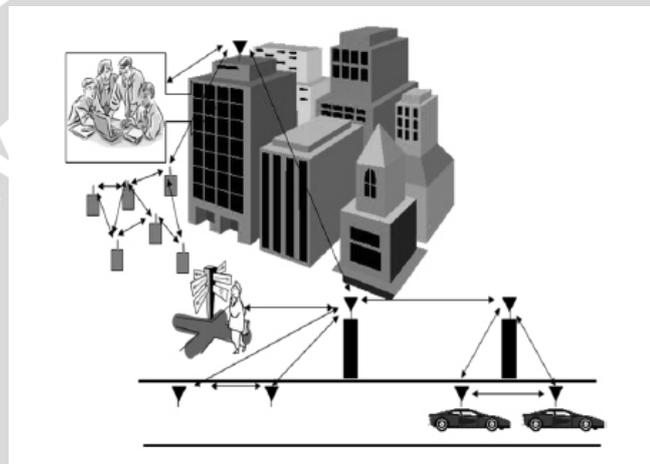
Sumber : Sistem komunikasi Seluler, Arif Nugroho, 2004 hal. 23

Proses acak *gaussian* $n(t)$ merupakan fungsi acak dengan harga n , pada saat *arbitrary time* t , dapat dikarakteristikan menggunakan *gaussian probability density function* atau fungsi rapat probabilitas dengan rumus:

$$p(n) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{n}{\sigma}\right)^2\right] \quad (2.37)$$

2.9.2 Kanal Rayleigh

Kanal *Rayleigh* merupakan model yang cocok untuk menggambarkan fenomena yang terjadi dalam *wireless communication*. Fenomena yang terjadi meliputi efek *multipath*, *time dispersion* dan *dopper shifts* yang terjadi dalam sistem komunikasi bergerak. Dalam sebuah lintasan radio (*radio link*), sinyal dari pemancar akan mengalami berbagai macam benturan dengan objek atau benda-benda yang ada di sekitar lintasan radio tersebut, seperti yang diilustrasikan pada gambar di bawah ini:



Gambar 2.20 Fenomena pada Lintasan Radio

Sumber: Wireless Communcation, Principles and Practice, Pretice Hall, Inc, Hal.112

Sinyal yang diterima oleh penerima merupakan jumlah superposisi dari keseluruhan sinyal yang dipantulkan akibat *multipath*. Hal ini menyebabkan kuat sinyal yang diterima oleh penerima akan bervariasi dengan cepat, dan terjadi fenomena sinyal *fading* cepat (*short term fading*). Karena rendahnya antena pemancar dan adanya struktur bangunan yang mengelilinginya menyebabkan fluktuasi yang cepat pada penjumlahan sinyal-sinyal *multipath* menurut distribusi statistik yang disebut *Rayleigh Fading*.

Ada tiga mekanisme dasar yang berbenturan dengan propagasi sinyal dalam sistem komunikasi gelombang radio, yaitu:

1. Pantulan (*refleksi*), terjadi ketika perambatan gelombang elektromagnet membentur sebuah permukaan yang halus dengan dimensi yang sangat

besar dibanding panjang gelombang sinyal , misalnya gedung dan dinding.

2. Pembelokan (*difleksi*), terjadi ketika lintasan radio dari pemancar ke penerima dihalangi oleh benda yang mempunyai permukaan yang tajam, misalnya pinggiran dari atap gedung.
3. Hamburan (*scattering*), terjadi ketika sebuah gelombang radio membentuk permukaan yang mempunyai dimensi yang lebih kecil dari panjang gelombangnya dan jumlahnya banyak, sehingga pantulan akan menyebar ke semua arah, misalnya daun pepohonan, rambu-rambu lalu lintas.

Akibat adanya pantulan, pembelokan dan hamburan menyebabkan timbulnya bermacam-macam lintasan dari pemancar ke penerima yang disebut sebagai lintas jamak (*multipath*) yang menyebabkan perubahan selubung sinyal yang diterima. Distribusi yang biasanya digunakan untuk menerangkan selubung sinyal yang melemah dan datar (*flat fading signal*) adalah distribusi *Rayleigh*. Sinyal yang terdistribusi *Rayleigh* ditujukan sebagai fungsi waktu dengan rumus sebagai berikut:

$$p(r) = \left\{ \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right), 0 \leq r \leq \infty \right. \quad (2.38)$$

Dimana: σ = nilai level sinyal yang diterima sebelum detektor

σ^2 = daya waktu rata-rata yang diterima sebelum detektor

Fungsi ditribusi kumulatif menyatakan presentasi lebih kecil dari nilai R_a tertentu, yang diperoleh dengan melakukan integrasi terhadap fungsi rapat peluang $p(r_a)$.

$$P(R_a) = \Pr_a(r_a \leq R_a) = \int_0^{R_a} p(r_a) dr_a = 1 - \exp\left(-\frac{R_a}{2\sigma^2}\right) \quad (2.39)$$

$r_{a,mean}$ distribusi *reyleigh* adalah:

$$r_{a,mean} = E[r_a] = \int_0^{\infty} r_a p(r_a) dr_a = \sigma \sqrt{\frac{\pi}{2}} = 1,2533\sigma \quad (2.40)$$

$\sigma^2_{r_a}$ merupakan varian dari distribusi *Reyleigh* yang mewakili daya ac pada selubung sinyal.

$$\begin{aligned}\sigma^2_{r_a} &= E[r_a^2] - E^2[r_a] = \int_0^{\infty} r_a^2 p(r_a) dr_a - \frac{\sigma^2}{2} \\ &= \sigma^2 \left(2 - \frac{\pi}{2} \right) = 0.4292\sigma^2\end{aligned}\quad (2.41)$$

Untuk menghitung nilai tengah r_a dapat digunakan persamaan berikut ini :

$$\frac{1}{2} = \int_0^{r_{a,median}} p(r_a) dr_a \Rightarrow r_{a,median} = 1,177\sigma \quad (2.42)$$

Jadi terdapat perbedaan nilai rata-rata dan nilai tengah sebesar 0,55 dB dalam distribusi *Reyleigh*. Sebagai catatan bahwa nilai tengah pada prakteknya sering digunakan, karena data *fading* biasanya diukur di lapangan dan kenyataannya distribusinya tidak akan diasumsikan. Dengan penggunaan nilai tengah sebagai pengganti nilai rata-rata maka dengan mudah kita membandingkan distribusi *fading* yang berbeda-beda dan memiliki nilai tengah yang bermacam-macam.

2.10 Teknik Estimasi BER Monte Carlo

Metode *Monte Carlo* ini digunakan untuk melakukan analisa keputusan yang melibatkan beberapa parameter untuk dilakukan pertimbangan secara simultan. Metode ini dapat digunakan secara luas karena didasarkan pada proses simulasi dengan pilihan kemungkinan secara *random*. Dengan demikian, jumlah iterasi yang dilakukan sangat menentukan tingkat ketelitian atas jawaban yang diperoleh. Metode ini seringkali juga disebut dengan metode percobaan statistik (*method of statistical trials*).

Metode ini mengasumsikan pola kejadian variabel perhitungannya pada dua model distribusi yaitu distribusi normal dan distribusi *uniform*. Asumsi ini dapat melemahkan suatu kasus yang mempunyai pola distribusi diluar kedua asumsi tersebut. Namun dengan sedikit melakukan usaha manipulasi statistik dengan melakukan transformasi data mentah pada variabel yang bersangkutan diubah untuk memenuhi dua asumsi distribusi tersebut dapat dilakukan dengan sederhana. Terdapat beberapa metode untuk mengestimasi BER dalam simulasi komputer, diantaranya adalah:

1. Metode simulasi *Monte Carlo*
2. Metode termodifikasi *Monte Carlo*

3. Teori Harga Ekstrem
4. *Tail Extrapolation*
5. *Quasi-Analytical*

Masing-masing teknik diatas, mempunyai kelebihan dan kekurangan yang dibedakan oleh cara penanganan dari fungsi rapat atau distribusi peluangnya. Dalam skripsi ini digunakan metode simulasi *Monte Carlo*.

Metode simulasi *Monte Carlo* merupakan metode simulasi estimasi BER yang sederhana, tetapi memerlukan waktu *running* yang lama. Jika kondisi dikirim dinyatakan sebagai ‘nol’ dan peluang bersyarat *bit* terkirim salah sebagai ‘nol’, maka dapat dinyatakan dalam persamaan berikut ini:

$$P(\text{error} / \text{'nol'}) = P_0 = \int_{V_T} f_0(v)dv \tag{2.43}$$

dengan V_T merupakan tegangan *threshold* pada proses *decision* dan $f_0(v)$ merupakan tegangan input pada saat pencuplikan, jika bit yang dikirim adalah nol. Persamaan 2.43 dapat dinyatakan menjadi:

$$P_0 = \int_{-\infty}^{\infty} h_0(v)f_0(v)dv \tag{2.44}$$

$$\text{dengan } h_0(v) = \begin{cases} 1, \dots, v \geq V_T \\ 1, \dots, v \geq V_T \end{cases}$$

Persamaan 2.44, ekuivalen dengan persamaan berikut:

$$P_0 = E[h_0(v)] \tag{2.45}$$

dan estimasi nilai P_0 adalah nilai rata-rata dari *sample*:

$$P_0 = \frac{1}{N} \sum_{i=1}^N h_0(v_i) \tag{2.46}$$

dengan $v_i = v(t_i)$, dan t_i adalah waktu simbol saat *decision* dilakukan. Lebih jelasnya, $h_0(v)$ adalah penjumlahan kesalahan yang terjadi. sehingga jika dalam sistem terdapat sebanyak N bit yang diproses dan ada n bit yang *error*, maka BER dapat dihitung dengan mudah, dengan rata-rata *sample*:

$$BER = \frac{n}{N} \tag{2.47}$$

Pada teknik *Monte Carlo*, nilai BER yang diinginkan akan menentukan besarnya jumlah data minimum yang harus dibangkitkan, jika jumlah data yang harus dibangkitkan adalah B , maka hubungan tersebut dapat dinyatakan dalam persamaan berikut:

$$B = \frac{10}{BER} \quad (2.48)$$

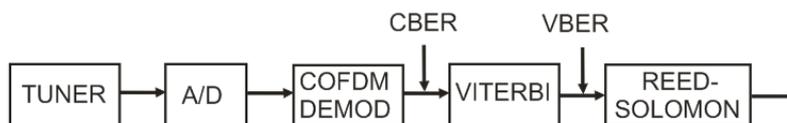
2.11 Alat Ukur

Jenis alat ukur yang digunakan yaitu Prodig-5 untuk mengukur *path loss* dan kualitas cakupan sedangkan alat ukur Pixelmetrix digunakan untuk mengukur *delay spread*. Kedua alat tersebut memiliki spesifikasi serta fungsi yang berbeda yaitu:

2.11.1 Prodig-5

Alat ukur ini digunakan untuk melakukan pengukuran *path loss* dan pengukuran kualitas cakupan. Merupakan alat ukur yang digunakan untuk mengukur nilai BER dari sistem yang diimplementasikan. Alat ukur ini tidak melakukan pengukuran terhadap nilai BER *output* sistem, melainkan nilai BER sebelum proses FEC (*Forward Error Corection*) atau setelah OFDM yang disebut dengan CBER, serta nilai BER setelah proses viterbi *decoding*, sebelum masuk RS (Reed-Solomon) *Decoder* yang disebut dengan VBER. Ketika proses pengukuran dilakukan, alat ini diset sesuai dengan *setting* pemancar yaitu:

1. Frekuensi kerja 578 Mhz
2. OFDM dengan carrier 8k
3. Modulasi 16 QAM
4. *Code Rate* $\frac{3}{4}$
5. *Guard Interval* 1/16



Gambar 2.21 Blok Diagram PRODIG-5

Sumber: PROMAX, user's manual prodig-5, 2005 hal. 4

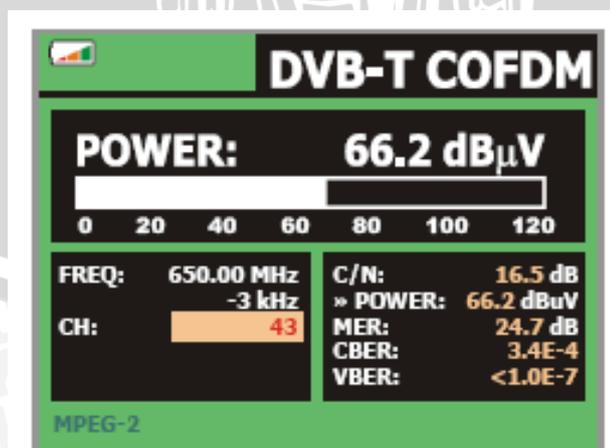
Sedangkan tampilan dari alat ukur Prodig-5 dapat dilihat pada gambar dibawah ini:



Gambar 2.22 Alat Ukur PRODIG-5

Sumber: PROMAX, user's manual prodig-5, 2005 hal. 2

Parameter sitem DVB-T yang dapat diukur oleh alat Prodig-5 ini adalah daya sinyal, CBER (BER sebelum blok viterbi), VBER (BER setelah blok viterbi) MER (*Modulation Error Rate*) serta C/N (*Carrier-to-Noise Ratio*). Sedangkan hasil yang dipergunakan pada skripsi ini adalah VBER dan C/N saja.



Gambar 2.23 Contoh Tampilan Hasil Pengukuran dengan PRODIG-5

Sumber: PROMAX, user's manual prodig-5, 2005 hal. 42

2.11.2 Pixelmetrix

Alat ukur ini digunakan untuk pengukuran *delay spread*, dimana terdapat 3 bagian yaitu CPU, *transport Stream* dan *COFDM interface*. Alat ukur ini dapat mengukur parameter yang tidak dapat dilakukan oleh Prodig-5, seperti *impulse response*, *Bandwidth*, *Packet interval*. Namun karena alat ini tidak praktis, maka alat ukur ini hanya digunakan untuk mengukur *delay spread* sinyal, dan hanya beberapa titik lokasi saja yang diukur nilai *delay spread*. Adapun spesifikasi dari alat ukur Pixelmetrix adalah sebagai berikut :

1. CPU (Main Frame)

- Prosesor Pentium 3 700 MHz
- Memori 256 Mb
- Hardisk 30,7 Mb
- Tegangan input 85-135 V (10 A), 180-256 V (5 A)

2. Transport Stream

- Prosesor 200 MHz *Power PC local Processor*
- Impedansi input 75 Ω
- Level sinyal -40 sampai 0 dBm
- Tipe konektor dual BNC

3. COFDM interface

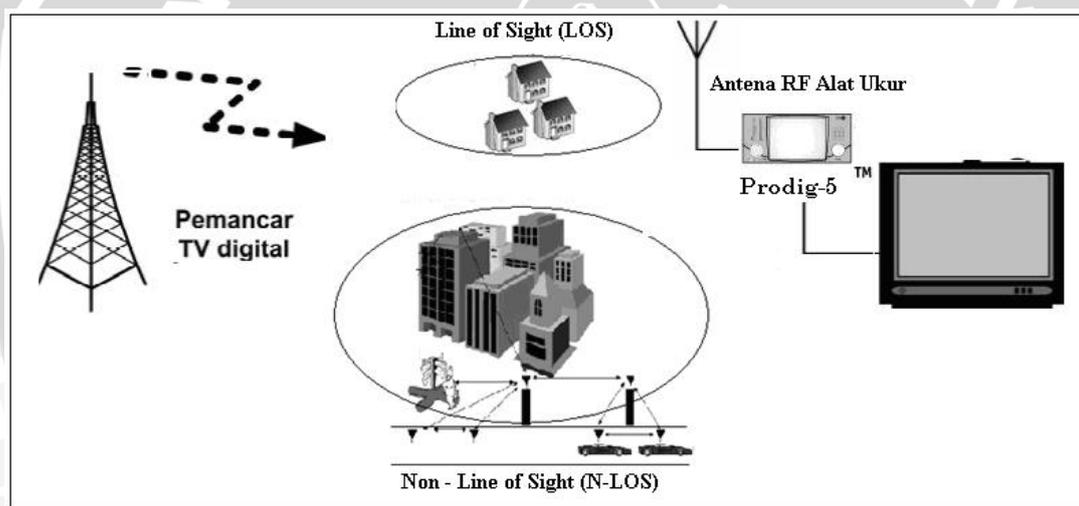
- Prosesor 200 MHz *Power PC local Processor*
- Standard DVB-T COFDM UHF dan VHM
- Impedansi input 75 Ω
- Range UHF 470-826 MHz, VHF 174-230 MHz
- Input power level -6 sampai -65
- Modulasi QPSK, 16 QAM, 64 QAM
- *Guard Interval* $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, dan $\frac{1}{32}$
- *Code rate* $\frac{1}{2}$, $\frac{2}{3}$, $\frac{3}{4}$, $\frac{5}{6}$, dan $\frac{7}{8}$
- Jumlah *carrier* 2k dan 8k

2.12 Jenis Pengukuran

Jenis pengukuran yang digunakan sebagai bahan analisis pada sistem *Digital Video Broadcast Terrestrial (DVB-T)* adalah sebagai berikut:

2.12.1 Pengukuran *Path Loss*

Path Loss adalah Model Redaman lintasan yang digunakan untuk sinyal dengan perambatan *Line of Sight (LOS)* atau *Non- Line of Sight (NLOS)*. Atau bisa disebut *loss* yang terjadi ketika data / sinyal melewati media udara dari antenna penerima dalam jarak tertentu. *Path loss* dapat timbul disebabkan oleh banyak faktor, seperti kontur tanah, lingkungan yang berbeda, medium propagasi (udara yang kering atau lembab), jarak antara antenna pemancar dengan penerima, lokasi dan tinggi antenna.



Gambar 2.24 Propagasi pengukuran *path loss*

Sumber: Wireless Commnucation, Principles and Practice, Pretice Hall, Inc, Hal.167

2.12.1.1 Metode *Path loss* (umum)

Path loss merupakan komponen penting dalam perhitungan dan analisis desain *link budget* sistem telekomunikasi. Perhitungan *path loss* dengan menggunakan rumus *Okumura-Hata* model untuk urban area. Model Hata didasarkan atas pengukuran empiris ekstensif yang dilakukan di lingkungan perkotaan. Dengan jarak antara *mobile station* ke *base station* dibuat teratur, mulai dari jarak 0,1 km sampai dengan jarak 20 km.

Menggunakan persamaan *Hata* dapat diringkas sebagai berikut:

$$L_{Hata}(\text{urban})[\text{dB}] = 69,55 + 26,16 \times \log(f) + [44,9 - 6,55 \times \log(hb)] \times \log(d) - 13,82 \times \log(hb) - A(hm)$$

Dimana:

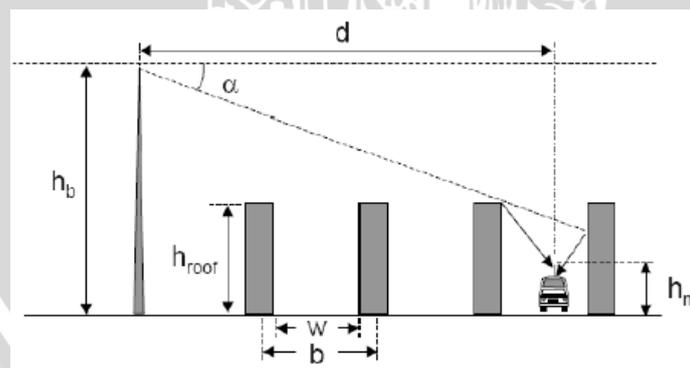
$$A(hm)[\text{dB}] = [11 \times \log(f) - 0,7] \times hm - [1,56 \times \log(f) - 0,8] \quad (2.49)$$

Dengan

- Lhata : *Path loss* (dB)
- F : frekuensi (MHz)
- Hb : node B antenna height (m) = 30 m
- d : jarak dari node B ke antenna mobile (km)
- A (hm) : *mobile antenna height gain correction factor*
- Hm : *mobile antenna height* (m) = 1,5 m

Dengan persamaan Model *Wasfish-Ikegami*, Model ini digunakan untuk menyempurnakan perhitungan *path loss* dengan mempertimbangkan data pada daerah *urban*, diantaranya:

- Ketinggian gedung (penghalang), h_r
- Lebar jalan, w
- Jarak antar gedung, b



Gambar 2.25 Model Propagasi Lintasan *Path Loss*

Sumber: Wireless Commnucation, Principles and Practice, Pretice Hall, Inc, Hal.168

Model ini dibedakan berdasar kondisi LOS dan N-LOS. Untuk kondisi LOS dapat dicari dengan persamaan sebagai berikut:

$$L = 42,6 + 26 \log d(km) + 20 \log f(MHz); d \geq 20km \quad (2.50)$$

Untuk kondisi N-LOS, total rugi-rugi transmisi merupakan penjumlahan dari *free space loss* (L_{fs}), *rooftop to street diffraction and scatter loss* (L_{rts}) dan *multiscreen loss* (L_{msd}), seperti ditunjukkan pada persamaan berikut ini:

$$L = L_{FS} + L_{RTS} + L_{MSD}; \text{ untuk } L_{RTS} + L_{MSD} > 0 \quad (2.51)$$

Dengan:

L_{fs} = *Free Space Loss*,

L_{rts} = *Rooftop to street diffraction loss*.

