

SIMULASI EQUALISASI ADAPTIF DENGAN MENGGUNAKAN ALGORITMA RECURSIVE LEAST SQUARES (RLS) PADA QPSK

Ariyanto Wijaya Saputro¹, Bambang Sumajudin², Achmad Rizal³

^{1,2,3}Jurusan Teknik Elektro, Sekolah Tinggi Teknologi Telkom
ar_riyan17@yahoo.com¹, sumajudin@stttelkom.ac.id², arz@stttelkom.ac.id³

Abstrak

Permasalahan utama dalam sistem komunikasi digital adalah diperlukannya bandwidth yang lebar untuk proses transmisi. Oleh karena itu perlu dilakukan pembatasan bandwidth pada bagian pengirim. Namun pembatasan bandwidth ini menyebabkan terjadinya tumpang tindih antar simbol yang bersebelahan. Kondisi ini disebut dengan *Intersymbol Interference (ISI)*.^[7] Pada modulasi *Quadrature Phase Shift Keying (QPSK)* selain permasalahan ISI, juga terdapat permasalahan lainnya yaitu adanya *Crosstalk*. *Crosstalk* terjadi karena pengaruh interferensi antara kanal In-phase dan kanal Quadrature. *Noise*, *ISI*, dan *crosstalk* menyebabkan terjadinya kesalahan deteksi pada penerima, sehingga nilai BER menjadi lebih besar. Untuk mengatasi *noise*, *ISI* dan *Crosstalk* maka digunakan equalizer adaptif. Ada beberapa algoritma yang biasa digunakan, diantaranya adalah *Least Mean Square (LMS)* dan *Recursive Least Squares (RLS)*. Dalam Tugas Akhir ini akan disimulasikan dan dianalisis kinerja equalisasi adaptif dengan algoritma RLS pada demodulator QPSK dalam mengatasi *noise* dan *ISI*. Agar equalizer adaptif dapat bekerja dengan baik, maka parameter-parameter equalizer perlu diset terlebih dahulu. Dari hasil simulasi diperoleh kesimpulan bahwa orde yang memberikan hasil optimal adalah orde 4. Untuk periode perubahan kanal setiap 50 simbol, maka periode perubahan bobot yang memberikan hasil optimum adalah setiap 10 simbol. Pada tugas akhir ini algoritma RLS konvergen pada iterasi ke 10. Untuk kondisi kanal dengan SNR 15 dB nilai ISI pada input equalizer adalah 67 %. Setelah melalui proses equalisasi nilai ISI turun menjadi 38 %. Nilai BER untuk demodulator QPSK tanpa equalizer adalah $1,5 \times 10^{-1}$, jika demodulator QPSK dipasang equalizer adaptif, nilai BER turun menjadi 6×10^{-3} .

Kata Kunci : Equalisasi adaptif, RLS, ISI, QPSK.

1. Pendahuluan

Permasalahan utama dalam sistem komunikasi digital adalah diperlukannya *bandwidth* yang lebar untuk proses transmisi. Oleh karena itu perlu dilakukan pembatasan bandwidth dengan menggunakan *pulse shaper*. Filter pembatas tersebut mengakibatkan terjadinya pelebaran pulsa, sehingga memungkinkan terjadinya interferensi antar simbol yang berdekatan. *ISI* selain ditimbulkan oleh efek pemfilteran pada pengirim juga bisa ditimbulkan karena kondisi kanal yang *multipath*.^[7] Untuk mengurangi *ISI* dapat diupayakan dengan menentukan filter pada pengirim dan filter pada penerima. Namun kondisi kanal transmisi yang berubah-ubah terhadap waktu (*time varying*) menyebabkan sinyal terima bersifat non-stasioner. Sehingga penggunaan filter pada pengirim dan penerima tidak cukup handal untuk mengurangi *ISI*.^[7] Maka dibutuhkan sebuah filter adaptif yang mampu menyesuaikan dengan kondisi kanal yang berubah-ubah tersebut. Filter adaptif ini biasa disebut dengan equalizer. Dengan menyesuaikan koefisien filter transversal secara dinamik, yaitu dengan menggunakan algoritma tertentu, maka bisa menanggulangi efek kanal.^[11] Algoritma yang biasa digunakan adalah *Least Mean Square (LMS)* dan *Recursive Least Squares (RLS)*. Pada penelitian sebelumnya telah dibahas equalisasi

adaptif dengan metode LMS^[7], sehingga pada penelitian ini algoritma yang akan digunakan adalah algoritma RLS. Dengan algoritma RLS, maka akan diperoleh kesalahan komputasi yang lebih kecil. Selain itu algoritma RLS mempunyai laju konvergensi yang lebih cepat dibanding LMS. Kelemahannya adalah algoritma ini lebih kompleks.^[11]

Permasalahan yang menjadi perhatian dalam penelitian ini antara lain :

- 1 Bagaimana pemodelan modulasi dan demodulasi QPSK yang disertai efek *noise*, *crosstalk*, dan *ISI* di bagian pengirim, kanal transmisi, dan penerima?
- 2 Bagaimana pemodelan demodulator QPSK yang ditambahkan equalizer adaptif dengan algoritma RLS?
- 3 Bagaimana kinerja sistem equalisasi adaptif dengan algoritma RLS dalam mengurangi efek *ISI*, *crosstalk*, dan *noise* AWGN?
- 4 Bagaimana pengaruh terhadap kinerja equalizer jika parameter-parameter equalizer diubah-ubah nilainya?

2. Dasar Teori

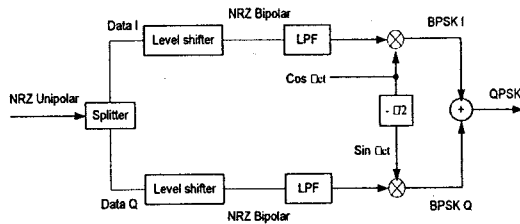
2.1 Quadrature Phase Shift Keying

Modulasi QPSK merupakan teknik modulasi yang menumpangkan sinyal informasi

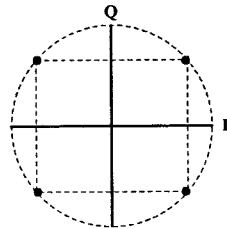
pada fasa dari sinyal *carrier*. Modulasi QPSK terdiri dari 4 simbol yaitu 00, 01, 10, dan 11. Antar simbol yang berdekatan berbeda fasa sebesar 90° . Persamaan sinyal QPSK secara umum :

$$s_{QPSK}(t) = \sqrt{\frac{2E_s}{T_s}} \cos \left[2\pi f_c t + (i-1) \frac{\pi}{2} \right] \quad 2.1$$

Blok diagram modulator QPSK ditunjukkan pada gambar 1. berikut:



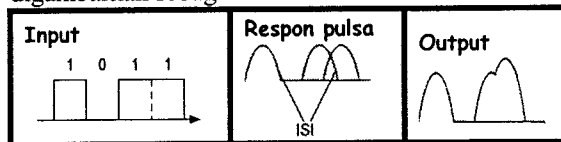
Gambar 1. Blok diagram modulator QPSK^[9]



Gambar 2. Diagram konstelasi QPSK

2.2 Inter Symbol Interference

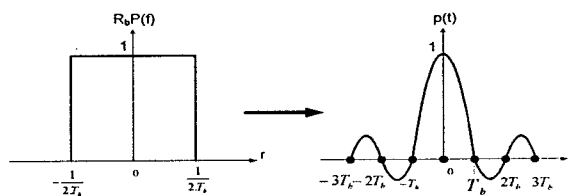
Karena lebar bandwidth sinyal NRZ unipolar tak hingga, maka sebelum dikirimkan bandwidthnya harus dibatasi. Pembatasan bandwidth menyebabkan terjadinya interferensi antar simbol yang berdekatan (*Inter Symbol Interference*). Proses terjadinya ISI bisa digambarkan sebagai berikut



Gambar 3. Terjadinya Intersymbol Interferensi

Untuk memperoleh sinyal yang bebas ISI, digunakan metode pertama Nyquist dengan menggunakan fungsi transfer ekuivalen $H_c(f)$ sedemikian rupa sehingga respon pulsanya memenuhi kondisi sebagai berikut :

$$P(f) = \begin{cases} T_b; & -\frac{R_b}{2} < f < \frac{R_b}{2} \\ 0; & |f| > \frac{R_b}{2} \end{cases} \quad 2.2$$



Gambar 4. Respon Filter LPF Nyquist

Respon pulsa ini memenuhi syarat untuk kriteria nyquist untuk kondisi bebas ISI : “Jika impuls-impuls sinkron berkecepatan $R_b = 1/T_b$ simbol/detik dapat ditransmisikan tanpa ISI melalui suatu kanal low pass ideal *brick-wall* berfasa linier

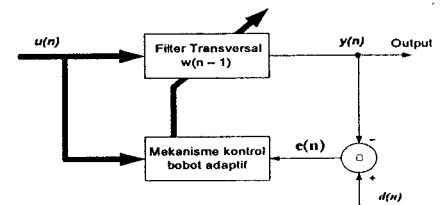
dengan batas minimum frekuensi *cut off* $f_c = R_b/2 = 1/2T_b$ ”^[5]

Namun untuk merealisasikan filter semacam ini sangat sulit, karena karakteristik fungsi transfer $H_c(f)$ harus mempunyai amplitude yang flat untuk frekuensi tertentu ($-BW < f < BW$) dan bernilai nol untuk frekuensi di luar frekuensi tersebut.^[5] Maka solusinya dibuat fungsi transfer $H_c(f)$ sebagai fungsi filter “*raised cosine*” dengan fungsi transfer sebagai berikut:

$$H(f) = \begin{cases} 1 & ; 0 < f \leq f_N(1-\alpha) \\ 0.5(1 - \sin \frac{\pi}{2\alpha} (\frac{f}{f_N} - 1)) & ; f_N(1-\alpha) < f \leq f_N(1+\alpha) \\ 0 & ; f_N(1+\alpha) < f \end{cases} \quad 2.3$$