Nilai L_{fs} dapat diperoleh dari persamaan:

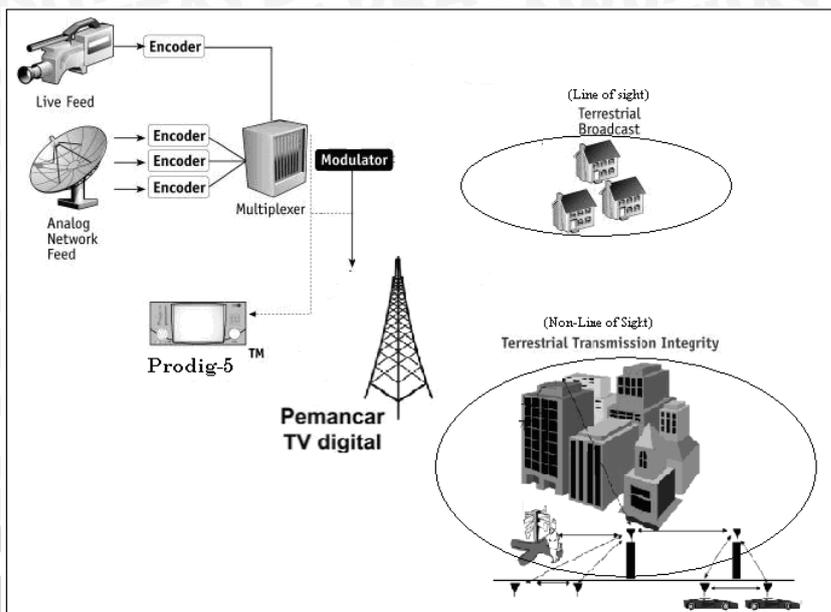
$$L_{FS} = 32,4 + 20 \log d(km) + 20 \log f(MHz) \quad (2.52)$$

Nilai L_{rts} dapat diperoleh dari persamaan:

$$L_{RTS} = -16,9 - 10 \log w(m) + 10 \log f(MHz) + 20 \log \Delta h_{mobile}(m) + L_{ori} \quad (2.53)$$

2.12.2 Pengukuran Kualitas Cakupan

Kualitas cakupan atau *coverage area* adalah suatu cakupan daerah yang dapat menerima atau mengakses sebuah informasi dari pemancar. Penentuan daerah cakupan dapat dilakukan dengan mengukur daya pada berbagai titik lokasi di dalam daerah cakupan nominal yang direncanakan. Sebelum melakukan pengukuran daya dan cakupan, perlu dibuat *Link Budget* yang bertujuan untuk menguji kelayakan sistem sesuai spesifikasi pemancar dan alat ukur yang digunakan.



Gambar 2.26 Propagasi Kualitas Cakupan pada Sistem DVB-T

Sumber: Wireless Communcation, Principles and Practice, Pretice Hall, Inc, Hal.171

2.12.2.1 Metode kualitas cakupan (umum)

Pengukuran daya dilakukan pada lokasi yang acak, dengan jarak yang bervariasi dari antenna pemancar. Pengambilan *sample* merata dari jarak yang terdekat sampai yang terjauh, sesuai dengan radius daerah cakupan yang direncanakan. Pada setiap lokasi pengukuran, dilakukan pencatatan daya terukur dan koordinat lokasi dari GPS. Koordinat ini nantinya digunakan dalam perhitungan jarak dari pemancar.

Tabel. 2.3 Contoh perhitungan *Link budget*

ERP (Pt + Gt)	75.00 dBm (+)
Redaman Lintasan	136.33 dB (-)
Model Hata,urban	
575 Mhz	
Ht = 100 m	
Hr = 1,5 m	
D = 5 km	
Gain antenna penerima (Gr)	5.00 dB (+)
Sensitivitas alat ukur	-60.00 dB (-)
Margin daya	3.67 dB



Estimasi persentasi wilayah cakupan dapat dilakukan dengan menerapkan asumsi variasi *shadowing* dalam dB yang berdistribusi *Gaussian* dengan simpang baku σ yang diperoleh dari pengukuran. Secara umum, jika ditentukan daya minimum yang dianggap memberikan kualitas sinyal memadai P_{\min} , maka proporsi daerah tercakup adalah rasio antara luas daerah dimana daya tarima lebih besar dari P_{\min} terhadap luas daerah cakupan nominal. Jika diasumsikan daya menurun terhadap jarak dengan eksponensial n dan simpang baku variasi *shadowing* σ maka proporsi cakupan dirumuskan sebagai berikut:

$$F = \frac{1}{2} + \frac{1}{2} \exp\left(\frac{1}{b^2}\right) \left(1 - \operatorname{erf}\left(\frac{1}{b}\right)\right) \quad (2.54)$$

Dengan $b = 10 n \log_{10} e / \sigma \sqrt{2}$ dan asumsi tambahan bahwa pada tepi sel terdapat margin daya nol [Jakes,1994]

Hasil yang diperoleh dari perhitungan *link budget* adalah *Maximum Allowable Path Loss* (MAPL) yang sangat menentukan untuk perhitungan jarak atau radius sel dalam menentukan *coverage area*. Radius sel dapat ditentukan untuk tiga morfologi daerah yang berbeda yaitu *urban*, *suburban*, dan *rural* dengan menggunakan model propagasi antara lain Okumura-Hata, COST-231-Hata dan Walfisch Ikegami. Didalam perhitungan *power link budget* kita juga dapat mengetahui level daya terima (*Receive Signal Level*) yang diterima oleh penerima.

2.12.2.2 Metode Okumura-Hata

Model Okumura Hata merupakan model yang disempurnakan dari Okumura model, valid untuk lingkungan *quasi smooth terrain* dan tidak mengakomodasi perubahan *radio path profile* yang cepat. Selain itu model ini hanya cocok untuk makro sel (radius sel lebih dari 1km).

Dimana :

$$150 \leq f_c \leq 1500 \text{ MHz}, 30 \leq h_b \leq 200 \text{ km}, 1 \leq d \leq 20 \text{ km} \quad (2.55)$$

Median path loss, $L_{\text{propagasi urban}}$ adalah :

$$L_U = 69.55 + 26.16 \log f_c - 13.82 \log h_b - a(h_m) + (44.9 - 6.55 \log h_b) \log d \quad (2.56)$$

- Untuk *small to medium sized city*, faktor koreksi tinggi antenna MS ($1 \leq h_m \leq 10$ m) adalah :

$$a(h_m) = (1.1 \log f_c - 0.7) h_m - (1.56 \log f_c - 0.8) \quad (2.57)$$

- Untuk *large city*

$$a(h_m) = 8.29 (\log 1.54 h_m)^2 - 1.1 \text{ dB for } f_c \leq 300 \text{ MHz}$$

$$a(h_m) = 8.29 (\log 1.54 h_m)^2 - 1.1 \text{ dB for } f_c \geq 300 \text{ MHz} \quad (2.58)$$

Sedangkan median path loss, $L_{\text{propagasi suburban}}$ adalah :

$$L_{SU} = L_{\text{propagasi urban}} - 2 [\log (f_c / 28)]^2 - 5.4 \quad (2.59)$$

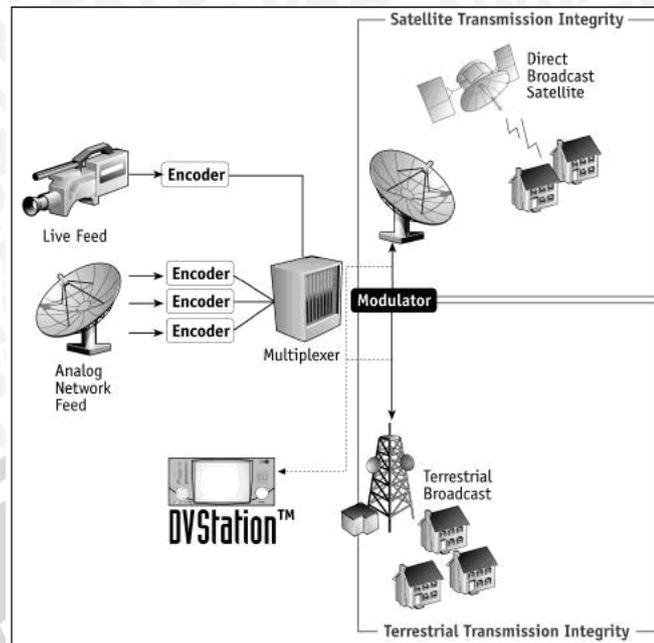
Dan median path loss, $L_{\text{propagasi rural open area}}$ adalah :

$$L_o = L_U - 4.78 (\log f_c)^2 - 18.33 \log f_c - 40.98 \quad (2.60)$$

2.12.3 Pengukuran *Delay Spread*

Panjang lintasan dan perlambatan gelombang yang berbeda-beda mengakibatkan sinyal-sinyal *multipath* sampai pada penerima dengan variasi waktu tunda. Sebuah impuls yang dikirimkan oleh pemancar akan diterima oleh penerima bukan sebuah impuls melainkan sebuah pulsa dengan lebar penyebaran.

Delay spread dapat menimbulkan interferensi antar simbol, karena setiap simbol akan saling bertumbukan dengan simbol sebelum dan sesudahnya. Level interferensi antar simbol ini ditentukan oleh kecepatan transmisi bit. Jadi kecepatan transmisi bit atau simbol pada sistem komunikasi bergerak digital dibatasi oleh *delay spread*.



Gambar 2.27 Propagasi Pengukuran *delay spread* pada Sistem DVB-T

Sumber: Wireless Communcation, Principles and Practice, Pretice Hall, Inc, Hal.168

Model pengukuran *delay spread* adalah dengan melakukan pengukuran menggunakan alat ukur pixelmetrik antara antenna pemancar dan *user* pada kondisi propagasi *Non Line Of Sight* (NLOS), untuk menghasilkan nilai *delay* serta redaman.

Parameter *delay spread* yang sering ditulis dalam bentuk *root mean square* (RMS) adalah *rms delay spread* yang diberikan pada persamaan berikut ini:

$$\sigma_{\tau} = \sqrt{\overline{\tau^2} - (\overline{\tau})^2} \tag{2.61}$$

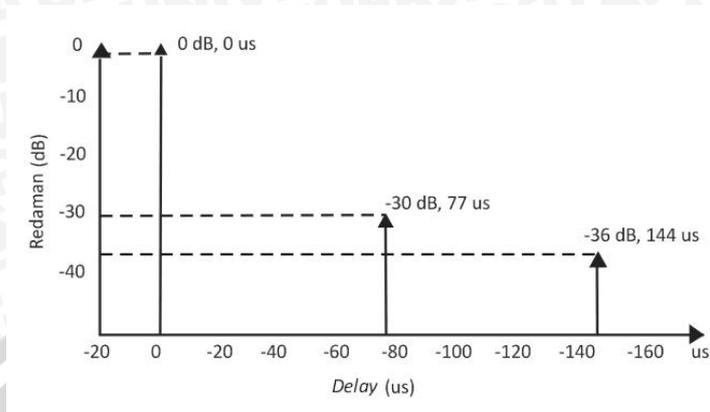
Dengan τ merupakan *mean excess delay* yang dinyatakan dengan persamaan:

$$\overline{\tau} = \frac{\sum_k P(\tau_k)\tau_k}{\sum_k P(\tau_k)} \tag{2.62}$$

Dan,

$$\overline{\tau^2} = \frac{\sum_k P(\tau_k)\tau_k^2}{\sum_k P(\tau_k)} \tag{2.63}$$

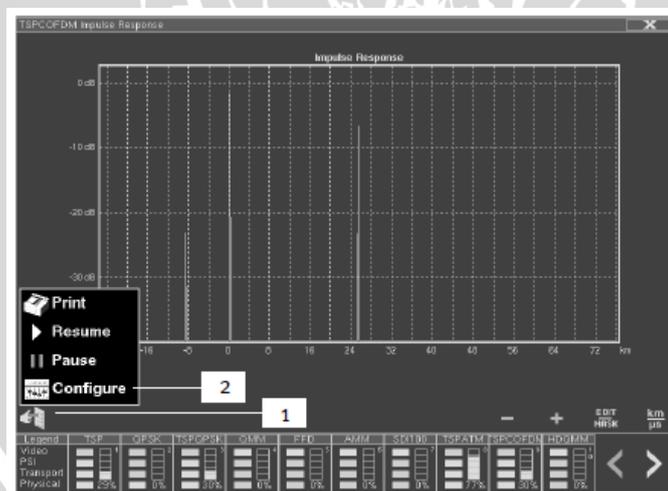
Contoh *Mean excess delay*, *rms delay spread*, dari suatu kanal *multipath* pada hasil pengukuran di RSAB Jakarta. Seperti terlihat pada *profile* dibawah ini:



Gambar 2.28 *Path delay* dan redaman pada Pengukuran *Delay Spread*

Sumber: Wireless Commnucation, Principles and Practice, Pretice Hall, Inc, Hal.211

Pada pengukuran *delay spread* ini dilakukan untuk mengetahui nilai *delay spread* atau waktu tunda sinyal. Hal tersebut dapat dilakukan dengan mengukur nilai *respon impulse* sinyal. Sinyal yang terkena *delay* ditandai dengan munculnya *echo* pada jarak tertentu.



Gambar 2.29 *Path Redaman Fungsi Jarak* dari Pengukuran *Delay Spread*

Sumber: <http://www.pixelmatrix.com>

BAB III METODOLOGI PENELITIAN

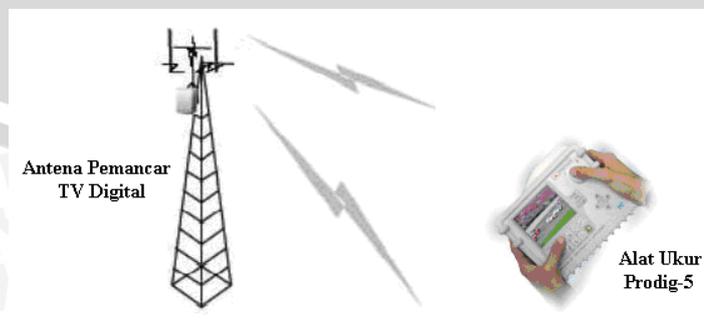
Pada bab ini akan dibahas mengenai metodologi pengukuran meliputi, pengukuran *path loss*, pengukuran kualitas cakupan dan pengukuran *delay spread*. Dan metodologi pemodelan sistem penyiaran Televisi Digital Terrestrial (DVB-T) yang diolah menggunakan *software Matlab 7.0.1* sebagai dasar analisis. Adapun langkah-langkah dari pengukuran sistem adalah sebagai berikut:

3.1 Langkah-Langkah Pengukuran Parameter Sistem

3.1.1 Pengukuran *Path Loss*

Pengukuran *path loss* merupakan pengukuran yang dilakukan untuk melihat sebuah redaman lintasan yang digunakan untuk sinyal dengan rambatan *Line of Sight* (LOS) maupun *Non-Line of Sight* (N-LOS), alat ukur yang digunakan adalah Prodig-5 untuk menghasilkan nilai C/N dan VBER. Langkah-langkah pengukuran *path loss* adalah sebagai berikut:

1. Menyiapkan alat ukur prodig-5.
2. Melakukan pengesetan alat ukur, sesuai dengan *setting* antenna pemancar yaitu: Frekuensi kerja 578 Mhz, OFDM carrier 8k, Modulasi 16 QAM, *code rate* $\frac{3}{4}$ dan *guard interval* 1/16.
3. Menentukan titik pengukuran *path loss*, yaitu pada *main lobe* antenna pemancar baik pada kondisi *Line of Sight* (LOS) maupun *Non-Line of Sight* (N-LOS).
4. Mengarahkan alat ukur prodig-5 pada antenna pemancar DVB-T yang diletakkan setinggi 100 meter pada antenna tower TVRI.



Gambar 3.1 Pengukuran *Path Loss* pada sistem DVB-T

Sumber: PROMAX, user's manual prodig-5, 2005 hal. 15

5. Memberikan power sinyal pada alat ukur prodig-5 pada setiap titik pengukuran yang berbeda dengan power sinyal maksimal 130 dB μ V.
6. Melakukan 32 kali pengukuran pada daerah dengan kondisi *Line of Sight* (LOS) maupun *Non-Line of Sight* (N-LOS).
7. Mencatat hasil pengukuran berupa nilai C/N dan VBER yang bervariasi sebagai bahan analisis.

Pada pengukuran *path loss* dilakukan dengan cara memilih lokasi uji secara acak pada daerah *main lobe* antenna pemancar. Hasil pengukuran *path loss* yang digunakan pada skripsi ini hanya nilai C/N dan nilai VBER, untuk melihat variasi nilai VBER sistem dalam skala besar, yaitu titik dengan kondisi *Line of Sight* (LOS) yaitu titik yang tidak terhalang oleh *obstacle*, maupun titik dengan kondisi *Non-Line of Sight* (NLOS) yang merupakan titik yang banyak terhalang oleh *obstacle*.

3.1.2 Pengukuran Kualitas Cakupan

Pengukuran kualitas cakupan merupakan pengukuran yang dilakukan untuk melihat kualitas penyiaran sistem DVB-T baik gambar maupun suara. Dan pengukuran kualitas cakupan ini dibedakan pada daerah dengan kondisi rambatan *Line of Sight* (LOS) dan *Non-Line of Sight* (N-LOS) untuk melihat variasi nilai VBER pada skala kecil, alat ukur yang digunakan adalah Prodig-5 untuk menghasilkan nilai C/N dan VBER. Langkah-langkah pengukuran *path loss* adalah sebagai berikut:

1. Menyiapkan alat ukur prodig-5.
2. Melakukan pengesetan alat ukur, sesuai dengan *setting* antenna pemancar yaitu: Frekuensi kerja 578 Mhz, OFDM carrier 8k, Modulasi 16 QAM, *code rate* $\frac{3}{4}$ dan *guard interval* 1/16.
3. Menentukan titik pengukuran *path loss*, yaitu pada *main lobe* antenna pemancar yang dibedakan pada kondisi *Line of Sight* (LOS) dan *Non-Line of Sight* (N-LOS) dengan area pengukuran 100 meter x 100 meter.
4. Mengarahkan alat ukur prodig-5 pada antenna pemancar DVB-T yang diletakkan setinggi 100 meter pada antenna tower TVRI.