2.5 Equalizer Adaptif

Pada proses modulasi sering terjadi *distorsi* yang mengakibatkan terjadinya kesalahan deteksi pada penerima. *Distorsi* tersebut bisa disebabkan karena adanya ISI, *crosstalk*, dan *noise*. ISI terjadi karena pengaruh pembatasan *bandwidth* dan pengaruh kanal *multipath*. Untuk mengurangi ISI bisa dilakukan dengan memasang filter tertentu pada pengirim dan penerima. Dalam kenyataannya kanal transmisi kondisinya berubah sehingga penggunaan sepasang filter pada pengirim dan penerima yang didesain berdasarkan rata-rata kondisi kanal, tidak cukup handal untuk mengurangi ISI. Oleh karena itu diperlukan sebuah filter adaptif yang mampu menyesuaikan dengan kondisi kanal yang berubah-ubah tersebut. Filter adaptif ini biasa disebut dengan equalizer.^[7]



Gambar 5. Blok diagram filter adaptif^[11]

Dengan menyesuaikan koefisien filter transversal secara dinamik (dengan menggunakan algoritma tertentu) bisa menanggulangi efek kanal.^[11] Algoritma yang biasa digunakan adalah *Least Mean Square* dan *Recursive Least Squares*.

2.6 Recursive Least Squares

Algoritma RLS mempunyai laju konvergensi yang lebih cepat juga tingkat kesalahan yang lebih kecil dibanding LMS. Kelemahan RLS adalah memerlukan proses komputasi yang lebih kompleks dibandingkan LMS.^[11]

Inti dari metode RLS adalah berusaha meminimalisasi nilai *cost function* ($J(n)$), yang didefinisikan sebagai penjumlahan dari kuadrat error.^[1]

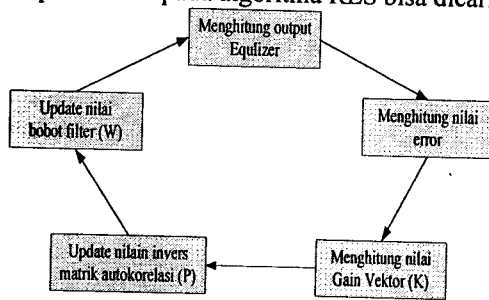
$$J(n) = \sum_{i=1}^n \beta(n,i) |e(i)|^2 \quad 2.4$$

$\beta(n,i)$ melambangkan *forgetting factor* yang nilainya antara 0 dan 1. Bentuk yang biasa

digunakan untuk *forgetting factor* adalah bentuk *exponensial*. Sehingga persamaan *cost function* yang baru adalah: [1]

$$J(n) = \sum_{i=1}^n \lambda^{n-i} |e(i)|^2 \quad 2.5$$

Kondisi $J(n)$ minimum terjadi saat turunan pertama $J(n)$ terhadap $w(n)$ bernilai nol. Dengan menggunakan bantuan *matrix inversion lemma* [1], maka persamaan pada algoritma RLS bisa dicari.



Gambar 6. Diagram Alir Algoritma RLS [14]

Proses adaptif algoritma RLS mempunyai step-step sebagai berikut:

- i) Inisialisasi bobot awal filter dan $P(0)$. Nilai $P(0) = \lambda^{-1} I$. I merupakan matriks identitas, λ merupakan konstanta positif yang nilainya kecil.
- ii) Menghitung nilai estimasi error:

$$\hat{d}(n) = d(n) - w^H(n-1)u(n) \quad 2.6$$
- iii) Menghitung nilai gain vector:

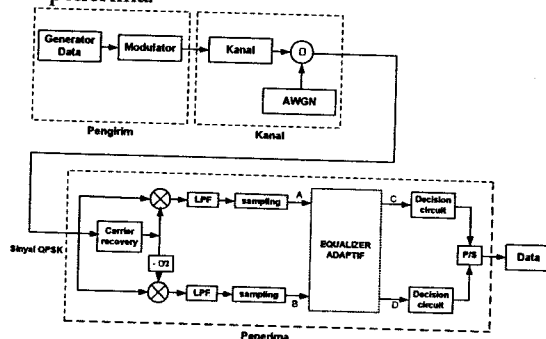
$$k(n) = \frac{\lambda^{-1} P(n-1)u(n)}{1 + \lambda^{-1} u^H(n)P(n-1)u(n)} \quad 2.7$$
- iv) Update nilai invers matrik autokorelasi:

$$P(n) = \lambda^{-1} P(n-1) - \lambda^{-1} k(n)u^H(n)P(n-1) \quad 2.8$$
- v) Update nilai bobot filter:

$$w(n) = w(n-1) + \lambda^{-1} k(n) \quad 2.9$$
- vi) Perbesar nilai n , lalu kembali ke step ii)

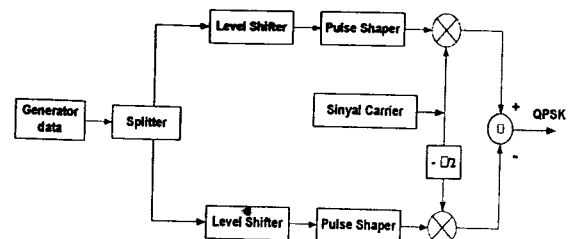
3. Simulasi

Secara garis besar sistem yang akan disimulasikan pada penelitian ini dibagi menjadi tiga bagian utama, yaitu bagian pengirim, kanal, dan penerima.