5. Memberikan power sinyal pada alat ukur prodig-5 pada setiap titik pengukuran yang berbeda dengan power sinyal maksimal 130 dB μ V.
6. Melakukan masing-masing 20 kali pengukuran pada daerah dengan kondisi *Line of Sight* (LOS) dan *Non-Line of Sight* (N-LOS).
7. Mencatat hasil pengukuran berupa nilai C/N dan VBER yang bervariasi pada kondisi *Line of Sight* (LOS) dan *Non-Line of Sight* (N-LOS) sebagai bahan analisis.

3.1.3 Pengukuran *Delay Spread*

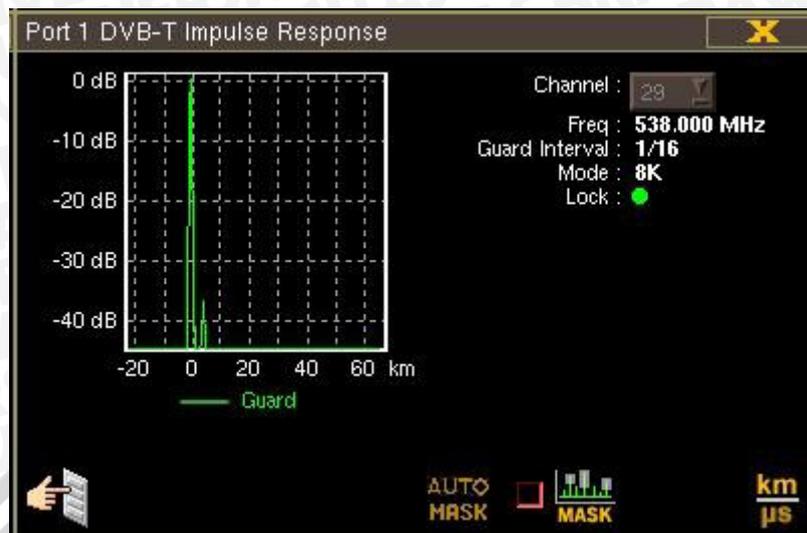
Pengukuran *delay spread* dilakukan untuk mengetahui nilai *delay spread* atau waktu tunda sinyal. Hal tersebut dapat dilakukan dengan mengukur nilai *respon impulse* sinyal. Sinyal yang terkena *delay* ditandai dengan munculnya *echo* pada jarak tertentu, berikut merupakan langkah-langkah pengukuran *delay spread* menggunakan alat ukur pixelmetrix:

1. Menyiapkan alat ukur pixelmetrix.
2. Melakukan pengesetan alat ukur, sesuai dengan *setting* antena pemancar yaitu: Frekuensi kerja 578 Mhz, OFDM carrier 8k, Modulasi 16 QAM, *code rate* $\frac{3}{4}$ dan *guard interval* 1/16.
3. Menentukan 5 titik pengukuran yang dianggap mempunyai redaman serta *delay* sinyal dan membaginya menjadi 3 *path* berdasarkan jarak pengukuran.
4. Menampilkan *path delay profile* pada alat ukur pixelmatrix dengan cara:
 - a) Dari layar *Impulse Response*, pilih menu.
 - b) Pilih Konfigurasi.

Catatan: EDIT / MASK ikon di kanan bawah layar digunakan untuk mendefinisikan pengaturan posisi. Km / micros ikon di kanan bawah layar digunakan untuk mengaktifkan tampilan grafik *respon impuls* sesuai dengan jarak dan waktu masing-masing.

- c) Pilih Sinyal *Config*.
- d) Pilih frekuensi saluran yang diperlukan dari Rencana dan Channel kotak *drop-down*.
- e) Set VHF dan UHF nilai menggunakan *slider*.

f) Pilih OK.



Gambar 3.2 Tampilan *path delay profile* pada Alat Ukur Pixelmatrix

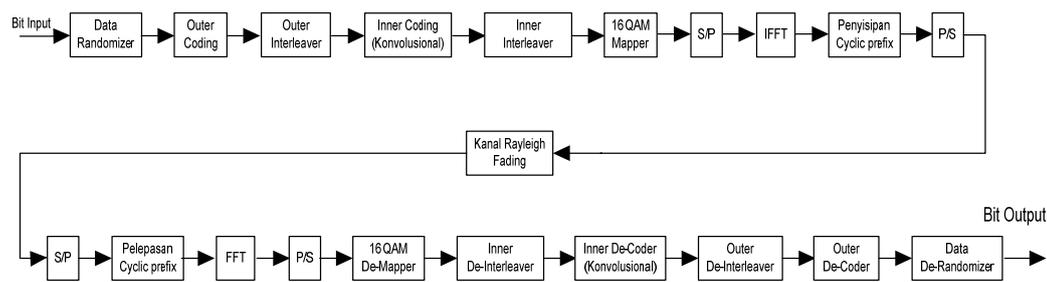
Sumber: <http://www.pixelmatrix.com>

5. Mencatat hasil pengukuran redaman dan *delay* terburuk pada setiap path, yang digunakan sebagai acuan pemodelan sistem yang diolah menggunakan *software* Matlab 7.0.

3.2 Pemodelan Sistem

Pemodelan sistem mengacu pada standar *European Standart Telecommunication Institute* (ETSI) 300 744. Menggunakan *software* Matlab versi 7.0.1. untuk proses pemodelannya. Data pengukuran merupakan nilai BER setelah proses viterbi *decoding* dan sebelum masuk RS *decoder*, maka nilai BER pemodelan yang diambil adalah BER setelah blok *outer Deinterleaving* merupakan tahap sebelum masuk ke RS *decoder*.

Adapun blok diagram sistem DVB-T (RF/*Transmission system*) yang digunakan dalam pemodelan dapat dilihat di bawah ini:



Gambar 3.3 Blok Diagram Pemodelan Sistem DVB-T

(Sumber: Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, Februari 2010)

3.2.1 Input Sistem

Informasi input sistem berupa bit-bit random 0 dan 1 yang dihasilkan oleh generator acak. Maka untuk menghasilkan suatu urutan *binary* acak berupa urutan bit serial, menggunakan perintah *randint* ($n, m, range, state$) dapat dilihat pada lampiran listing program. Perintah *state* digunakan agar dapat membedakan bit input. Sehingga percobaan yang satu dengan yang lain berifat *independent*. Bit yang digunakan dalam pemodelan berjumlah 1.000.000 bit. Urutan bit tersebut diproses secara serial dengan memecah data sebesar 1.000.000 bit tersebut menjadi urutan data sepanjang 200.000 sebanyak 5 buah.

3.2.2 Proses Randomisasi

Proses *randomization* dilakukan oleh *Pseudo Random Binary Sequence* (PRBS) yang bergenerator polynomial $1+X^{14}$. PRBS tersebut akan diinisialisasi setiap 8 transport paket dengan memasukan bit inialisasi “100101010000000”, di mana setiap paket berisi 188 byte data dari sinyal input atau sebanyak 188×8 bit biner, sehingga total data yaitu 12.032 bit biner.

Pada proses ini tiap bit informasi akan di-*xor*-kan dengan hasil *xor* antara bit 15 dan 16 dari PRBS. Untuk proses di sisi decoder data akan di-*xor*-kan kembali untuk memperoleh kembali data aslinya.

3.2.3 Outer coding

Reed-Solomon (204,188), dengan $n = 204$ dan $k = 188$ yang merupakan pendekan dari RS (255,239) digunakan sebagai *outer coding* sesuai dengan

yang disebutkan pada standar ETSI 300744. Kode RS (204,188) merupakan pengkodean simbol *non-biner*, di mana setiap 1 simbol terdiri dari 8 bit biner.

Pada pengkodean ini, bit-bit biner yang berasal dari proses randomisasi diambil per 8 bit (1 byte) untuk dijadikan 1 simbol. Setiap 188 simbol yang terbentuk akan dikodekan menjadi 204 simbol untuk setiap 1 *codeword* dengan menyisipkan 16 simbol *parity*. Sebelum dikodekan simbol-simbol yang terbentuk diubah dahulu ke dalam *Galois field* menggunakan fungsi *gf*. Proses pengkodean tersebut dilakukan dengan menggunakan fungsi *rsenc* pada sisi *encoder* dan *rsdec* pada sisi *decoder*.

3.2.4 Outer interleaver

Pada *outer interleaver* digunakan *convolutional interleaver* yang menginterleave data per *byte*. *Interleaver* yang digunakan mempunyai *branch* (*N*) sebanyak 12 masing-masing *shift register* (*J*) memuat data sebanyak 17 *byte*.

Pada *convolutional interleaver* data ditulis setiap 17 *byte* menggunakan fungsi *muxintrlv* dan *delay* yang digunakan mulai dari *branch* 0 sampai $N-1 = 11$ yaitu 0, 17, 34, 51, 68, 85, 102, 119, 136, 153, 170, dan 187. Pada sisi decoder digunakan fungsi *muxintrlv* untuk memperoleh kembali sinyal yang telah diinterleave.

3.2.5 Inner Coding

Kode konvolusional merupakan pengkodean yang digunakan untuk *inner coding*. Sistem DVB-T menggunakan kode konvolusional dengan kode induk 1/2. Sistem ini memungkinkan perubahan *code rate* dengan cara melakukan *puncturisasi* kode induk.

Parameter yang perlu dimasukkan yaitu *constraint length* = 7 dan generator *polynomial* untuk masing-masing output dengan menghubungkannya ke bentuk *octal*, sehingga untuk generator pertama $1111001 = 171_{\text{oct}}$ dan untuk generator kedua $1011011 = 133_{\text{oct}}$.

Untuk pengkodean ini, data yang berasal dari keluaran *inner interleaver* diubah dahulu menjadi bit-bit biner, di mana setiap 1 *byte* akan diubah menjadi 8 bit biner menggunakan fungsi *de2bi*. Setelah data menjadi bit-bit biner maka data tersebut siap untuk dikodekan kembali menggunakan kode konvolusional di

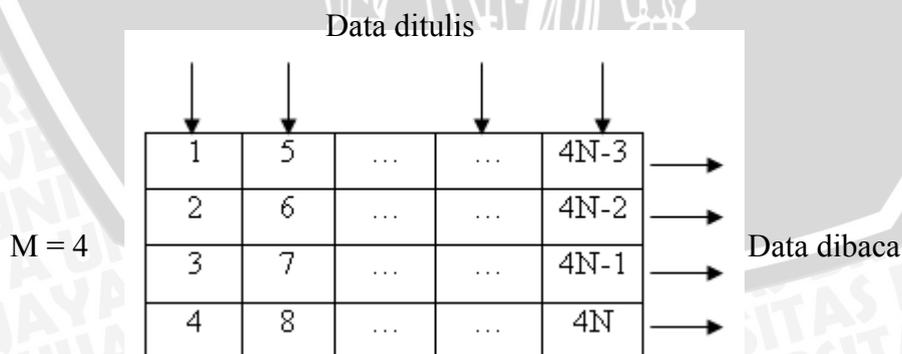
mana setiap 1 bit input akan dikodekan menjadi 2 bit output. Proses pengkodean ini dilakukan dengan menggunakan fungsi *convenc* untuk bagian enkoder dan fungsi *vitdec* pada sisi decoder yang berarti bahwa proses pengkodeannya menggunakan algoritma Viterbi, yang pada dasarnya pengkodean dilakukan dengan mencari jarak terpendek dari jumlah *hamming distance* yang menuju pada suatu *state*.

Pemodelan sistem menggunakan *code rate* 1/2 yang merupakan *code rate* induk dari sistem DVB-T, serta *code rate* 3/4 yang merupakan hasil *puncture* dari *code rate* 1/2.

Proses *puncture* kode induk mengikuti standar ETSI 300 744, dimana bit yang ditransmisikan adalah $X_1Y_1Y_2Y_3$. Pada sisi pemancar, maka bit X_2 dan X_3 dihilangkan sehingga dari tiap 6 bit hasil keluran *code rate* 1/2, hanya 4 bit saja yang ditransmisikan. Sedangkan pada sisi penerima, maka bit yang dihilangkan tadi akan diganti dengan bit 0, untuk kemudian dilakukan proses *viterbi decoding*.

3.2.6 Inner Interleaver

Untuk *inner interleaver* digunakan *blok interleaver*. Pada proses ini bit-bit output yang berasal dari kode konvolusional ditulis secara per kolom dan dibaca per baris. Pada sisi dekoder dilakukan proses yang berkebalikan yaitu menulis bit-bit input per baris dan mengeluarkan bit-bit tersebut per kolom. Adapun banyaknya baris yang digunakan yaitu sebanyak 4 baris dan dapat diilustrasikan pada Gambar 3.4.



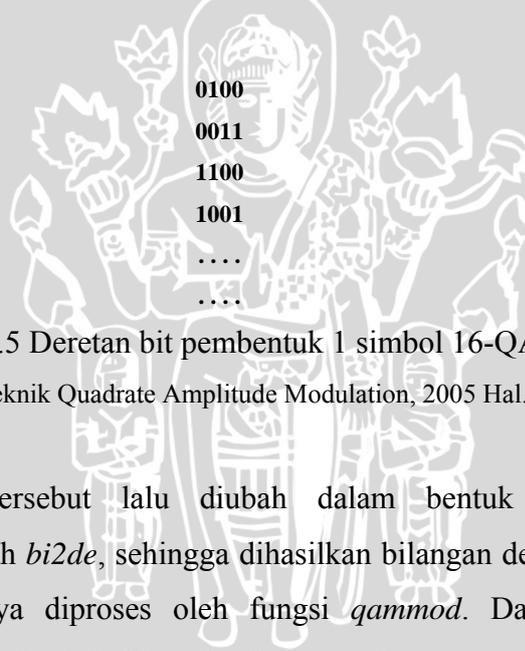
Gambar 3.4 Blok *Interleaver* dengan banyaknya baris (M) = 4

Sumber: *Wireless Communicaton, Principles and Practice*, Prentice Hall, 2002 hal. 115

Proses pada blok sistem ini dilakukan dengan menggunakan perintah *reshape* dan *vec2mat*.

3.2.7 Mapping 16 QAM

Untuk proses *mapping* dan *demapping* digunakan 16 QAM baik pada bagian modulator maupun pada demodulator. Pada tahap ini, setiap 4 bit input akan dipetakan menjadi 1 simbol kompleks yaitu $I+jQ$, di mana I (*Inphase*) merupakan bagian real dan Q (*Quadrature*) merupakan bagian imajiner. Proses *mapping* dan *demapping* menggunakan fungsi *qammod* pada sisi *mapper* dan *qamdemod* pada sisi *de-mapper* yang telah tersedia pada *Matlab* versi 7.0.1. Proses *mapping* ini dilakukan dengan cara menyusun bit terlebih dahulu. Karena pada 16 QAM satu simbol berisi 4 buah bit, maka deretan bit disusun menjadi urutan bit seperti pada Gambar 3.5 berikut.

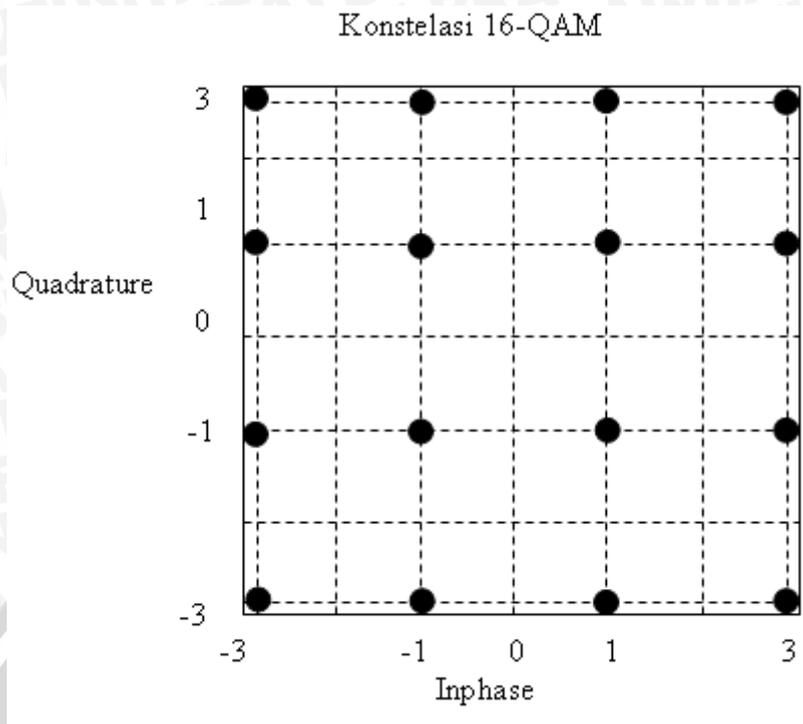


Gambar 3.5 Deretan bit pembentuk 1 simbol 16-QAM

Sumber: Teknik Quadrate Amplitude Modulation, 2005 Hal. 22

Deretan bit tersebut lalu diubah dalam bentuk desimal dengan menggunakan perintah *bi2de*, sehingga dihasilkan bilangan desimal antara 0-15. Bilangan ini nantinya diproses oleh fungsi *qammod*. Dan pada penerima dilakukan proses yang berkebalikan, hasil proses *demapping* yang masih berupa bilangan desimal akan diubah kembali menjadi deretan biner dengan menggunakan fungsi *bi2de*.

Hasil *mapper* yang sesungguhnya merupakan gabungan dari sinyal real dan sinyal imajiner tersebut. Dimana setiap kombinasi sinyal real dan sinyal imajiner yang berupa simbol kompleks dalam bentuk $I+jQ$ akan dipetakan dalam diagram konstelasi 16 QAM seperti Gambar 3.6 berikut.

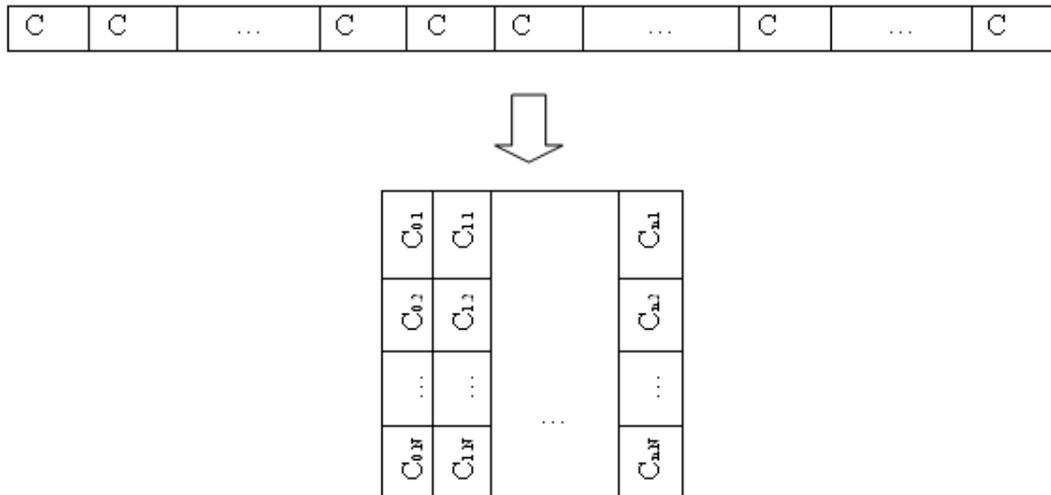


Gambar 3.6 Diagram konstelasi 16 QAM

Sumber: Teknik Quadrature Amplitude Modulation, 2005 Hal. 43

3.2.8 Serial to Parallel

Input dari *Serial to Parallel Converter* adalah sederetan simbol yang telah dilevelisasi oleh *mapper* 16 QAM. Pada pemodelan ini, sesuai standar ETSI EN 300 744 untuk mode 8k, *subcarrier* yang digunakan sebanyak 8192 *subcarrier*, tetapi hanya 6817 *subcarrier* saja yang berisi data, sedangkan *subcarrier* yang lain diisi nol (*zero padding*). Proses ini dilakukan dengan menggunakan perintah *reshape*. Proses *serial to parallel* dapat diilustrasikan pada Gambar 3.7. Dari gambar 3.7 tersebut dapat dilihat data diubah menjadi bentuk paralel dengan 8192 baris dan jumlah kolom sebanyak panjang data dibagi dengan 8192.



Gambar 3.7 Proses *Serial to Parallel*

Sumber: Teknik Quadrature Amplitude Modulation, 2005 Hal. 61

3.3.9 Inverse Fast Fourier Transform (IFFT) dan Fast Fourier Transform (FFT)

Pada bagian pemancar OFDM dapat diimplementasikan dengan IFFT dan pada bagian penerima diimplementasikan dengan FFT. Pada fungsi IFFT sinyal dalam domain frekuensi akan ditransformasikan ke domain waktu dan fungsi FFT digunakan untuk memisahkan kembali sinyal informasi yang dikirim.

Proses ini dilakukan dengan menggunakan fungsi *ifft* pada sisi pemancar dan *fft* pada sisi penerima. Kedua fungsi tersebut melakukan operasi data per kolom. Dimana pada IFFT, digunakan 131072 titik IFFT. Sehingga setiap simbol disampling sebanyak 16 kali.

3.3.10 Guard Interval

Untuk memecahkan masalah *Inter Symbol Interference* (ISI) dilakukan penyisipan *guard interval* secara periodik pada bagian awal pada tiap simbol OFDM. *Guard Interval* dipergunakan adalah 1/32 dan 1/16 dari total 1 simbol OFDM, dimana 1 simbol OFDM terdiri dari 131072 titik. Sehingga besarnya *guard interval* bervariasi antara 4096 titik hingga 8192 titik. Penyisipan *guard interval* untuk 1 simbol OFDM dapat digambarkan pada Gambar 3.8.

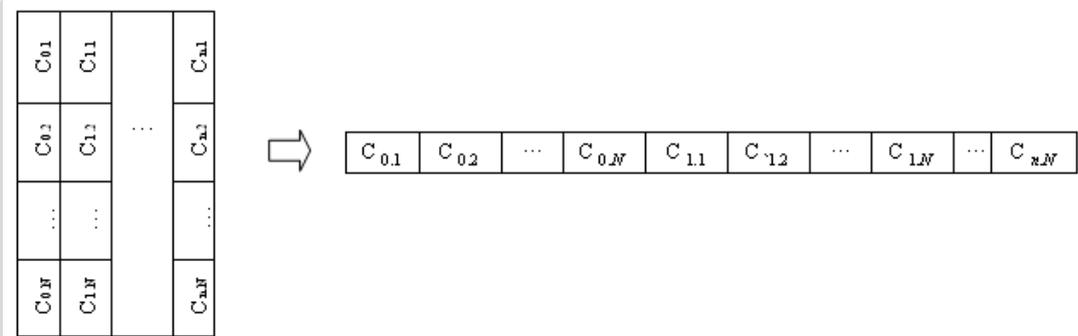


Gambar 3.8 Penyisipan *Guard Interval* 1/16 pada Simbol OFDM

Sumber: Henrik Schulze, Theory And Application of OFDM, John Wiley&Sons, 2005 hal. 160

2.2.11 Parallel to Serial

Setelah proses penyisipan *guard interval*, maka keseluruhan simbol yang dibentuk akan di ubah ke bentuk serial untuk ditransmisikan. Proses *parallel to serial*, ditunjukkan oleh Gambar 3.9.



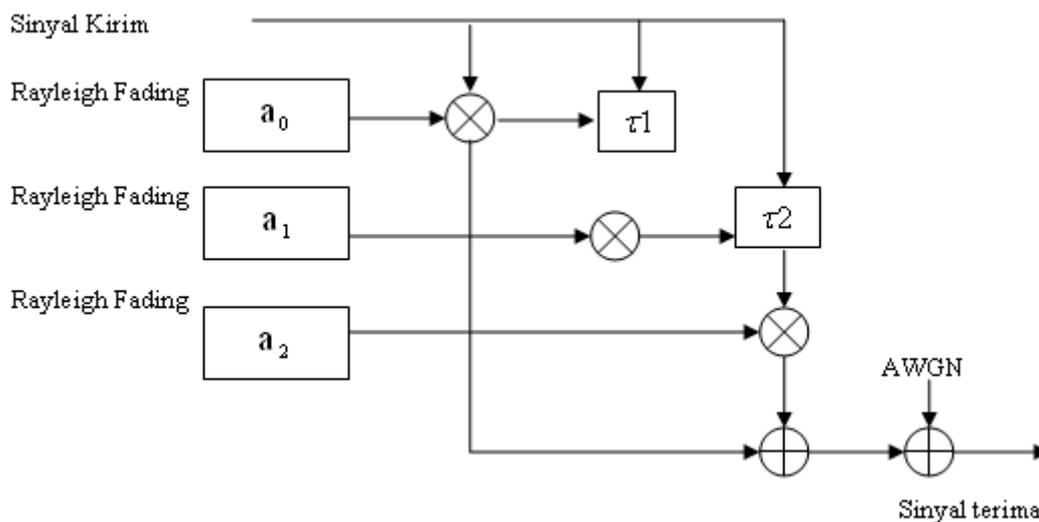
Gambar 3.9 Proses *Parallel to Serial*

Sumber: Teknik Quadrature Amplitude Modulation, 2005 Hal. 66

Proses ini menggunakan fungsi *reshape* yang telah tersedia di *Matlab* 7.0.1. Pada penerima nantinya, akan dilakukan proses yang berkebalikan yaitu *serial to parallel* seperti yang telah dijelaskan pada sub bab 3.2.8 sebelumnya.

3.2.12 Kanal Propagasi

Kanal diasumsikan memiliki 3 buah *path*. *Path* pertama tidak ada *delay* dengan redaman 0 dB, *path* kedua mempunyai *delay* sebesar 77 μ s dan redaman -30 dB dan *path* ketiga mempunyai *delay* sebesar 147 μ s dengan redaman -36 dB seperti yang dapat dilihat pada Gambar 3.10. Asumsi tersebut berdasarkan pada *respons implus* terburuk dari hasil pengukuran, yaitu pada lokasi RSAB Jakarta.



Gambar 3.10 Pemodelan Kanal

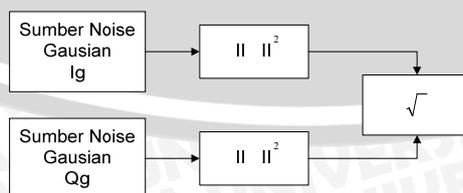
Sumber: *Wireless Communicaton, Principles and Practice*, Prentice Hall, 2002 hal. 156

3.2.12.1 Kanal Multipath Terdistribusi Rayleigh

Kanal juga dimodelkan sebagai *fading* yang terdistribusi secara *Rayleigh*. *Envelope* sinyal *fading* yang dibangkitkan merupakan proses kompleks *gaussian* yang mempunyai bagian real yang *independent* dengan bagian imajiner-nya. Metode untuk membangkitkan *envelope* sinyal *fading* ini dapat diilustrasikan sebagai berikut :

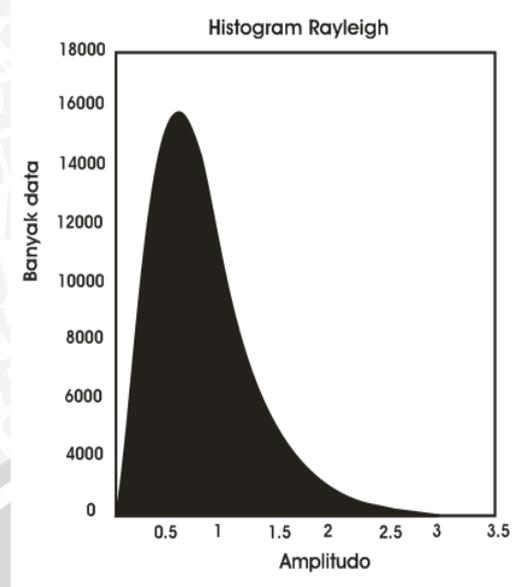
$I_g(n)$ dan $Q_g(n)$ adalah komponen *inphase* dan *quadrature* dari proses kompleks *Gaussian*, dimana pembangkitan *envelope fading* adalah yaitu :

- $I_g(n) = \text{randn}(1, \text{panjang sinyal input})$
- $Q_g(n) = \text{randn}(1, \text{panjang sinyal input})$
- $Envelope = \sqrt{I_g(n)^2 + Q_g(n)^2}$



Gambar 3.11 Pembangkitan Rayleigh Fading

Sumber: Rizka Fadillah, *Studi Tentang Teknik OFDM untuk Penyiaran TV Digital Terrestrial (DVB-T)*, Tugas Akhir, 2006

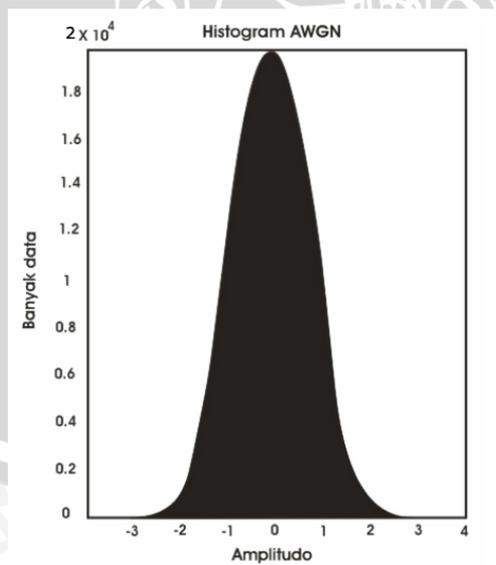


Gambar 3.12 Histogram Distribusi Rayleigh

Sumber: Rizka Fadillah, *Studi Tentang Teknik OFDM untuk Penyiaran TV Digital Terrestrial (DVB-T)*, Tugas Akhir, 2006

3.2.12.2 Kanal Additive White Gaussian Noise (AWGN)

AWGN merupakan *thermal noise* yang terdistribusi normal dengan nilai rata-rata nol. Pada pemodelan, fungsi pada *Matlab* yang digunakan untuk membangkitkan AWGN ini adalah *awgn*. Gambar 3.13 adalah histogram dari AWGN.



Gambar 3.13 Histogram AWGN

Sumber: Rizka Fadillah, *Studi Tentang Teknik OFDM untuk Penyiaran TV Digital Terrestrial (DVB-T)*, Tugas Akhir, 2006

Histogram pada gambar 3.10 menggambarkan banyaknya jumlah data terhadap *amplitude* tertentu. Pada pemodelan, parameter yang mempengaruhi fungsi dari *awgn* pada *Matlab* adalah SNR. SNR yang digunakan dinyatakan dalam bentuk *decibel* (dB). Jadi jika dituliskan nilai SNR adalah 0 dB, maka maksudnya adalah perbandingan energi sinyal dan energi *noise* adalah satu.

Sesuai pada pemodelan kanal pada gambar 3.10 AWGN dibangkitkan setelah sinyal *multipath* tersebut dijumlahkan. Dengan menggunakan asumsi tersebut, maka kanal AWGN terjadi pada penerima.

3.3 Perhitungan BER

BER dihitung dengan menggunakan metode *Monte Carlo*, yaitu dengan membandingkan antara deretan *bit* pada pengirim dengan deretan bit yang dideteksi pada sisi penerima, kemudian jumlah *bit* yang salah dibagi dengan jumlah *bit* yang dibangkitkan. Pada pemodelan *Matlab* perhitungan BER dapat dilakukan dengan menggunakan fungsi *biterr*. Dan pada pemodelan ini, yang dibandingkan adalah *input* dari blok *outer interleaver* dan *output* dari blok *outer de-interleaver*.



BAB IV

ANALISA DATA HASIL PENGUKURAN DAN PEMODELAN

Pada bab ini, akan dianalisis data hasil pengukuran sistem dengan data hasil pemodelan yang telah dilakukan di Jakarta. Parameter yang dianalisis adalah nilai *Bit Error Rate* (BER) setelah proses *viterbi decoding* atau disebut pula dengan VBER.

4.1 Data Hasil Pengukuran

Seperti yang telah dijelaskan pada bab 3, bahwa pada skripsi ini terdapat tiga macam data hasil pengukuran yang digunakan, yaitu data hasil pengukuran *path loss*, data hasil pengukuran kualitas cakupan dan data hasil pengukuran *delay spread*. Yang dimaksud dengan nilai VBER, yaitu nilai BER setelah blok *viterbi decoding* pada data hasil pengukuran *path loss* dan data hasil pengukuran kualitas cakupan yang diperoleh dengan menggunakan alat ukur Prodig-5, hasil data tersebut digunakan untuk melakukan evaluasi kinerja sistem pada kondisi propagasi LOS (*Line of Sight*) maupun NLOS (*Non Line of Sight*). Sedangkan data hasil pengukuran *delay spread* yang diperoleh dengan menggunakan alat ukur Pixelmetrix, menghasilkan sebuah *path delay profile* dari sinyal yaitu redaman serta *delay* yang nantinya akan dijadikan sebagai dasar untuk pemodelan sistem. Berikut ini merupakan hasil dari ketiga data pengukuran tersebut.

4.1.1 Data Hasil Pengukuran *Path Loss*

Berikut ini merupakan data hasil pengukuran *path loss*. Dari data hasil pengukuran ini, diperoleh nilai C/N dan VBER yang bervariasi pada sistem skala besar yaitu kawasan dengan kondisi LOS (*Line of Sight*) dan NLOS (*Non Line of Sight*). Data pengukuran ini diperoleh dari titik-titik lokasi yang berada pada *main lobe* antenna pemancar. Pengukuran ini dilakukan mulai dari Regent Jakarta sampai Pasar Pal Merah dengan keadaan cuaca cerah berawan. Adapun data hasil pengukuran *path loss* dapat dilihat pada tabel di bawah ini:

Tabel 4.1 Data Pengukuran *Path Loss*

No	Level Sinyal Alat Ukur (dB μ V)	C/N (dB)	VBER	Lokasi Titik Pengukuran	Jarak Antena Tx dengan Titik Pengukuran (Km)
1	18,4	0	0,1	Regent Jakarta	3,2
2	19,7	0	0,1	Stasiun Gondangdia	4,5
3	20,4	0	0,1	Belakang Istana Presiden	5,7
4	22,8	0	0,1	Patung Tani	5,1
5	23,5	0	0,1	Jl.Lombok	3,8
6	24	0	0,1	POM Abdul Muiz	4,4
7	25,4	0	0,1	Twin Plasa	3,4
8	25,8	0	0,1	Surapati	3,8
9	27,9	0	0,1	SMA 78 Kemanggisan	3,1
10	29,5	0	0,1	Binus (Jl.anggrek)	2,4
11	59	15,8	1,00E-07	Barito Pasifik	1,8
12	36,7	18,2	0,000082	KH.Mas Mansyur	2,1
13	36,6	18,5	0,000026	Jl.Kebun Pala III	2,3
14	37	19,3	0,000077	Jl.Mess (BTN)	2,3
15	37,4	19,4	0,0012	Jl.Pejompongan	1,0
16	36,9	19,7	0,00044	Kali Cidek (BI)	3,9
17	57,4	19,9	1,00E-07	Jembatan Jati Baru	3,6
18	38,7	20,6	0,0002	Jl.Mess Dalam	2,1
19	38,8	20,7	4,70E-07	JAAC Belakang	2,7
20	39,2	20,9	0,00016	Jl.Ps Tanah Abang	2,9
21	58	21,6	1,00E-07	Hotel Santika	1,7
22	40,6	22,4	0,1	Taman Lawang	3,1
23	40,9	23,7	1,00E-07	Kampg Bali (BPPT)	3,8
24	46,9	23,8	9,00E-07	Atas Jembatan Tomang	4,4
25	47,7	26,5	0,000095	Depan JDC	1,2
26	48,4	28,8	0,00022	Dukuh pinggir	2,1
27	55,1	29,1	0,000037	LAN (Benhil)	1,3
28	52,6	29,3	0,00011	Depan Makam Karet	2,0
29	59,4	38,8	0,1	Senayan (Depan Plasa)	1,3
30	62,9	29,6	0,1	TVRI (Parkiran)	0
31	66,3	28,9	0,1	Gatot Subroto	0,7
32	67,1	23,8	0,1	Pasar Pal Merah	0,7

Sumber : Data pengukuran *Path Loss*, *Preventive Maintenance* TVRI Jakarta, Februari 2010.

Dari tabel 4.1 di atas dapat dijelaskan bahwa, data pengukuran *path loss* diperoleh dengan menggunakan alat ukur Prodig-5 yang menghasilkan nilai C/N dan VBER yang bervariasi dengan memberikan level sinyal pada alat ukur di setiap titik pengukuran.

Dari nilai-nilai di atas, lokasi titik pengukuran serta jarak pengukuran *path loss* dengan antena pemancar sangat berpengaruh terhadap nilai C/N dan VBER yang diperoleh. Level sinyal kurang dari 36,6 dB μ V menghasilkan C/N = 0 dan

VBER = 0,1 hal ini belum memenuhi persyaratan dari QEF (*Quasi Error Free*) yaitu nilai BER maksimum sebesar 2×10^{-4} .

QEF (*Quasi Error Free*) memiliki pengertian bahwa dalam waktu 1 jam, hanya terdapat 1 *event error*, yang berkolerasi pada nilai BER output sistem senilai 1×10^{-11} . QEF tersebut dapat terjadi bila nilai BER setelah proses *viterbi decoding*, yaitu sebelum masuk ke blok Reed-Solomon *decoder* bernilai 2×10^{-4} .

Sedangkan level sinyal antara 36,6 dB μ V sampai dengan 59 dB μ V hampir semua nilai VBER nya memenuhi syarat QEF meskipun ada beberapa nilai VBER yang belum memenuhi syarat yaitu 0,0012, 0,00044 dan 0,1 akan tetapi hal ini tidak mempengaruhi kinerja sistem. Sedangkan level sinyal diatas 59 dB μ V meskipun nilai C/N nya memenuhi syarat dari QEF, namun nilai VBER yang diperoleh yaitu 0,1 masih belum memenuhi syarat dari QEF yaitu nilai BER maksimum sebesar 2×10^{-4} . Hal tersebut dikarenakan oleh jarak pengukuran serta lokasi titik pengukuran yang terletak pada daerah NLOS (*Non Line of Sight*).

Dari pengukuran *path loss* didapatkan dua parameter pengukuran yaitu C/N dan VBER yang dapat dinyatakan dalam persamaan berikut:

Rumus *Carrier to noise ratio*:

$$\bullet \quad \frac{C}{N} = \frac{E_b \cdot m \cdot R_b / m}{N_0 \cdot W} = \frac{E_b \cdot R_b}{N_0 \cdot W}$$

Keterangan:

C = Daya sinyal (dB)

R_b = Kecepatan bit (μ s)

N = Daya noise (dB)

E_b = Energi bit (dB μ V)

W = Bandwidth yang digunakan (MHz)

N_0 = Kerapatan daya noise (Watt/Hz)

- Rumus kecepatan bit

$$R_b = R_s (\log 2.J)$$

R_s = kecepatan simbol ($R_s = 1/T_s$) $T_s = 896\mu$ s

J = tingkat modulasi QAM

Maka: $R_b = 896\mu$ s ($\log 2.J$)
= 430 bps

- Bandwidth yang digunakan = 578 MHz

- $\frac{C}{N} = \frac{E_b R_b}{N_0 W}$ maka: $\frac{C}{N} = \frac{59.430}{N_0 578} = 15,8 \text{ dB}$

$$N_0 = \frac{59.430}{578} = 43,89$$

$$= \frac{43,89}{15,8} = 2,77 \text{ Watt/Hz}$$

- $\frac{C}{N} = \frac{E_b R_b}{N_0 W}$ maka: $\frac{C}{N} = \frac{36,7.430}{N_0 578} = 18,2 \text{ dB}$

$$N_0 = \frac{36,7.430}{578} = 27,3$$

$$= \frac{27,3}{18,2} = 1,5 \text{ Watt/Hz}$$

4.1.2 Data Hasil Pengukuran Kualitas Cakupan

Berikut ini merupakan data hasil pengukuran kualitas cakupan yang dibedakan berdasarkan kondisi lokasi pengukuran. Seperti yang telah dijelaskan, pengukuran ini dibedakan atas kondisi LOS (*Line of Sight*) dan NLOS (*Non Line of Sight*). Tabel berikut merupakan data hasil pengukuran kualitas cakupan dengan kondisi LOS (*Line of Sight*) yang terdiri dari 3 lokasi, yaitu depan Makam Karet, Gedung Parkir BPPT Lantai 8 dan Lantai 21.