Gambar 7. Blok Simulasi Secara Umum

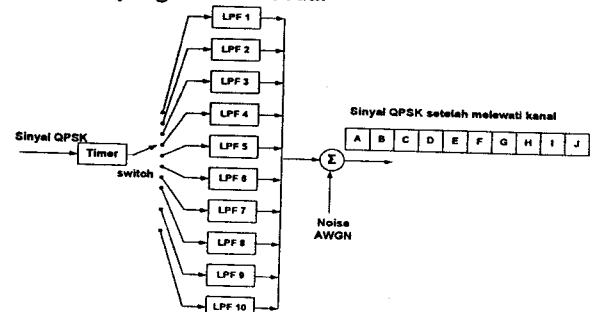
3.1 Pemodelan di Bagian Pengirim



Gambar 8. Blok Simulasi Bagian Pengirim

3.2 Pemodelan Kanal

Kanal yang digunakan pada penelitian ini merupakan kombinasi dari 10 LPF butterworth dengan nilai frekuensi *cutoff* dan frekuensi *stopband* yang berbeda-beda.



Gambar 9. Blok Simulasi Bagian Kanal

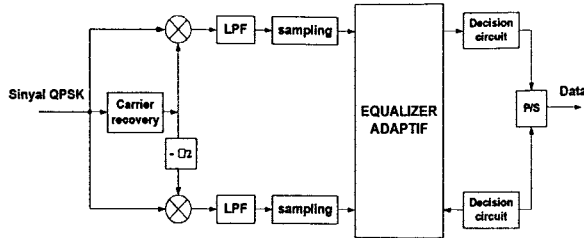
Latar belakang digunakan kanal model ini adalah ketika sinyal dilewatkan pada LPF, maka output yang dihasilkan akan mengalami redaman dan *delay*. Jika spesifikasi filter berbeda-beda, maka *delay* dan redaman yang terjadi juga berbeda-beda pula. Pemodelan kanal seperti ini dianggap mewakili kondisi kanal *multipath*, yang bisa menambah ISI. Dengan pemodelan kanal seperti ini periode perubahan kanal bisa diatur apakah ingin cepat atau lambat. Sehingga bisa diketahui seberapa cepat periode perubahan kanal yang masih mampu ditangani equalizer tersebut.

Tabel 1. Spesifikasi LPF yang digunakan

| | Jenis LPF | Frek. Cutoff | Frek. Stopband |
|--------|--------------------|--------------|----------------|
| LPF 1 | Butterworth orde 1 | 22,5 kHz | 50 kHz |
| LPF 2 | Butterworth orde 1 | 25 kHz | 50 kHz |
| LPF 3 | Butterworth orde 1 | 27,5 kHz | 50 kHz |
| LPF 4 | Butterworth orde 1 | 30 kHz | 50 kHz |
| LPF 5 | Butterworth orde 1 | 32,5 kHz | 50 kHz |
| LPF 6 | Butterworth orde 1 | 60 kHz | 200 kHz |
| LPF 7 | Butterworth orde 1 | 80 kHz | 200 kHz |
| LPF 8 | Butterworth orde 1 | 100 kHz | 200 kHz |
| LPF 9 | Butterworth orde 1 | 110 kHz | 200 kHz |
| LPF 10 | Butterworth orde 1 | 120 kHz | 200 kHz |

3.3 Pemodelan di Bagian Penerima

Model penerima yang digunakan adalah demodulator QPSK yang sudah diintegrasikan dengan equalizer adaptif.



Gambar 10. Blok Simulasi Bagian Penerima

3.3.1 Pemodelan equalizer adaptif

Pemodelan equalizer adaptif pada penelitian ini diturunkan dari rumus 2.31, 2.32, 2.34, dan 2.35 yang parameternya dimodifikasi ke dalam bentuk kompleks.

- Input equalizer : $u(n) = U_I(n) + jU_Q(n)$
 Desire respon : $d(n) = d_I(n) + jd_Q(n)$
 Koefisien filter : $w(n) = w_I(n) + jw_Q(n)$
 Output Equalizer : $Y(n) = Y_I(n) + jY_Q(n)$
 Estimasi error : $e(n) = e_I(n) + je_Q(n)$

Setelah dilakukan penurunan rumus, maka rumus equalizer dengan algoritma RLS untuk kondisi parameter kompleks adalah sebagai berikut:

$$e_I(n) = d_I(n) - (w_I^T(n-1) \cdot U_I(n) - w_Q^T(n-1) \cdot U_Q(n)) \quad 3.1$$

$$e_Q(n) = d_Q(n) - (w_I^T(n-1) \cdot U_Q(n) + w_Q^T(n-1) \cdot U_I(n)) \quad 3.2$$

$$w_I(n) = w_I(n-1) + K_I(n) \cdot e_I(n) - K_Q(n) \cdot e_Q(n) \quad 3.3$$

$$w_Q(n) = w_Q(n-1) + K_I(n) \cdot e_Q(n) + K_Q(n) \cdot e_I(n) \quad 3.4$$

$$P_I(n) = \lambda^{-1} \cdot P_I(n-1) - \lambda^{-1} \cdot P_I(n-1) (K_I(n) U_I^T(n) - K_Q(n) U_Q^T(n)) + \lambda^{-1} \cdot P_Q(n-1) (K_I(n) U_Q^T(n) + K_Q(n) U_I^T(n)) \quad 3.5$$