Tabel 4.2 Data Pengukuran Kualitas Cakupan Lokasi Depan Makam Karet (LOS jarak 2 km)

No	Level Sinyal Alat Ukur (dB μ V)	C/N (dB)	VBER
1	61	16	1,00E-07
2	58,8	19,1	0,00062
3	57,8	22,6	1,00E-07
4	43,1	24,7	0,00018
5	56,7	25,1	1,00E-07
6	51,1	26,5	0,0000097
7	54,4	26,9	1,00E-07
8	54	7,5	1,00E-07
9	54,5	27,9	1,00E-07
10	53,9	28,6	1,00E-07
11	48,8	29,2	0,0000041
12	48,2	29,5	0,0000097
13	62,4	29,7	1,00E-07

Tabel 4.2 (Lanjutan)

No	Level Sinyal Alat Ukur (dB μ V)	C/N (dB)	VBER
14	49,8	30	0,0000031
15	57	30	2,80E-07
16	56,3	30,3	1,00E-07
17	54,5	30,9	1,60E-07
18	49,4	31,1	0,00023
19	53,2	31,9	1,00E-07
20	54,8	32,2	1,00E-07

Sumber : Data pengukuran Kualitas Cakupan (LOS), *Preventive Maintenance* TVRI Jakarta, Februari 2010.

Tabel 4.3 Data Pengukuran Kualitas Cakupan Lokasi Gedung Parkir BPPT Lantai 8 (kondisi LOS jarak 4 km)

No	Level Sinyal Alat Ukur (dB μ V)	C/N (dB)	VBER
1	59	15,8	1,00E-07
2	57,9	16,8	3,60E-08
3	38,7	19	8,90E-08
4	56,5	19,1	1,00E-07
5	60,9	20,4	1,00E-07
6	40,7	20,6	0,000032
7	53,2	22,9	1,00E-07
8	42,1	23,3	0,00016
9	44,9	23,4	0,000021
10	56,9	23,5	1,00E-07
11	43,9	23,9	1,80E-07
12	43,9	24,2	0,000023
13	61,2	24,7	1,00E-07
14	46,4	24,7	0,00001
15	48,5	24,9	0,0000046
16	50,9	26,1	1,00E-07
17	46,8	26,3	1,00E-07
18	63	26,5	1,00E-07
19	55,4	26,7	1,00E-07
20	47,8	27,8	0,000021

Sumber : Data pengukuran Kualitas Cakupan (LOS), *Preventive Maintenance* TVRI Jakarta, Februari 2010.

Tabel 4.4 Data Pengukuran Kualitas Cakupan Lokasi Gedung BPPT Lantai 21 (kondisi LOS jarak 4 km)

No	Level Sinyal Alat Ukur (dB μ V)	C/N (dB)	VBER
1	55,6	13,6	6,80E-07
2	37,6	19,2	0,000033
3	56	21	1,00E-07
4	52,7	21,3	1,00E-07
5	39,8	22,1	2,70E-07
6	43,2	23,2	1,00E-07
7	44,4	24	0,000024
8	42,9	24,1	1,00E-07
9	42,8	24,4	3,10E-07
10	50,6	24,4	1,00E-07
11	43,4	25,3	1,00E-07
12	44,6	25,3	1,00E-07
13	46,7	26,3	1,00E-07
14	48,1	26,3	1,00E-07
15	48,2	26,9	1,00E-07
16	53,2	29,1	1,00E-07

Sumber : Data pengukuran Kualitas Cakupan (LOS), *Preventive Maintenance* TVRI Jakarta, Februari 2010

Sedangkan berikut ini merupakan data hasil pengukuran kualitas cakupan dengan kondisi NLOS (*Non Line of Sight*). Pada pengukuran dengan kondisi ini, hanya sedikit data yang terekam oleh alat ukur, Hal ini disebabkan karena pada lokasi titik pengukuran yaitu di Taman Lawang, semua titik pengukuran tidak hanya terhalang oleh gedung-gedung yang tinggi, tetapi juga terhalang oleh pepohonan yang lebat. Sehingga hal tersebut mengakibatkan sinyal yang diterima mengalami redaman yang tinggi selain itu juga mengalami efek *multipath*. Sehingga tidak banyak data yang dapat direkam oleh alat ukur.

Tabel 4.5 Data Pengukuran Kualitas Cakupan Lokasi Taman Lawang (kondisi NLOS jarak 3 km)

No	Level Sinyal Alat Ukur (dB μ V)	C/N (dB)	VBER
1	37	19,2	0,1
2	38,5	20,8	0,1
3	40,6	22,4	0,1

Sumber : Data pengukuran Kualitas Cakupan (NLOS), *Preventive Maintenance* TVRI Jakarta, Februari 2010

Tabel 4.6 Data Pengukuran Kualitas Cakupan Lokasi Gedung BPPT Lantai 3 (kondisi NLOS jarak 4 km)

No	Level Sinyal Alat Ukur (dB μ V)	C/N (dB)	VBER
1	38,2	18,9	0,1
2	38,3	20,5	0,1
3	42,4	24,2	0,1
4	43,1	24,7	0,00018
5	55,9	26	1,00E-07
6	54,5	27,9	1,00E-07
7	48,8	29,2	0,0000041
8	56,3	30,3	1,00E-07

Sumber : Data pengukuran Kualitas Cakupan (NLOS), *Preventive Maintenance* TVRI Jakarta, Februari 2010

Dari pengukuran kualitas cakupan didapatkan dua parameter pengukuran yaitu C/N dan VBER yang dapat dinyatakan dalam persamaan berikut:

Rumus *Carrier to noise ratio*:

$$\bullet \quad \frac{C}{N} = \frac{E_b \cdot m \cdot R_b / m}{N_0 \cdot W} = \frac{E_b \cdot R_b}{N_0 \cdot W}$$

Keterangan:

C = Daya sinyal (dB)

R_b = Kecepatan bit (μ s)

N = Daya noise (dB)

E_b = Energi bit (dB μ V)

W = Bandwidth yang digunakan (MHz) N_0 = Kerapatan daya noise (Watt/Hz)

- Rumus kecepatan bit

$$R_b = R_s (\log 2.J)$$

R_s = kecepatan simbol ($R_s = 1/T_s$) $T_s = 896\mu$ s

J = tingkat modulasi QAM

Maka: $R_b = 896\mu$ s ($\log 2.J$)

$$= 430 \text{ bps}$$

- Bandwidth yang digunakan = 578 MHz

$$\bullet \quad \frac{C}{N} = \frac{E_b \cdot R_b}{N_0 \cdot W} \text{ maka: } \frac{C}{N} = \frac{61 \cdot 430}{N_0 \cdot 578} = 16 \text{ dB}$$

$$N_0 = \frac{61.430}{578} = 45,38$$

$$= \frac{45,38}{16} = 2,83 \text{ Watt/Hz}$$

(kondisi *Line of Sight* (LOS))

- $\frac{C}{N} = \frac{E_b R_b}{N_0 W}$ maka: $\frac{C}{N} = \frac{37\ 430}{N_0\ 578} = 19,2 \text{ dB}$

$$N_0 = \frac{37.430}{578} = 27,5$$

$$= \frac{27,5}{19,2} = 1,4 \text{ Watt/Hz}$$

(kondisi *Non-Line of Sight* (LOS))

4.1.3 Data Hasil Pengukuran *Delay Spread*

Seperti yang telah dijelaskan sebelumnya, bahwa data hasil pengukuran *delay spread* akan dijadikan dasar pemodelan kanal sistem. Pengukuran ini menghasilkan *path delay profile*, dimana sinyal yang terkena *delay* ditandai dengan munculnya sinyal *echo* pada jarak tertentu. Berikut ini merupakan tabel hasil pengukuran *delay spread*.

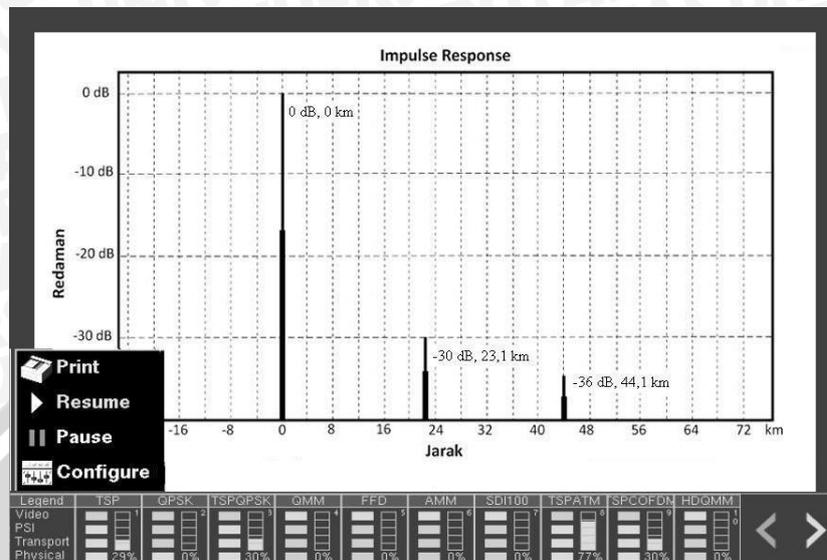
Tabel 4.7 Data Pengukuran *Delay Spread*

No	Path 1	Path 2	Path 3	Keterangan
1	no delay	23 μs (-30dB)	None	TVRI (LOS)
2	no delay	33 μs (-27dB)	None	BPPT Lt 3NLOS
3	no delay	33 μs (-36dB)	47 μs (-37 dB)	SCTV (NLOS)
4	no delay	73 μs (-38dB)	None	Lap CBD (LOS)
5	no delay	77 μs (-30dB)	147 μs (-36 dB)	RSAB (NLOS)

Sumber : Data pengukuran *Delay Spread*, *Preventive Maintenance* TVRI Jakarta, Februari 2010

Nilai pada tanda kurung merupakan besarnya redaman pada *path* tersebut. Berdasarkan tabel 4.7, maka dapat dilihat bahwa hasil pengukuran *delay spread* pada lokasi RSAB merupakan hasil terburuk yang terukur, dengan *delay* serta redaman terbesar pada *path* ke-3. Berdasarkan tabel 4.7, maka *delay* yang terjadi telah melebihi *guard interval* sistem yang hanya sebesar 56 μs , tetapi masih lebih kecil dari periode simbol yaitu sebesar 896 μs (mode 8k).

Grafik *path delay profile* dari hasil pengukuran di RSAB dapat dilihat pada gambar 4.1 di bawah ini:



Gambar 4.1 *Path Redaman Fungsi Jarak* dari Pengukuran *Delay Spread*

Sumber : Grafik pengukuran *Delay Spread, Preventive Maintenance*

TVRI Jakarta, Februari 2010

Berdasarkan gambar 4.1, dapat dihitung secara matematis menggunakan persamaan jarak yaitu:

$$s = v \times t \quad (4.1)$$

Keterangan: s = jarak (m)

v = kecepatan gelombang radio (m/s)

t = waktu (s)

Persamaan kecepatan gelombang radio yaitu:

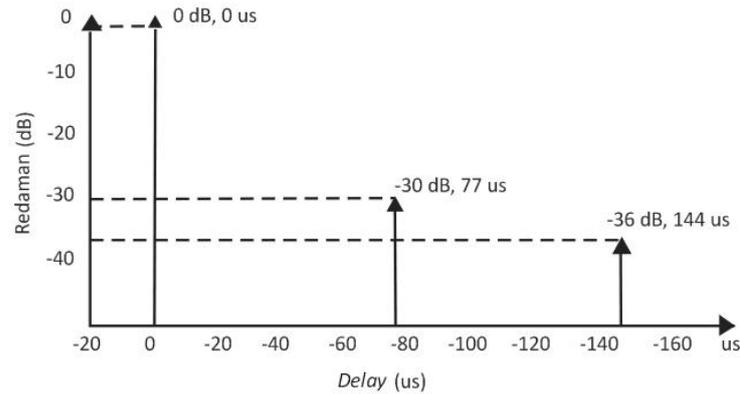
$$v = \frac{1}{\sqrt{\epsilon_0 \cdot \mu_0}} \text{ dengan,}$$

$$\epsilon_0 = \frac{1}{4\pi \cdot 9} \cdot 10^{-9} \text{ C/N.m}^2 \text{ dan}$$

$$\mu_0 = 4\pi \cdot 10^{-7} \text{ W/A.m} = 3 \times 10^8 \text{ m/s}$$

Dari persamaan 4.1, jarak yang terkena *delay* dihitung sebagai berikut:

- Pada *path 2 delay* 77 μs , maka $77 \mu\text{s} \times 3 \cdot 10^8 \text{ m/s} = 23,1 \text{ km}$.
- Pada *path 3 delay* 147 μs , maka $147 \mu\text{s} \times 3 \cdot 10^8 \text{ m/s} = 44,1 \text{ km}$.



Gambar 4.2 *Path delay* dan Redaman pada Pengukuran *Delay Spread*

Sumber : Grafik pengukuran *Delay Spread*, *Preventive Maintenance*

TVRI Jakarta, Februari 2010

Contoh perhitungan *Mean excess delay*, *rms delay spread*, dari suatu kanal *multipath* pada hasil pengukuran di RSAB Jakarta. Seperti terlihat pada gambar 4.2 diatas:

- *Mean excess delay*:

$$\bar{\tau} = \frac{\sum_k a_k^2 \tau_k}{\sum_k a_k^2} = \frac{\sum_k P(\tau_k) \tau_k}{\sum_k P(\tau_k)}$$

$$\bar{\tau} = \frac{(1)0 + (0,001)77 + (0,012)144}{(0,012 + 0,001 + 1)} = \frac{1,805}{1,013} = 1,78 \mu s$$

- Momen kedua *delay profile*:

$$\bar{\tau}^2 = \frac{(1)0^2 + (0,001)77^2 + (0,012)144^2}{(0,012 + 0,001 + 1)} = \frac{254,75}{1,013} = 251,48 \mu s$$

- *Rms delay spread*:

$$\sigma_{\tau} = \sqrt{251,48(1,78)^2} = 15,7 \mu s$$

- Coheren Bandwidth:

$$B_c \approx \frac{1}{5\sigma_{\tau}} = 12,7 \text{ kHz}$$

4.2 Parameter Sistem dan Parameter Pemodelan

Pada skripsi ini, akan dibandingkan data hasil pengukuran yang dilakukan di Jakarta dengan data hasil pemodelan sistem yang dilakukan dengan bantuan *software* Matlab 7.0.1. Parameter pemodelan yang dilakukan menyerupai sistem yang diimplementasikan, seperti yang telah dijelaskan pada bab 3.

4.2.1 Parameter Sistem

Sistem yang diimplementasikan di Jakarta memiliki parameter sistem sebagai berikut:

Tabel 4.8 Parameter DVB-T yang diimplementasikan di Jakarta

No	Parameter Sistem	Keterangan
1	Modulasi	16 QAM
2	<i>Code Rate</i>	3/4 (<i>Punctured</i>)
3	<i>Guard Interval</i>	1/16 (56 μ s)
4	<i>Mode</i>	8k (durasi simbol 896 μ s)
5	Frekuensi Kerja	578 MHz
6	<i>Standar Video</i>	MPEG-2
7	<i>Resolusi</i>	720 x 576

Sumber: Euopian Standart Telekomunication Institute (ESTI)
TR 101 190 v 1.2.1

4.2.2 Parameter Pemodelan

Analisa hasil pemodelan dilakukan berdasarkan model sistem DVB-T yang telah dijelaskan pada bab sebelumnya. Berikut ini merupakan parameter pada pemodelan :

1. Pemodelan dilakukan dengan model *baseband*, artinya menggunakan sinyal asli dan tidak menggunakan *pilot carrier*. (Sumber: Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, 2010)
2. Pemodelan sistem diasumsikan sinkron dengan data hasil pengukuran. (Sumber: Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, 2010)
3. Data input berupa deretan bilangan biner berjumlah 1 juta bit, sebab nilai BER yang diamati bukan merupakan BER output sistem, sehingga tidak perlu mengirimkan data berupa video. (Sumber: Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, 2010)

4. *Outer Interleaver* yang digunakan adalah *convolutional interleaver* dengan *branch* (baris) sebanyak 12 baris dan masing-masing register (J) yang berada dalam sebuah baris memuat data sebanyak 17 byte. (Sumber: Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, 2010)
5. *Inner coding* yang digunakan adalah kode konvolusional dengan *rate* $3/4$ yang merupakan hasil *puncture code rate* $1/2$ dengan *constraint length* = 7 serta *code rate* $1/2$. (Sumber: Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, 2010)
6. *Inner interleaver* yang digunakan adalah blok interleaver dengan banyaknya baris sama dengan 4. (Sumber: Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, 2010)
7. Tipe modulasi yang digunakan adalah 16 QAM. (Sumber: *Digital Communication, Fundamental and Application*, 2 edition, Prentice Hall International Inc, 2001)
8. Banyaknya *subcarrier* yang digunakan adalah 6817 buah. (Sumber: (ESTI) TR 101 190 v 1.2.1, Hal 13)
9. Jumlah titik IFFT/FFT adalah 131072. (Sumber: *Theory And Application of OFDM and CDMA*, John Willey&Sons, Inc, 2005.)
10. Durasi simbol (T_s) adalah 896 μs . (Sumber: (ESTI) TR 101 190 v 1.2.1, Hal 13)
11. *Guard interval* yang digunakan adalah $1/16$ dan $1/32$ dari durasi simbol. (Sumber: (ESTI) TR 101 190 v 1.2.1, Hal 18)
12. Kanal yang digunakan dipengaruhi oleh noise AWGN dan *Reyleigh* yang mempunyai 3 buah *path*. *Path* pertama tidak terdelay dengan redaman 0 dB, *path* kedua terdelay 77 μs dan redaman -30 dB dan *path* ketiga terdelay sebesar 147 μs dengan redaman -36 dB. (Sumber: Pemodelan Kanal Propagasi Sistem DVB-T Jakarta, 2010)

Kanal yang digunakan pada pemodelan merupakan kanal dengan *delay spread* terbesar yang terukur pada lokasi RSAB Jakarta seperti yang telah dijelaskan pada bab sebelumnya.

Pemodelan menggunakan nilai S/N atau SNR (*Signal to Noise Ratio*), sedangkan data yang diperoleh pada data pengukuran adalah C/N (*Carrier to Noise Ratio*). Karena skripsi ini tidak mensimulasikan *pilot carrier*, maka nilai C/N yang terukur dianggap sama dengan nilai S/N pada pemodelan yang dilakukan.

4.3 Hasil Pemodelan dan Analisa Data

Analisa yang digunakan pada skripsi ini berdasarkan standar ETSI 300 744, yang merupakan standar dari sistem DVB-T. Tabel 4.9 merupakan tabel yang menjelaskan nilai SNR yang dibutuhkan sistem untuk mencapai QEF (*Quasi Error Free*) yang dilengkapi pula dengan *bit rate* yang dihasilkan setelah Reed-Solomon *Decoder*. Pada tabel 4.9 dapat dilihat nilai *bit rate* yang diperoleh dengan variasi nilai *guard interval*, *code rate*, *level* modulasi serta jenis kanal.

Tabel 4.9 Standar parameter pada sistem DVB-T

Modulation	Code rate	Required C/N for BER = 2×10^{-4} after Viterbi QEF after Reed-Solomon			Bit rate (Mbit/s)			
		Gaussian channel	Ricean channel (F ₁)	Rayleigh channel (F ₂)	$\Delta/T_U = 1/4$	$\Delta/T_U = 1/8$	$\Delta/T_U = 1/16$	$\Delta/T_U = 1/32$
QPSK	1/2	3,1	3,6	5,4	4,98	5/53	5,85	6,03
QPSK	2/3	4,9	5,7	8,4	6,64	7/37	7,81	8,04
QPSK	3/4	5,9	6,8	10,7	7,46	8/29	8,78	9,05
QPSK	5/6	6,9	8,0	13,1	8,29	9/22	9,76	10,05
QPSK	7/8	7,7	8,7	16,3	8,71	9/68	10,25	10,56
16-QAM	1/2	8,8	9,6	11,2	9,95	11/06	11,71	12,06
16-QAM	2/3	11,1	11,6	14,2	13,27	14/75	15,6156	16,09
16-QAM	3/4	12,5	13,0	16,7	14,93	16/59	17,52	18,10
16-QAM	5/6	13,5	14,4	19,3	16,59	18/43	19,49	20,11
16-QAM	7/8	13,9	15,0	22,8	17,42	19/35	20,56	21,11
64-QAM	1/2	14,4	14,7	16,0	14,93	16/59	17,56	18,10
64-QAM	2/3	16,5	17,1	19,3	19,91	22/12	23,42	24,13
64-QAM	3/4	18,0	18,6	21,7	22,39	24/88	26,35	27,14
64-QAM	5/6	19,3	20,0	25,3	24,88	27/65	29,27	30,16
64-QAM	7/8	20,1	21,0	27,9	26,13	29/03	30,74	31,67

Sumber : European Standart Telecommunication Institute (ETSI)
EN 300 477 v1.4.1, hal 40

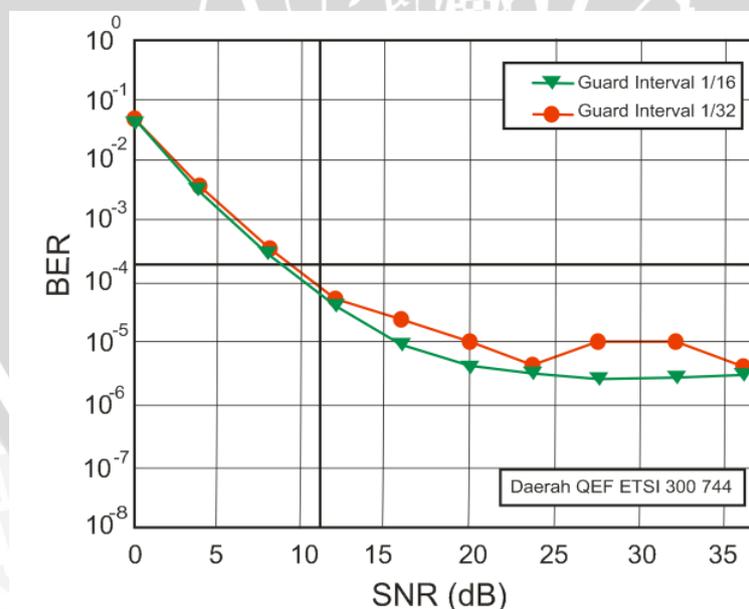
Berdasarkan tabel 4.9 di atas, maka dapat dilihat bahwa pemilihan parameter sistem seperti *code rate*, *guard interval* dan level modulasi serta asumsi kanal mempengaruhi nilai SNR yang dibutuhkan sistem untuk mencapai QEF (*Quasi Error Free*), juga mempengaruhi nilai *bit rate* yang diterima pada penerima.

QEF (*Quasi Error Free*) memiliki pengertian bahwa dalam waktu 1 jam, hanya terdapat 1 *event error*, yang berkolerasi pada nilai BER output sistem senilai 1×10^{-11} . QEF tersebut dapat terjadi bila nilai BER setelah proses *viterbi decoding*, yaitu sebelum masuk ke blok Reed-Solomon *decoder* bernilai 2×10^{-4} . Berdasarkan Tabel 4.9 tersebut, maka sistem dengan level modulasi 16-QAM

dengan *code rate* 1/2 akan mencapai kondisi QEF (*Quasi Error Free*) pada nilai SNR 11.2 dB bila kanal diasumsikan merupakan kanal *Rayleigh*. Sedangkan sistem dengan asumsi kanal yang sama, level modulasi 16-QAM dengan *code rate* 3/4 akan mencapai kondisi QEF (*Quasi Error Free*) pada nilai SNR 16.7 dB. Berdasarkan tabel 4.9 tersebut, pemilihan nilai *guard interval* hanya akan mempengaruhi nilai *bit rate* yang diterima oleh penerima.

Proses pemodelan sistem dilakukan dengan menggunakan nilai dari pengukuran *delay spread* yaitu hasil *delay* serta redaman terburuk pada daerah RSAB yang kemudian diolah menggunakan *software* Matlab versi 7.0.1, adapun *listing* program dalam pemodelan sistem dapat dilihat pada lampiran B. Dari *listing* program tersebut dapat diperoleh nilai BER rata-rata dengan *code rate* 1/2 dan 3/4 serta menggunakan *guard interval* 1/16 dan 1/32. Nilai rata-rata tersebut digunakan sebagai titik-titik pada grafik pemodelan sistem.

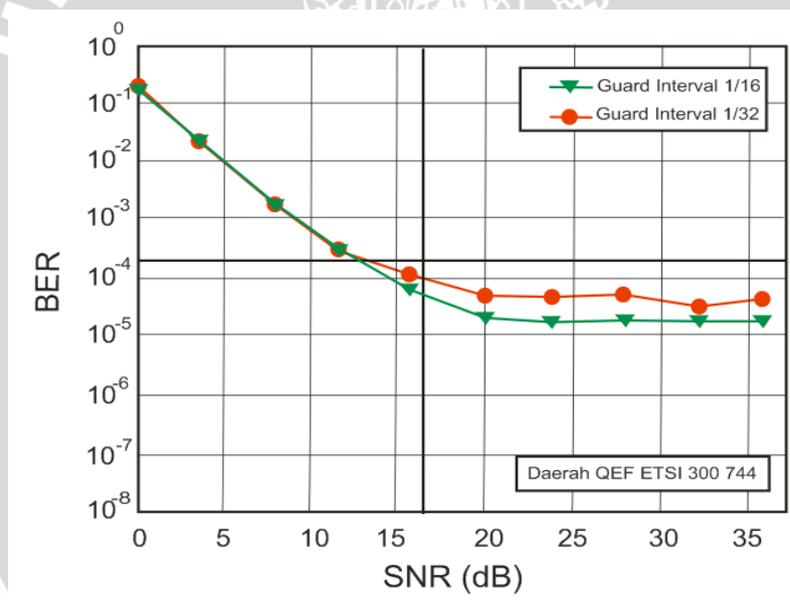
Berikut ini merupakan hasil pemodelan sistem. Grafik 4.1 merupakan hasil pemodelan dengan nilai *code rate* sebesar 1/2, dengan nilai *guard interval* 1/16 dan 1/32. Sedangkan grafik 4.2 merupakan hasil pemodelan sistem dengan *code rate* 3/4, juga dengan nilai *guard interval* 1/16 dan 1/32. Titik-titik pada grafik pemodelan sistem berasal dari lampiran C.



Grafik 4.1 BER Fungsi SNR Pemodelan Sistem dengan Nilai *Code Rate* 1/2

(Sumber: Pemodelan Sistem, *Preventive Maintenance*
TVRI Jakarta, Februari 2010)

Pada grafik 4.1 dapat dijelaskan bahwa, sistem dengan *code rate* $\frac{1}{2}$ dapat mencapai QEF (*Quasi Error Free*) pada SNR 11,2. Dari hasil pemodelan terdapat masing-masing tiga titik dari *guard interval* $\frac{1}{16}$ dan $\frac{1}{32}$ yang tidak memenuhi syarat QEF. Hal ini disebabkan karena pada titik tersebut mempunyai nilai SNR dan BER kurang dari nilai yang dianjurkan untuk mencapai daerah QEF yaitu dengan SNR 0,4 dan 8. Sedangkan nilai BER nya 0,0434 - 0,0035 - 0,0002 untuk *guard interval* $\frac{1}{16}$ dan 0,0461 - 0,0040 - 0,0003 untuk *guard interval* $\frac{1}{32}$. Pengaruh penggunaan *guard interval* $\frac{1}{16}$ dan $\frac{1}{32}$ mulai tampak pada saat nilai SNR berada pada daerah QEF ETSI 300 744, walaupun bila dilihat tidak mempengaruhi kinerja sistem. Hal tersebut disebabkan karena redaman yang besar pada *path* kedua dan ketiga dari pemodelan kanal. Sehingga walaupun *delay* yang terjadi melebihi nilai *guard interval*, tetapi hal ini tidak mempengaruhi kinerja sistem.



Grafik 4.2 BER Fungsi SNR Pemodelan Sistem dengan Nilai *Code Rate* $\frac{3}{4}$

(Sumber: Pemodelan Sistem, *Preventive Maintenance*
TVRI Jakarta, Februari 2010)

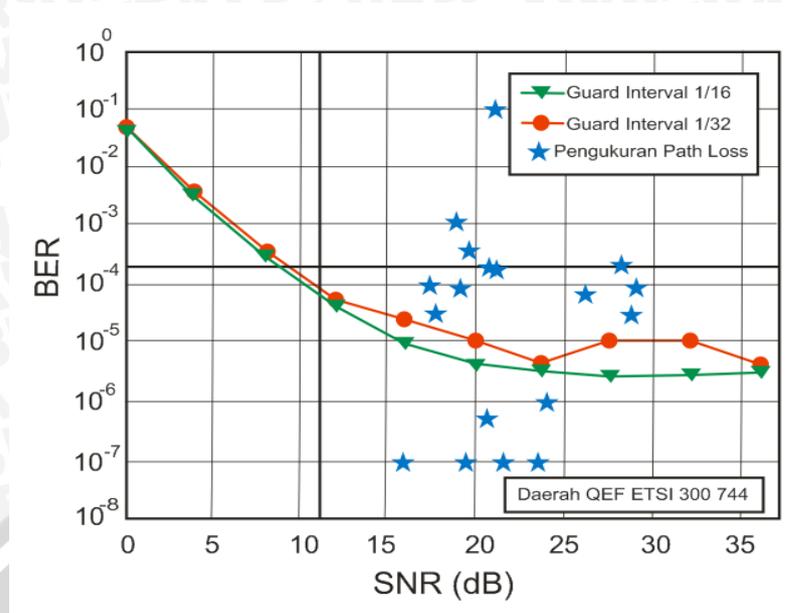
Pada grafik 4.2 dapat dijelaskan bahwa, sistem dengan *code rate* $\frac{3}{4}$ dapat mencapai QEF (*Quasi Error Free*) pada SNR 16,7. Dari hasil pemodelan terdapat masing-masing lima titik dari *guard interval* $\frac{1}{16}$ dan $\frac{1}{32}$ yang tidak memenuhi syarat QEF. Hal ini disebabkan karena pada titik tersebut mempunyai nilai SNR dan BER kurang dari nilai yang dianjurkan untuk mencapai daerah

QEF. Selain itu juga terdapat masing-masing satu titik dari kedua *guard interval* tersebut mempunyai nilai BER yang dianjurkan. Namun hanya mempunyai nilai SNR sebesar 16 dB hal ini kurang dari yang syaratkan oleh QEF pada *code rate* $\frac{3}{4}$. Serta pengaruh penggunaan *guard interval* 1/16 dan 1/32 juga mulai tampak pada saat nilai SNR berada pada daerah QEF ETSI 300 744, walaupun bila dilihat tidak mempengaruhi kinerja sistem. Hal itu disebabkan karena redaman yang besar pada *path* kedua dan ketiga dari pemodelan kanal. Sehingga walaupun *delay* yang terjadi melebihi nilai *guard interval*, tetapi hal ini tidak mempengaruhi kinerja sistem.

Berdasarkan kedua grafik tersebut, maka dapat dilihat bahwa *code rate* $\frac{1}{2}$ dapat memperbaiki kinerja sistem. Bila dibandingkan dengan nilai *code rate* maka sistem dapat mencapai kondisi QEF (*Quasi Error Free*) pada nilai SNR sekitar 11 dB, lebih kecil 5 dB bila dibandingkan nilai SNR yang dibutuhkan oleh sistem dengan *code rate* $\frac{3}{4}$ untuk mencapai kondisi QEF (*Quasi Error Free*).

4.3.1 Perbandingan Data Hasil Pemodelan dan Data Hasil Pengukuran *Path Loss*

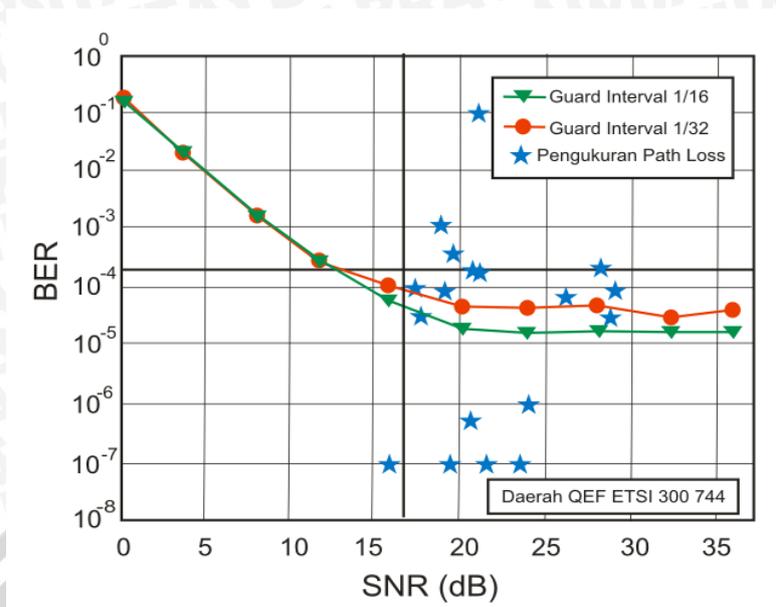
Seperti yang telah dijelaskan bahwa pengukuran *path loss* dilakukan dengan memilih titik lokasi pada area *main lobe* antena pemancar. Pengukuran ini dilakukan untuk melihat variasi skala besar yaitu pada kondisi LOS (*Line of Sight*) dan NLOS (*Non Line of Sight*) dari nilai BER sistem. Berikut merupakan grafik perbandingan data hasil pemodelan dengan data hasil pengukuran *path loss* yang digambarkan dengan titik-titik yang nilainya berasal dari tabel 4.1.



Grafik 4.3 BER Fungsi SNR Pemodelan *Code Rate*1/2 dengan Data Pengukuran *Paht Loss*.

(Sumber: Data Pengukuran *path loss* dan Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, Februari 2010)

Berdasarkan grafik 4.3 dapat dilihat bahwa hampir semua titik pengukuran memiliki nilai BER yang lebih besar dari hasil pemodelan. Namun bila dilihat dari syarat (*Quasi Error Free*), yaitu nilai BER maksimum sebesar 2×10^{-4} , maka hanya terdapat 3 titik pengukuran yang memiliki nilai BER lebih besar dari 2×10^{-4} yaitu 0,0012, 0,00044 dan 0,1, sehingga tidak memenuhi syarat QEF (*Quasi Error Free*). Pada titik pengukuran yang memiliki nilai BER sekitar 0,1, walaupun memiliki nilai SNR yang tinggi, yaitu sekitar 21 dB hal tersebut diakibatkan karena adanya pengaruh dari efek *multipath*.



Grafik 4.4 BER Fungsi SNR Pemodelan *Code Rate 3/4* dengan Data Pengukuran *Paht Loss*.

(Sumber: Data Pengukuran *path loss* dan Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, Februari 2010)

Berdasarkan grafik 4.4 dapat dilihat bahwa juga hampir semua titik pengukuran memiliki nilai BER yang lebih besar dari hasil pemodelan. Namun masih memenuhi syarat QEF (*Quasi Error Free*) yaitu nilai BER maksimum sebesar 2×10^{-4} , tetapi terdapat 3 titik pengukuran yang memiliki nilai BER lebih besar dari 2×10^{-4} dan 1 titik yang mempunyai nilai SNR 15,8 dB meskipun mempunyai nilai VBER 1,00E-07 namun hal ini belum memenuhi syarat *code rate 3/4* untuk kondisi QEF (*Quasi Error Free*) sebesar 16,7 dB.

Namun secara umum, sistem dengan menggunakan *code rate 1/2* maupun *code rate 3/4* memiliki kinerja yang baik dengan skala besar (global), yang ditunjukkan dari banyaknya titik lokasi yang memiliki nilai BER memenuhi syarat QEF (*Quasi Error Free*).

4.3.2 Perbandingan Data Hasil Pemodelan dan Data Hasil Pengukuran Kualitas Cakupan

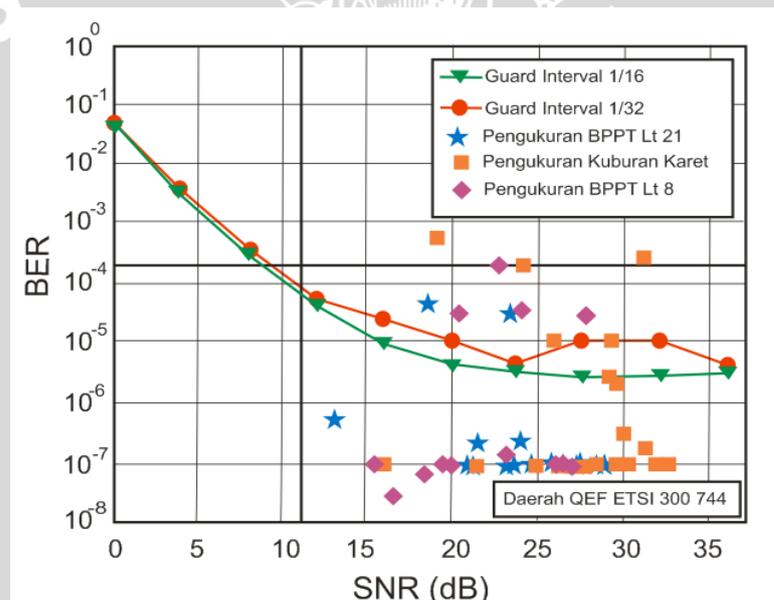
Pengukuran kualitas cakupan merupakan pengukuran yang dilakukan untuk melihat variasi nilai BER sistem pada skala kecil. Pengukuran jenis ini

juga dilakukan untuk melihat sejauh mana sistem dapat memberikan layanan pada suatu daerah.

Pengukuran kualitas cakupan ini dibedakan berdasarkan asumsi kondisi penerima, yaitu kondisi LOS (*Line of Sight*) serta pada kondisi NLOS (*Non-Line of Sight*).

4.3.2.1 Kondisi LOS

Lokasi pengukuran yang dilakukan pada kondisi LOS terdiri dari 3 lokasi yaitu depan Makam Karet, Gedung Parkir BPPT Lantai 8 dan Lantai 21. Pengukuran ini menghasilkan banyak data yang dapat dijadikan bahan analisa. Berikut merupakan hasil perbandingan data pemodelan dengan data hasil pengukuran pada kondisi LOS (*Line of Sight*), titik - titik pengukuran pada grafik 4.5 berasal dari tabel 4.2, 4.3 dan 4.4.