$$P_Q(n) = \lambda^{-1} \cdot P_Q(n-1) - \lambda^{-1} \cdot P_Q(n-1) (K_I(n) U_I^T(n) - K_Q(n) U_Q^T(n)) - \lambda^{-1} \cdot P_I(n-1) (K_I(n) U_Q^T(n) + K_Q(n) U_I^T(n)) \quad 3.6$$

$$K_I(n) = \frac{(AxC) + (BxD)}{C^2 + D^2} \quad 3.7$$

$$K_Q(n) = \frac{(BxC) - (AxD)}{C^2 + D^2} \quad 3.8$$

$$A = (P_I(n-1) \cdot U_I(n) - P_Q(n-1) \cdot U_Q(n)) \quad 3.9$$

$$B = (P_I(n-1) \cdot U_Q(n) + P_Q(n-1) \cdot U_I(n)) \quad 3.10$$

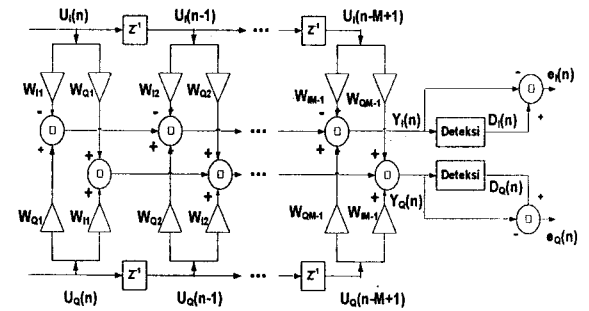
$$C = \lambda + (U_I^T(n) \cdot P_I(n-1) - U_Q^T(n) \cdot P_Q(n-1)) \cdot U_I(n) - (U_I^T(n) \cdot P_Q(n-1) + U_Q^T(n) \cdot P_I(n-1)) \cdot U_Q(n) \quad 3.11$$

$$D = (U_I^T(n) \cdot P_I(n-1) - U_Q^T(n) \cdot P_Q(n-1)) \cdot U_Q(n) + (U_I^T(n) \cdot P_Q(n-1) + U_Q^T(n) \cdot P_I(n-1)) \cdot U_I(n) \quad 3.12$$

Proses equalisasi adaptif dalam penelitian ini mempunyai tahapan-tahapan sebagai berikut:

- i) Inisialisasi *forgetting factor* (\square), kondisi bobot awal dan invers matriks autokorelasinya. Kondisi awal yang digunakan adalah:
 - $\square = 1$.
 - $\square = 0,004$.
 - $P_i = (1/\square) \cdot I$.
 - $P_q = (1/\square) \cdot I$.
 - $W_i = 0,4 \cdot \text{rand}(N,1)$.
 - $W_q = -0,2 \cdot \text{rand}(N,1)$.
 - Ket = *rand* membangkitkan bialangan *random*.
- ii) Menghitung nilai *estimasi error* untuk lengan *inphase* dan lengan *quadrature*. Rumus yang digunakan adalah persamaan 3.1 dan 3.2.

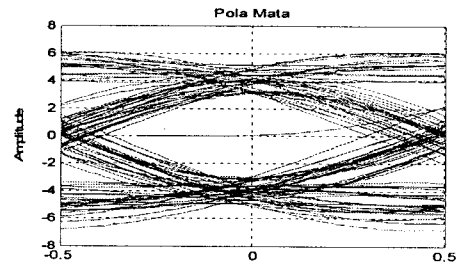
- iii) Update nilai *gain vector* ($K(n)$) untuk lengan *inphase* dan *quadrature*. Rumus yang dipakai adalah persamaan 3.7 sampai dengan 3.12.
- iv) Update nilai *invers* autokorelasi ($P(n)$) untuk lengan *inphase* dan lengan *quadrature*. Rumus yang digunakan adalah persamaan 3.5 dan 3.6.
- v) Update nilai bobot filter $w(n)$ untuk lengan *inphase* dan lengan *quadrature* dengan menggunakan persamaan 3.3 dan 3.4. Proses updating nilai bobot ($w(n)$) berbeda jika dibandingkan proses updating $K(n)$ dan $P(n)$. Proses updating $K(n)$ dan $P(n)$ dilakukan setiap 1 sampel masuk equalizer, sedangkan updating nilai bobot filter dilakukan setiap 500 sampel (10 simbol) masuk equalizer. Hal ini dilakukan untuk menyesuaikan dengan periode perubahan kanalnya, sehingga proses updating nilai bobot filter tidak terlalu cepat juga tidak terlalu lambat.
- vii) Perbesar nilai n , lalu kembali ke step ii) dan demikian seterusnya.



Gambar 11. Blok Simulasi Equalizer Adaptif

3.4 Menghitung nilai ISI

Dalam penelitian ini untuk menghitung nilai ISI dilakukan dengan pendekatan statistik, standard deviasi.



Gambar 12. Contoh Polamata Dengan ISI 16 %

Jika Y merupakan suatu himpunan dari $y_1, y_2, y_3, y_4, y_5, \dots, y_N$, maka himpunan Y bisa ditulis sebagai berikut : $Y = \{ y_1, y_2, y_3, y_4, y_5, \dots, y_N \}$. Nilai standard deviasi Y bisa hitung dengan menggunakan persamaan berikut ini :

$$\delta(Y) = \sqrt{\left(\frac{1}{N} \sum_{i=1}^N (y_i - \bar{Y})^2 \right)} \quad 3.13$$

\bar{Y} menyatakan nilai rata-rata dari Y . N menyatakan banyaknya anggota Y .

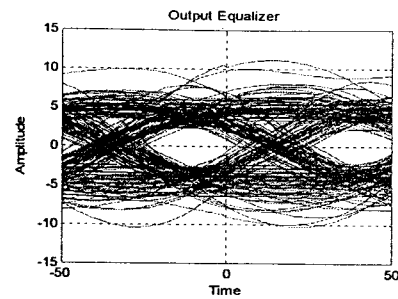
$$\bar{Y} = \frac{y_1 + y_2 + y_3 + y_4 + \dots + y_N}{N} \quad 3.14$$

Nilai ISI merupakan nilai standar deviasi dibagi nilai rata-ratanya.

$$ISI = \frac{\delta(Y)}{Y}$$

3.15

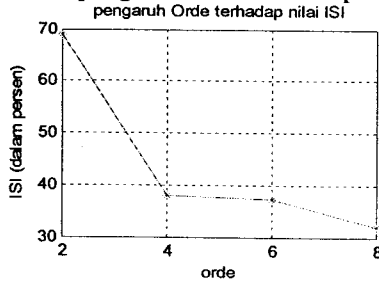
Ket : Nilai $y_1, y_2, y_3, y_4, y_5, \dots, y_N$ merupakan nilai sampel dari keluaran LPF yang disampling dengan periode sampling 5 kHz (satu sampel per simbol).



Gambar 16. output equalizer orde 8

4. Analisis Performansi Sistem.

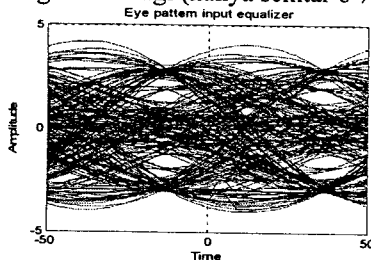
4.1 Analisis pengaruh orde terhadap nilai ISI



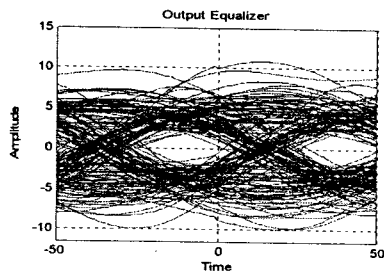
Gambar 13. Pengaruh orde filter terhadap nilai ISI

Berdasarkan gambar 13. dapat ditarik analisis:

- Semakin besar orde filter, maka nilai ISI output equalizer akan semakin kecil.
- Perbaikan yang maksimum tercapai saat orde yang digunakan 8, yaitu 42,496 %.
- Perubahan yang signifikan terjadi saat perubahan dari orde 2 ke orde 4, selisih perbaikannya mencapai 22,936 %. Ketika ordonya ditingkatkan menjadi 6 dan 8 peningkatan perbaikannya ISInya tidak signifikan lagi (hanya sekitar 6 %).

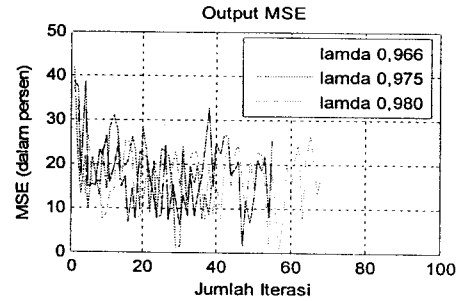


Gambar 14. input equalizer

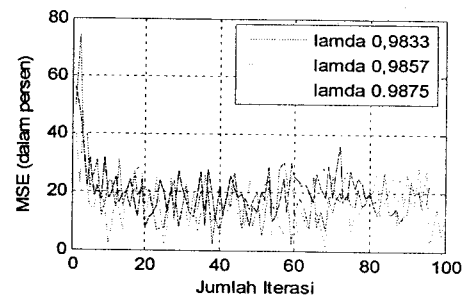


Gambar 15. output equalizer orde 4

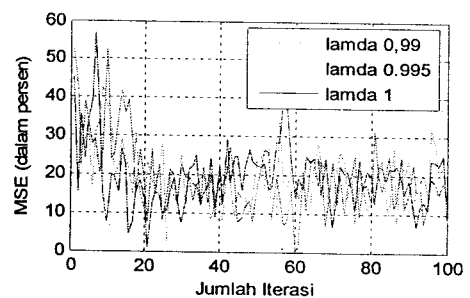
4.2 Analisis pengaruh forgetting factor (\square).