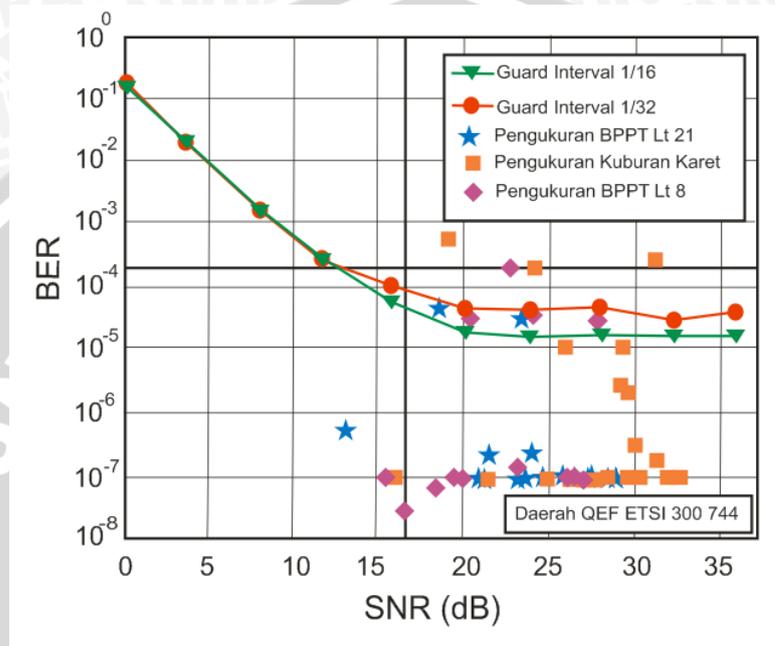


Grafik 4.5 BER Fungsi SNR Pemodelan *Code Rate* $\frac{1}{2}$ dengan Data Pengukuran Kualitas Cakupan LOS

(Sumber: Data Pengukuran Kualitas Cakupan dan Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, Februari 2010)

Berasarkan grafik 4.5, tampak jelas bahwa hampir semua titik pada pengukuran kondisi LOS (*Line of Sight*) mempunyai nilai BER yang lebih baik dari pada hasil pemodelan. Hal ini menunjukkan sistem pada kondisi LOS (*Line of Sight*) memiliki karakteristik propagasi yang lebih baik dari pemodelan yang

dilakukan. Variasi hasil pengukuran pada kondisi LOS dipengaruhi oleh pemilihan titik pengukuran. Namun ada beberapa titik pengukuran yang memiliki nilai BER yang lebih buruk dari pada pemodelan. Dari semua hasil pengukuran pada kondisi LOS (*Line of Sight*), hanya dua titik yang mempunyai nilai BER lebih buruk dari 2×10^{-4} yang merupakan syarat terjadinya QEF (*Quasi Error Free*) yaitu 0,00062 dan 0,00023.



Grafik 4.6 BER Fungsi SNR Pemodelan *Code Rate* 3/4 dengan Data Pengukuran Kualitas Cakupan LOS

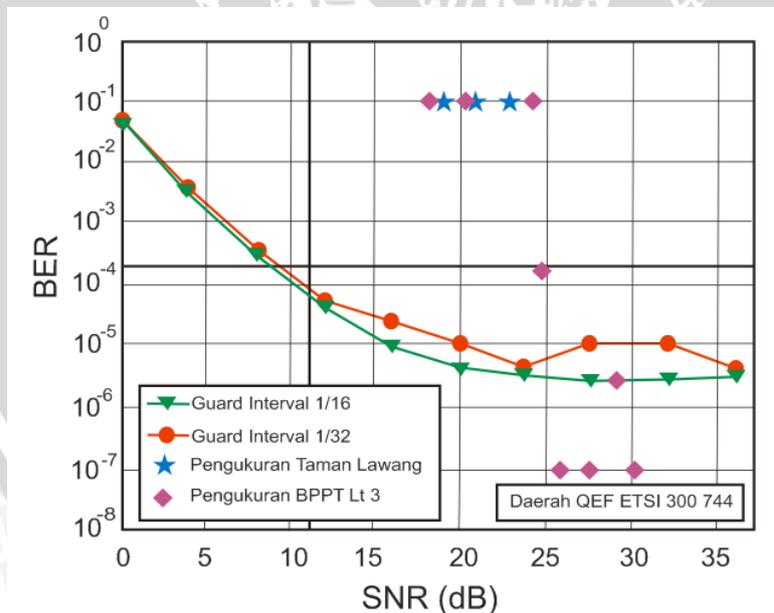
(Sumber: Data Pengukuran Kualitas Cakupan dan Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, Februari 2010)

Berasarkan grafik 4.6, juga hampir semua titik pada pengukuran kondisi LOS mempunyai nilai BER yang lebih baik dari pada hasil pemodelan. Namun selain juga mempunyai 2 titik yang mempunyai nilai BER lebih buruk dari 2×10^{-4} yaitu 0,00062 dan 0,00023. Pada pemodelan menggunakan *code rate* $\frac{3}{4}$ juga terdapat tiga titik yang tidak memenuhi syarat dari SNR yang dianjurkan yaitu 16,7 dB. Adapun titik-titik tersebut mempunyai nilai SNR hanya 13,6 dB, 15,8 dB dan 16 dB. Titik dengan nilai BER lebih buruk dari pada pemodelan sistem yaitu 0,00062 dan 0,00023 merupakan titik pengukuran yang dilakukan dengan kondisi terhalang *obstacle* seperti pohon, mobil maupun *obstacle* lain yang berukuran tidak besar.

Berdasarkan perbandingan data hasil pemodelan dengan data hasil pengukuran kualitas cakupan, dapat dilihat bahwa sistem DVB-T yang diimplementasikan di Jakarta dapat memberikan pelayanan optimal pada daerah dengan kondisi LOS, yaitu daerah yang tidak terhalang *obstacle*. Sedangkan pada daerah NLOS, sistem tidak dapat bekerja dengan optimal akibat efek *multipath*.

4.3.2.2 Kondisi NLOS

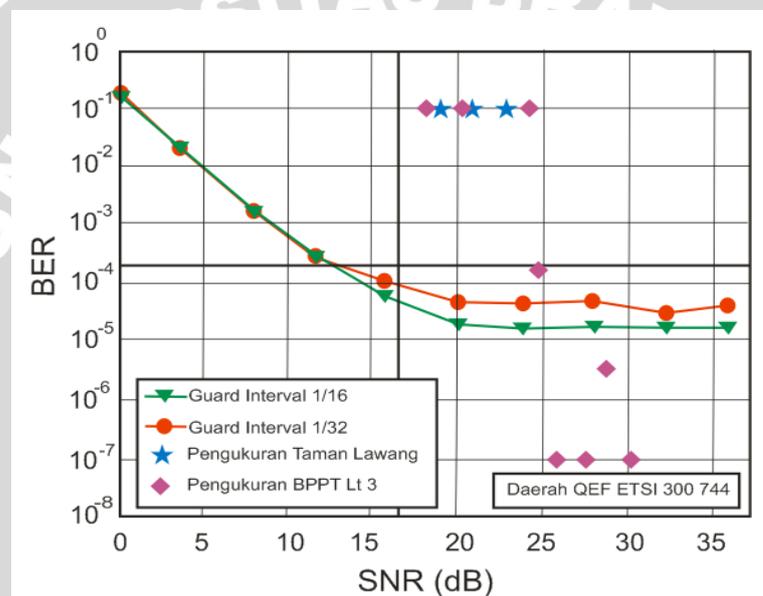
Berikut ini adalah perbandingan hasil pemodelan sistem dengan data yang didapat melalui pengukuran kualitas cakupan pada kondisi NLOS. Data hasil pengukuran tidak banyak yang dapat direkam oleh alat ukur, Hal tersebut disebabkan karena pada lokasi Taman Lawang, semua titik pengukuran tidak hanya terhalang oleh gedung-gedung yang tinggi, tetapi juga terhalang oleh pepohonan yang lebat. Sehingga hal tersebut mengakibatkan sinyal yang diterima mengalami redaman yang tinggi selain juga mengalami efek *multipath*. Sehingga hanya sedikit data yang dapat dijadikan bahan analisis. Titik-titik pengukuran pada grafik 4.7 berasal dari tabel 4.5 dan 4.6.



Grafik 4.7 BER Fungsi SNR Pemodelan *Code Rate* $\frac{1}{2}$ dengan data Pengukuran Kualitas Cakupan NLOS

(Sumber: Data Pengukuran Kualitas Cakupan dan Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, Februari 2010)

Berdasarkan grafik 4.7 dapat dijelaskan bahwa, tampak terdapat titik-titik hasil pengukuran yang memiliki nilai BER lebih buruk dari syarat QEF (*Quasi Error Free*) yaitu 0,1 selain itu juga terdapat titik-titik hasil pengukuran yang memenuhi syarat QEF (*Quasi Error Free*). Untuk titik yang tidak memenuhi syarat dari QEF dengan nilai BER maksimum 2×10^{-4} adalah pada lokasi Taman Lawang. Hal tersebut disebabkan karena pada lokasi Taman Lawang, semua titik pengukuran tidak hanya terhalang oleh gedung-gedung yang tinggi, tetapi juga terhalang oleh pepohonan yang lebat. Sehingga hal tersebut mengakibatkan sinyal yang diterima mengalami redaman yang tinggi selain juga mengalami efek *multipath*.



Grafik 4.8 BER Fungsi SNR Pemodelan *Code Rate* 3/4 dengan Data Pengukuran Kualitas Cakupan NLOS

(Sumber: Data Pengukuran Kualitas Cakupan dan Pemodelan Sistem, *Preventive Maintenance* TVRI Jakarta, Februari 2010)

Berdasarkan grafik 4.8 juga mempunyai titik-titik yang sama pada grafik 4.7. Namun bila diperhatikan berdasarkan grafik 4.8, maka semua titik yang memiliki nilai BER lebih baik dibandingkan hasil pemodelan sistem yaitu $1,00E-07$ dan $0,0000041$ adalah titik-titik yang berada pada lokasi gedung BPPT lantai 3, sebab pada tempat tersebut walaupun banyak terhalang oleh gedung-gedung tinggi di sekitar jalan Thamrin, lokasi pengukuran masih terletak pada daerah yang tinggi yaitu di lantai 3. Sehingga dari pengukuran yang mempunyai

nilai BER bagus yaitu sesuai syarat QEF (*Quasi Error Free*) dengan BER maksimum 2×10^{-4} merupakan titik dimana sinyal terima tidak mengalami efek *multipath*.



BAB V PENUTUP

Dari pelaksanaan skripsi ini, dapat diambil beberapa kesimpulan dan saran, yaitu:

5.1 KESIMPULAN

1. Berdasarkan hasil pengukuran sistem DVB-T, pada pengukuran *path loss*, kualitas cakupan dan *delay spread* semuanya mempunyai kinerja yang baik pada daerah dengan kondisi LOS (*Line of Sight*) dan memenuhi standar dari QEF ETSI 300 744 dengan nilai BER maksimum 2×10^{-4} dan nilai SNR 11,2 untuk *code rate* $\frac{1}{2}$ dan SNR 16,7 untuk *code rate* $\frac{3}{4}$.
2. Berdasarkan hasil pemodelan, menggunakan *guard interval* 1/16 dan 1/32 tidak mempengaruhi kinerja sistem. Karena nilai BER yang dihasilkan lebih banyak berada pada daerah QEF (*Quasi Error Free*), dengan nilai BER maksimum sebesar 2×10^{-4} .
3. Berdasarkan hasil pemodelan, pemilihan nilai *code rate* $\frac{1}{2}$ dapat meningkatkan kinerja sistem sekitar 5 dB, hal ini disebabkan karena *code rate* $\frac{1}{2}$ mempunyai SNR 11,2 dB sedangkan *code rate* $\frac{3}{4}$ mempunyai SNR 16,7 dB berdasarkan syarat QEF.
4. Sistem tidak dapat bekerja dengan baik pada daerah NLOS (*Non-Line of Sight*), karena pada kondisi ini banyak terdapat *obstacle* dan mengakibatkan diperoleh nilai BER lebih besar dari 2×10^{-4} .
5. Sistem memiliki jarak jangkauan penerimaan sekitar 4 km, yang berada pada lokasi di atas Jembatan Tomang (kondisi LOS), dengan nilai SNR = 23,8 dB dan nilai BER = 9,00E-07.

5.2 SARAN

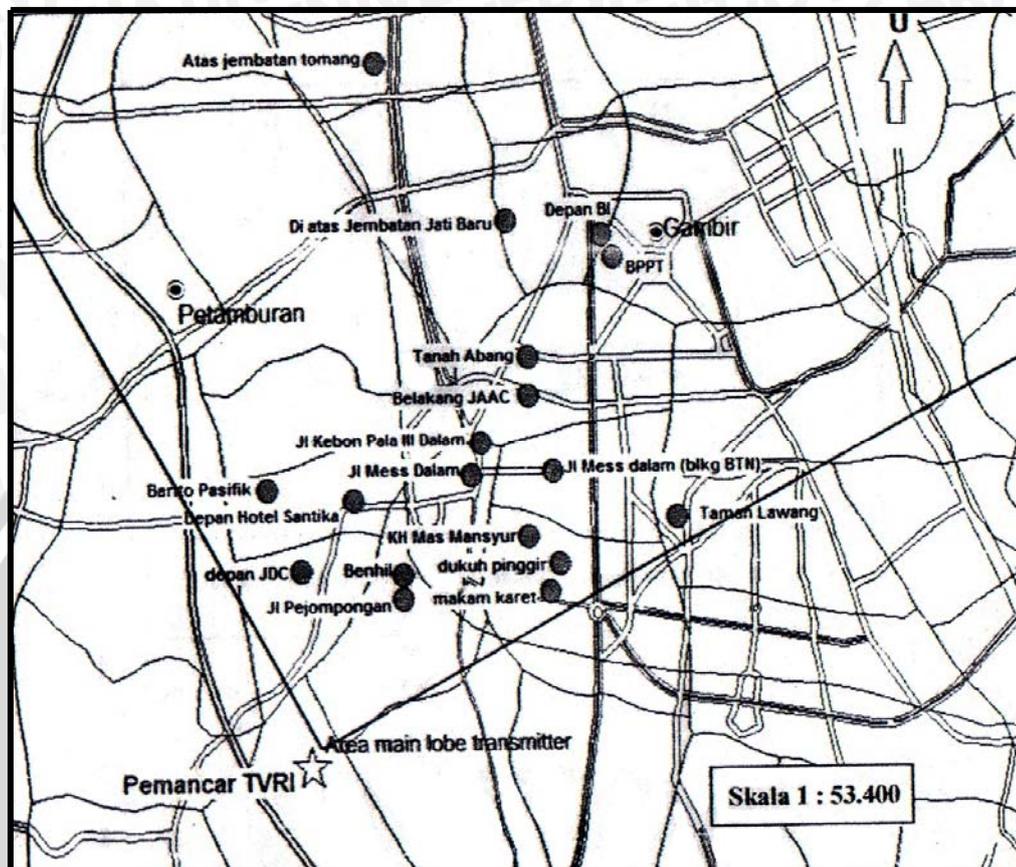
1. Untuk dapat menjangkau daerah dengan kondisi propagasi yang buruk atau NLOS (*Non-Line of Sight*), sistem sebaiknya mengubah nilai *code rate* dari $\frac{3}{4}$ menjadi $\frac{1}{2}$ karena pemilihan nilai *code rate* ini dapat meningkatkan kinerja sistem sekitar 5 dB.
2. Untuk menjangkau daerah yang lebih luas, disarankan menggunakan modulasi QPSK, walaupun hal itu dapat menurunkan nilai *bit rate* sistem berdasarkan standar parameter pada sistem DVB-T.
3. Untuk hasil yang lebih sempurna, perlu dilakukan pengukuran *delay spread* pada daerah-daerah yang memiliki BER buruk yaitu pada kondisi yang banyak terdapat *obstacle* atau kondisi NLOS (*Non-Line of Sight*), sehingga didapatkan hasil pengukuran yang lebih lengkap. Karena hasil pengukuran dari *delay spread* adalah suatu acuan untuk melakukan desain sistem.



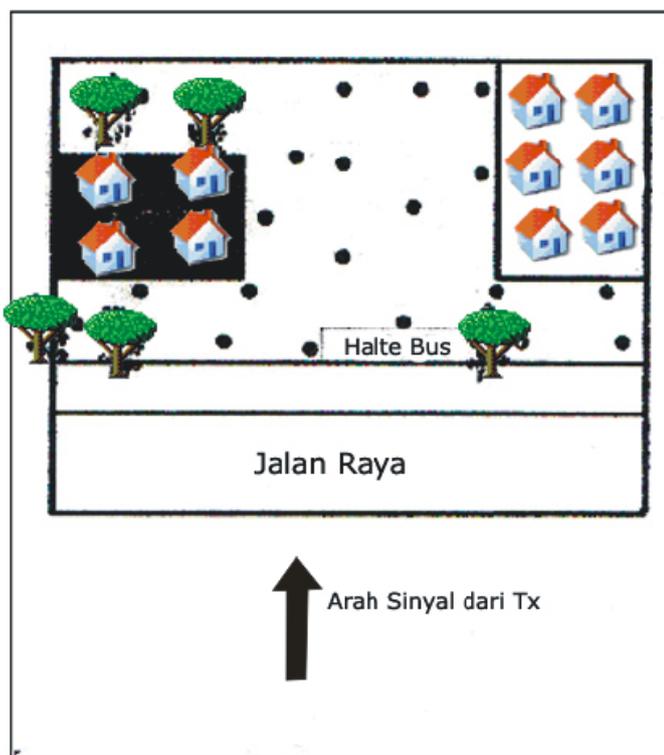
DAFTAR PUSTAKA

1. *European Standart Telecommunication Institute (ETSI) EN 300 744 v1.4.1.*
2. Walter Fischer, *Digital Television, A Practical Guide for Beginners*, Rohde&Schwarz Gmbh&Co, 2004.
3. Rizka Fadillah, *Studi Tentang Teknik OFDM untuk Penyiaran TV Digital Terrestrial (DVB-T)*, Tugas Akhir.
4. William C. Y. Lee, *Mobile Communication Design Fundamentals*, 2 edition, A Willey-Interscience Publication, John Willey&Sons, Inc, 1993.
5. Hanif Cana Pribadi, *Perbandingan Kinerja Sistem ATSC dan DVB-T untuk Transmisi Video melalui Kanal Frequency Selective*, Tugas Akhir.
6. Kaveh Palavan, Allen H. Levesque, *Wireless Information Network*, A Willey-Interscience Publication, John Willey&Sons, Inc, 1995
7. Henrik Schulze dan Christian Luders, *Theory And Application of OFDM and CDMA*, John Willey&Sons, Inc, 2005.
8. Theodore S. Rappaport, *Wireless Communicaton, Principles and Practice*, 2 edition, Prentice Hall, Inc, 2002.
9. Bernard Sklar, *Digital Communication, Fundamental and Aplication*, 2 edition, Prentice Hall International Inc, 2001.
10. Wang, Weizheng, *Communications Toolbox*, The Math Work, Inc. 1996
11. Promax, *User's Manual Prodig-5*, 2005.
12. <http://www.es.lth.se/home/channelmodelling/docs.pdf>.
13. en.wikipedia.org/wiki/rayleigh.