Gambar 17. Ouput MSE untuk nilai \square tertentu



Gambar 18. Ouput MSE untuk nilai \square tertentu



Gambar 19. Ouput MSE untuk nilai \square tertentu

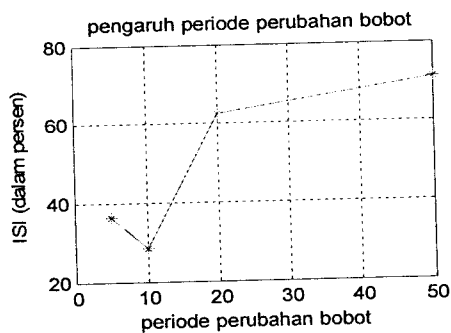
Tabel 2. Pengaruh perubahan *forgetting factor* (\square)

| Nilai Lamda (\square) | Panjang Memori | Berhenti pada iterasi |
|---------------------------|----------------|-----------------------|
| 0,966 | 30 | 40 |
| 0,975 | 40 | 55 |
| 0,98 | 50 | 68 |
| 0,9833 | 60 | 82 |
| 0,9857 | 70 | 96 |
| 0,9875 | 80 | 100 |
| 0,99 | 100 | 100 |
| 0,995 | 200 | 100 |

| | | |
|---|--------------|-----|
| 1 | tak terbatas | 100 |
|---|--------------|-----|

- Penjelasan tabel 2. adalah sebagai berikut:
- Jumlah iterasi total pada simulasi ini adalah 100 iterasi. Nilai ini diperoleh dari 500 simbol dibagi 5 simbol. (data yang dikirimkan 500 simbol, sementara perubahan bobot terjadi setiap 5 simbol).
 - Pada saat nilai $\square = 0,966$, dengan menggunakan rumus diperoleh panjang memorinya 30. Berdasarkan hasil simulasi pada gambar 4.6, pada \square tersebut proses equalisasi terhenti pada iterasi ke 40.
 - Untuk nilai \square 0,966 sampai 0,98577 karena keterbatasan memori, proses equalisasi tidak bisa mencapai iterasi yang seharusnya (berhenti sebelum iterasi ke 100)
 - Untuk nilai \square 0,9875 sampai 1, proses equalisasi berhenti pada iterasi yang seharusnya (100 iterasi).

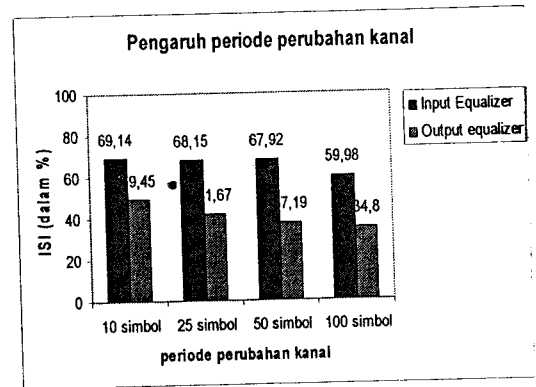
4.3 Analisis pengaruh periode perubahan bobot terhadap nilai ISI.



Gambar 18. Pengaruh nilai orde terhadap ISI

- Berdasarkan gambar 18. bisa disimpulkan :
- Periode perubahan bobot yang optimum terjadi saat periode perubahannya setiap 10 simbol, dengan perbaikan ISI mencapai 44,554 %.
 - Saat periode perubahan bobot dipercepat menjadi setiap 5 simbol, perbaikan ISInya justru mengalami penurunan menjadi 36,462 %.
 - Periode perubahan bobot dengan perbaikan ISI terburuk saat perubahan bobotnya setiap 50 simbol, hanya 1,816 %. Karena periode perubahan bobot sama dengan periode perubahan kanal, maka proses adaptasi bobot sering terlambat.

4.4 Analisis pengaruh periode perubahan kanal terhadap nilai ISI.

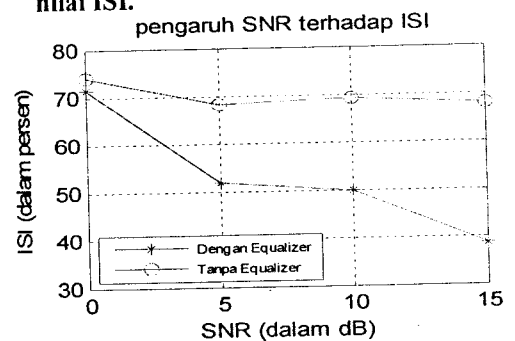


Gambar 19. pengaruh periode perubahan kanal

Berdasarkan gambar 19. bisa ditarik kesimpulan

- Nilai ISI input equalizer terbesar terjadi saat periode perubahan kanal setiap 10 simbol, yaitu 69,1335 %. Pada kondisi ini kanal berubah 50 kali (500 simbol dibagi 10 simbol).
- Semakin lambat periode perubahan kanal, maka ISI yang ditimbulkan (input equalizer) akan semakin kecil.
- Nilai ISI pada input equalizer terkecil terjadi saat periode perubahan kanal setiap 100 simbol, yaitu 59,98 %. Pada kondisi ini kanal hanya berubah sebanyak 5 kali (500 simbol dibagi 100 simbol).
- Untuk semua kondisi periode perubahan kanal yang disimulasikan pada penelitian ini, equalizer adaptif mampu melakukan perbaikan ISI yang cukup signifikan dengan nilai perbaikan bervariasi antara 19,6817 % sampai dengan 30,7215 %.