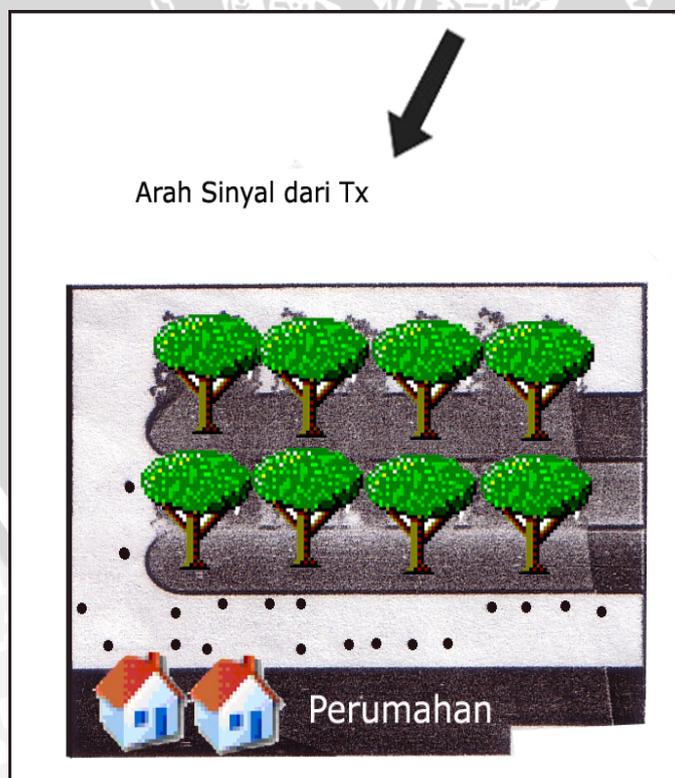
LAMPIRAN A
 PETA LOKASI PENGUKURAN



A.1. Peta Lokasi Pengukuran *Path loss*



A.2 Denah Lokasi Pengukuran Kualitas Cakupan Lokasi depan Makam Karet



A.2 Denah Lokasi Pengukuran Kualitas Cakupan Lokasi Taman Lawang

LAMPIRAN B

LISTING PROGRAM

```

%-----
%Transmitter
%-----
Clear all
clc;
data_generator=randint(1,1000000,[0 1],1);%input
awal = 1;
akhr = 200000;
for dodo = 1:5
    tic; disp(sprintf('Mulai iterasi ke %d',dodo));
    bit_input(1,1:200000) = data_generator(1,awal:akhr);
    datapar=vec2mat(bit_input,8*188*8);
    uk_datapar=size(datapar);
    out_PRBS=zeros(uk_datapar(1),uk_datapar(2));
    %%%%%%%%%%%%%%%Proses Randomisasi
    tic;
    disp('proses randomisasi')
    for i = 1:uk_datapar(1)
        gen_PRBS=[1 0 0 1 0 1 0 1 0 0 0 0 0 0 0];
        for j =1:uk_datapar(2)
            bb(j)=xor(gen_PRBS(14),gen_PRBS(15));%XOR PRBS bit ke 14 dan 15
            out_PRBS(i,j)=xor(bb(j),datapar(i,j));%XOR input dengan hasil XOR
            PRBS bit 14 dan 15
            gen_PRBS = [bb(j) gen_PRBS(1:14)];%PRBS yang baru
        end
    end
    %out_random2 = vec2mat(out_PRBS,uk_datapar(1)*uk_datapar(2));
    out_random = zeros(1,uk_datapar(1)*uk_datapar(2));
    for i=1:uk_datapar(1)
        for j=1:uk_datapar(2)
            out_random(1,(i-1)*uk_datapar(2)+j) = out_PRBS(i,j);

```

```

end
end
input_RSenc=out_random;
disp('seleseai proses random')
toc;tic;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Proses Outer Coding (Reed Solomon)
disp('proses RS encoder')
n=204; k=188; % panjang coe word dan panjang pesan
m=ceil(log2(n+1)); % jumlah bit per simbol
t=(n-k)/2; % kemampuan koreksi error RS

% Menyesuaikan data sesuai format RS
msg1=vec2mat(input_RSenc,m); % meletakkan m bit tiap 1 baris
msg2=bi2de(msg1,'left-msb'); % mengubah biner ke desimal
msg3=vec2mat(msg2,k); % menempatkan satu word pada baris (1 paket RS)
msg=gf(msg3,m); % membuat galois array dalam GF(2^m)
out_RSenc=rsenc(msg,n,k); % encoding

% Mengubah data dalam format asli (urutan sequence biner)
dcl = double(out_RSenc.x); % data sudah tidak dalam galois field
hasil_RSenc=vec2mat(dcl,1);
disp('seleseai proses pada Reed Solomon')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%Proses outer interleaver
toc;tic;
disp('Proses Outer interleaving dan inner coding')
input_inv=hasil_RSenc';
input_inv_biner = de2bi(input_inv,'left-msb');
input_inv_biner = vec2bi(input_inv_biner,1);
bit=187*12; %zero padding agar data bisa keluar semua dari outer interleaver
bit_inv=[input_inv zeros(1,bit)];
delay = [0 17 34 51 68 85 102 119 136 153 170 187];
out_inv = muxintrlv(bit_inv,delay);

```

```

buffer_outer_interleaver=de2bi(out_inv,m,'left-msb');
out_inv_seri=ve2mat(buffer_outer_interleaver,1);

%%%%%%%%%%%%%% Proses inner coding
input_in_code = reshape( out_inv_seri,length(out_inv_seri),1 );
trell = poly2trellis([7],[ 171 133 ]); % Define Trellis
code = convenc(input_in_code,trell); % code rate 1/2
hasil_konvolusi = code;
tcode = -2* code+ 1;
punctcode = tcode;
punctcode(3:6:end)=[]; % code rate 3/4
punctcode(5:5:end)=[]; % code rate 3/4
hasil_konvolusi = ( 1-punctcode )/2;
disp('Selesai Proses inner coding dan outer interleaving')
%%%%%%%%%%%%%% Proses inner interleaver
toc;tic;
disp('Masuk Proses inner interleaving')
input_in_inter=hasil_konvolusi';
hsl_konv = round(length(input_in_inter)/4);
sisa_konv =mod(length(input_in_inter),4);
if sisa_konv>0
    input_in_inter((hsl_konv+1)*4)=0;
end

% menulis data per kolom dengan kedalaman I=4
I=4;
input_in_intrlv=input_in_inter;
intrlv_kolom=reshape(input_in_intrlv,I,length(input_in_intrlv)/I);

%membaca data per basis
intrlv_baris =vec2mat(intrlv_kolom,1);
disp('Selesai Proses inner interleaving')

```

```
%%%%%%%%%%%%%% mapping 16 QAM
```

```
toc;tic;
```

```
disp('Proses mapping 16 QAM')
```

```
mau_map = round(length(intrlv_baris)/4);
```

```
sisa_mau_map = mod(length(intrlv_baris),4);
```

```
if sisa_mau_map>0
```

```
    intrlv_baris((mau_map+1)*4)=0
```

```
end
```

```
intrlv_baris = intrlv_baris';
```

```
M = 16;    % 16 QAM
```

```
kk = log2(M);    % jumlah bit per simbol
```

```
nsamp = 8; % kalo pake filter 4
```

```
% Penyusun kode Gray konstelasi 16-QAM
```

```
mapping = [0 1 3 2 4 5 6 7 12 13 15 14 8 9 11 10].?;
```

```
uk_intlv_baris = size(intrlv_baris);
```

```
% menata urutan data dalam 4 bit (16 QAM)
```

```
input_map =
```

```
bi2de(reshape(intrlv_baris,kk,length(intrlv_baris)/kk).?',left-msb');
```

```
%Map from binary coding to Gray coding.
```

```
input_map = mapping(input_map+1);
```

```
modul_qam=qammod(input_map,16);
```

```
%scatterplot(modul_qam);
```

```
inphase = real(modul_qam);
```

```
quadrature = imag(modul_qam);
```

```
disp('Selesai Proses mapping')
```

```
%%%%%%%%%%%%%% Pembangkit OFDM
```

```
toc;tic;
```

```
Disp('Proses OFDM')
```

```
Kmax = 6187;    % banyaknya subcarrier (8k mode)
```

```
FS = 8192*16;
```

```
G = 1/16;
```

```
% G = 1/32;    % guard interval 1/16
```

```
input_ofdm = modul_qam;
```

```

simbol_input = vec2mat(input_ofdm,Kmax); % Serial to paralel
uk_simbol_input = size(simbol_input);
simbol_paralel = zeros(uk_simbol_input(2),uk_simbol_input(1));
for i = 1:uk_simbol_input(1)
    for j = 1:uk_simbol_input(2)
        simbol_paralel(j,i) = simbol_input(i,j);
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Proses IFFT
ifft_out=ifft(simbol_paralel,FS);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Penambahan guard interval
titik_guard = G*FS; % Banyaknya titik (1/16)* FS
guard_time = zeros(titik_guard,uk_simbol_input(1));
for i = 1:uk_simbol_input(1)
    for j = 1:titik_guard
        guard_time(j,i) = ifft_out(FS-titik_guard+j,i);
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Simbol Total = Simbol + Guard interval
simbol_total = zeros(titik_guard+FS,uk_simbol_input(1));
for i = 1:uk_simbol_input(1)
    simbol_total((1:titik_guard),i) = guard_time((1:titik_guard),i);
    simbol_total((titik_guard+1):(titik_guard+FS),i) = ifft_out(1:FS,i);
end
uk_simtot = size(simbol_total);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Paralel to Serial
tx = reshape(simbol_total,1,uk_simtot(1)*uk_simtot(2));
disp('Selesai Proses Pada transmiter')
toc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Kanal

%-----

tic;

p= 1

for snr = 0:4:36

disp('Masuk Kanal multipath')

%path 1 (tidak terdelay dan path gain 0 db)

Inpashe 1=(randn(1, length(tx)))*0.707;

Quadrature 1 = (randn(1, length(tx)))*0.707;

envelope 1 = sqrt(Inpashe1.^2 + Quadrature 1.^2);

rms_envelope1 = sqrt(mean(envelope1.^2));

envelope_1 = envelope1/rms_envelope1;

hasil_p1 = envelope_1.*tx;

%path2 (terdelay 135 us dan path gain - 15db)

geser = round((77/896)*FS);

txx2 zeros(1, length(tx));

txx2(1,geser+1: length(tx)) = tx(1,(1:length(tx)-geser));%delay 0.4us =

delay 5 titik

Pathgain2 = 10^(-30/10);

inpashe2 = (randn(1, length(tx)))*0.707;

Quadrature2 = (randn(1, length(tx)))*0.707;

Envelope2 = sqrt(Inpashe length2.^2 + Quadrature2.^2);

rms_envelope2 = sqrt(mean(envelope2.^2));

envelope_2 = envelope2/rms_envelope2;

hasil_p2 = envelope_2.*txx2* Pathgain2;

%path3 (terdelay 135 us dan path gain - 15db)

geser = round((147/896)*FS);

txx3 = zeros(1, length(tx));

txx3(1,geser+1 length(tx)) = tx(1,(1:length(tx)-geser)); %delay 0.4 us = delay 5 titik

```

Pathgain2 = 10^(-36/10);

Inpashe3 =(radn(1,length(tx)))*0.707;
Quadrature3 = randn(1,length(tx))*0.707;
envelope3 = sqrt(Inpashe2.^2 + Quadrature2.^2);
rms_envelope3 = sqrt (mean(envelope2.^2));
envelope_3 = envelope2/rms_envelope2;
hasil_p3 = envelope_2.*tx2*Pathgain2;

tx_tot = hasil_p1+hasil_p2+hasil_p3;

disp('Keluar Kanal multipath')
toc;
%-----
%%%%%%%%%%
Receiver
%-----
tic;
disp(sprintf('Mulai FFT data dengan snr %d',snr));
tx_noise = awgn(tx_tot,snr,'measured'); % add AWGN @ receiver
rx_par = reshape(tx_noise,uk_simtot(1),uk_simbol(2));%Serial to paralel
%Pengambilan Guard Time
simbol_awal = zeros(Fs,uk_simbol_input(1));
for i = 1:uk_simbol_input(1)
    for j = 1:FS
        fft_in(j,i) = rx_part(titik_guard+j,i);
    end
end
%Input FFT
uk_fft = size(fft_in);

%Proses FFT
fft_out = fft(fft_in,FS);

```

```

%Data Paralel
data_par = zeros(Kmax,uk_fft(2));
for i =1:uk_fft(2);
    data_part(1:Kmax,i) = fft(1:Kmax,i);
end
ukur_par = size(data_par);
data_ser = reshape(data_par,ukur_part(1)*ukur_par(2),1);
data_serial = zeros(length(input_ofdm),1);
data_serial((1:length(input_ofdm)),1);
data_ser((1:length(input_ofdm)),1);
disp('Selesai proses FFT')
toc;tic;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DEMAPPING 16-QAM
disp('Proses demapping 16 QAM')
% scatterplot(yrx);
out_demap = qamdemod(data_serial,16);
% Symbol-to-Bit Mapping
% A. Define a vector that inverts the mapping operation.
[dummy demapping] = sort(mapping);
% Map between Gray an binary coding.
out_demap = demapping(out_demap+1);
% Do ordinary decimal-to-binary mapping.
z = de2bi(out_demap = demp,'left-msb');
% Convert z from a matrix to a vector.
z = reshape(z.',prod(size(z)),1);
input_deinter = z;
uk_input_deinter = size(input_deinter);
disp('Selesai Proses Demapping')
toc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Inner Deinterleaver
tic;
disp('Masuk Proses inner decoding dan deinterleaving')
% Menyusun data dalam kolom kedalaman 1

```

```

deinter_kolom = ver2mat(input_deinter,length(input_deinter)/4);
deinter_baris = reshape(deinter_kolom,1,length(deinter_kolom)*4),%
Menyusun kembali data dalam urutan serial yang benar
%%%%%%%%%%%%%% Inner Decoding
input_inner_decode = deinter_baris;
ncode = -2*input_inner_decode+1
nicode = zeros(2*len,1);
nicode(1:6:end) = ncode(1:4:end) ; % Write actual data
nicode(2:6:end) = ncode(2:4:end) ; % Write actual data
nicode(4:6:end) = ncode(3:4:end) ; % Write actual data
nicode(5:6:end) = ncode(4:4:end) ; % Write actual data
nicode = (1-nicode)/2;
in_decode = vitdec(nicode,trell,14,'trunc','hard'); % Decode 1/2
in_decode = vitdec(input_inner_decode,trell,2,'trunc','hard');% Decode 3/4
%%%%%%%%%%%%%% Outer Deinterleaver
In_decode2 = vec2mt(in_decode,8);
input_out_deinter = bi2de(in_decode2,'left-msb'); %Mengubah ke desimal
out_ouintr = muxdeintrlv(input_out_deinter,delay);
data_ouintr = out_ouintr(bit+1 :lngth(bit_inv),1);
data_ouintr_biner = de2bi(data_ouintr,'left-msb');
data_ouintr_biner = vec2mat(data_ouintr_biner,1);
%%%%%%%%%%%%%% Outer Decoder (RS Decoder)
Input_RSdec = data_ouintr;
RS_dec1 = vec2mat(input_RSdec,n);
RS_dec2 = gf(RS_dec1,m);
out_RSdec = rsdec(RS_dec2,n,k);
% Mengubah data kembali ke bentuk asli
out_RSdec2 = double(out_RSdec.x);
out_RSdec3 = vec2mat(out_RSdec2,1);
out_RSdec4 = de2bi(out_ouintr_RSdec3,m,'left-msb');
biner_out_RSdec2 = vec2mat(out_RSdec4,1);
biner_out_RSdec = biner_out_RSdec2';
disp('Selesai Proses decoding dan deinterleaving')

```

```

toc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Proses Derandomizer
tic;
disp('Proses Derandomisasi')
in_derand = vec2mat(biner_out_RSdec,8*188*8);
out_PRBS_derandom = zeros(uk_datapar(1),uk_datapar(2));
%in_derand = vec2mat(biner_out_RSdec,8*188*8);
for i =1:uk_datapar(1)
    gen_PRBS=[1 0 0 1 0 1 0 1 0 0 0 0 0 0 0];
    for j =1:uk_datapar(2)
        bb(j) = xor(gen_PRBS(14),gn_PRBS(15)); %XOR PRBSbit ke 14 dan 15
        out_PRBS_derandom(i,j) = xor(bb(j),in_derand(i,j)); %XOR input dengan hasil
        XOR PRBS bit 14 dan 15
        gen_PRBS = [bb(j) gen_PRBS(1:14)]; % PRBS yang baru
    end
end
derabdom = out_PRBS_derandom;
derandom2 = vec2mat(derandom,1);
derandom3 = zeros(length(bit_input),1);
for i = 1:(length(bit_input))
    derandom3(i,1) = derandom2(i,1);
end
derandom4 = derandom3';
%out_derandom = vec2mat(derandom4,8); bit input diganti sementara
Disp('Selesai Proses')
toc;

% cek kebenaran
% [Abit,Bber] = biterr(derandom,datapar);
[Abit,Bber] = biterr(data_ouintr_biner,input_inv_biner);
s(p,1)=snr;
c(p,1)=Abut
p=p+1

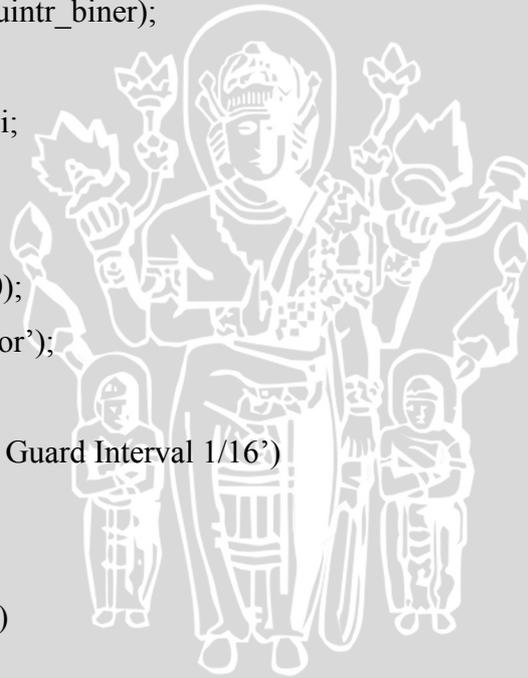
```

```

end
posisi(:,dodo) = c(:,1);
awl = awl+200000;
akhr = akhr+200000;

clear bit_input datapar c tx input_in_code_in_decode out_random
input_RSenc input_RSenc input_ofdm simbol_input simbol_paralel
end
hasil = zeros(10,1);
for jk = 1:10
    hasil(jk,1) = sum(posisi(jk,:));
end
pembagi = length(data_ouintr_biner);
pembagi = pembagi*5;
ber_total = hasil/ pembagi;
Snr = 0:4:36;
figure(1);
ber_total = ber_total(1:10);
semilogy(Snr,ber_total,'-or');
grid on;
title('BER After Viterbi, Guard Interval 1/16')
xlabel('SNR(dB)');
ylabel('BER');
axis([0 37 0.00000001 1])

```



LAMPIRAN C

TABEL HASIL PEMODELAN SISTEM

C.1. Nilai BER rata-rata Sistem dengan *code rate* 1/2, *Guard Interval* 1/16

No	SNR	BER rata-rata
1	0	0,0434
2	4	0,0035
3	8	0,0002
4	12	4,595E-05
5	16	1,225E-05
6	20	4,685E-06
7	24	4,145 E-06
8	28	4,145 E-06
9	32	4,565 E-06
10	36	4,595 E-06

C.2. Nilai BER rata-rata Sistem dengan *code rate* 1/2, *Guard Interval* 1/32

No	SNR	BER rata-rata
1	0	0,0461
2	4	0,0040
3	8	0,0003
4	12	4,703E-05
5	16	1,910 E-05
6	20	8,470E-06
7	24	4,865 E-06
8	28	9,371 E-06
9	32	9,731 E-06
10	36	5,406 E-06

C.3. Nilai BER rata-rata Sistem dengan *code rate* 3/4, *Guard Interval* 1/16

No	SNR	BER rata-rata
1	0	0,2358
2	4	0,0250
3	8	0,0018
4	12	0,0002
5	16	8,830E-05
6	20	3,622E-05
7	24	3,045E-05
8	28	2,919E-05
9	32	2,781 E-05
10	36	2,698 E-05

C.4. Nilai BER rata-rata Sistem dengan *code rate* 3/4, *Guard Interval* 1/32

No	SNR	BER rata-rata
1	0	0,2415
2	4	0,0269
3	8	0,0020
4	12	0,0002
5	16	8,426E-05
6	20	5,116E-05
7	24	4,992 E-05
8	28	5,857 E-05
9	32	4,072 E-05
10	36	4,989 E-05