4.5 Analisis pengaruh noise AWGN terhadap nilai ISI.



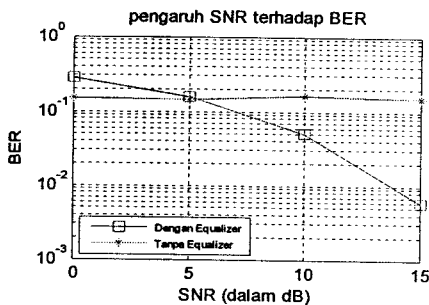
Gambar 20. Pengaruh SNR terhadap nilai ISI

Berdasarkan gambar 20. dapat ditarik kesimpulan sebagai berikut :

- Noise AWGN menyebabkan kinerja equalizer adaptif terganggu. Semakin besar noise (semakin kecil nilai SNR), maka menyebabkan kinerja equalizer semakin jelek, sehingga nilai ISI semakin besar. Demikian juga sebaliknya, semakin besar SNR, maka nilai ISI semakin kecil.

- Pada saat kondisi SNR = 0 dB, perbedaan nilai ISI untuk kondisi tanpa equalizer dan dengan equalizer kurang dari 5 %.
- Semakin besar nilai SNR, perbaikan ISInya semakin besar. Pada saat SNR 5 dB perbedaan nilai ISI antara tanpa equalizer dan dengan equalizer sekitar 15 %. Saat SNR 10 dB perbedaannya performansinya sekitar 20 %, dan saat 15 dB terjadi per-bedaan performansi terbesar, sekitar 30 %.

4.6 Analisis pengaruh noise AWGN terhadap nilai BER.

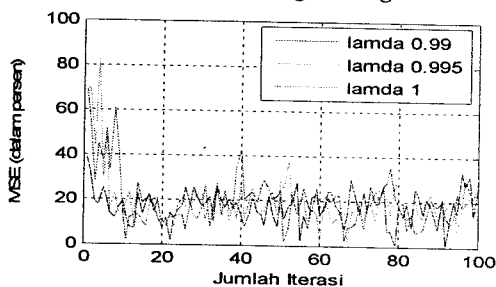


Gambar 21. Pengaruh SNR terhadap nilai BER

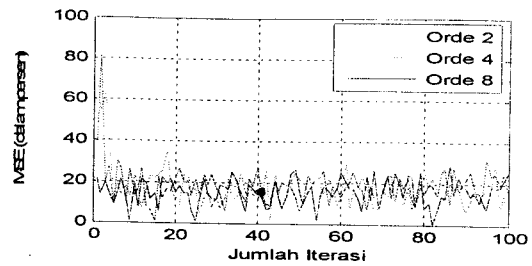
Berdasarkan gambar 21. dapat ditarik kesimpulan sebagai berikut :

- Pada saat kondisi noisanya cukup besar (saat SNR 0 dB dan 5 dB), output demodulator dengan equalizer mempunyai performansi yang lebih jelek dibandingkan dengan demodulator tanpa equalizer. Hal ini dikarenakan adanya noise yang terlalu besar, sehingga proses equalisasi menjadi terganggu sehingga nilai BER tetap tinggi (tidak ada perbaikan BER).
- Saat kondisi SNR 10 dB, mulai terlihat adanya perbaikan. Pada kondisi SNR ini nilai BER tanpa equalizer adalah $1,6 \times 10^{-1}$, sedangkan demodulator dengan equalizer nilai BERnya turun menjadi 5×10^{-2} .
- Saat kondisi SNR 15 dB, perbaikan BERnya antara tanpa equalizer dengan yang menggunakan equalizer cukup besar. Pada saat tidak menggunakan equalizer nilai BERnya $1,5 \times 10^{-1}$. Pada saat menggunakan equalizer nilai BERnya turun drastic menjadi 6×10^{-3} .

4.7 Analisis waktu konvergensi algoritma RLS.



Gambar 22. konvergensi pada 3 kondisi lamda



Gambar 23. konvergensi pada 3 kondisi orde

Berdasarkan gambar 22 dan gambar 23 dapat disimpulkan bahwa algoritma RLS mempunyai waktu konvergensi yang cenderung tetap untuk berbagai kondisi lamda dan orde yang disimulasikan. Pada kedua simulasi tersebut waktu konvergensinya adalah pada iterasi ke 10.

4.8 Analisis pengaruh crosstalk terhadap nilai ISI dan BER.

Crosstalk terjadi karena pengaruh interferensi antara kanal *inphase* dan *quadrature*. Adanya Crosstalk mengakibatkan terjadinya kesalahan deteksi pada penerima, sehingga nilai BER meningkat. Hasil simulasi menunjukkan bahwa equalizer adaptif yang dipasang pada demodulator QPSK mampu mengurangi efek crosstalk sehingga nilai BER turun dari 0,057 menjadi 0,006.

5. Kesimpulan.

Berdasarkan hasil simulasi yang dilakukan pada penelitian ini, maka bisa ditarik kesimpulan sebagai berikut :

1. Orde yang optimum tercapai saat orde yang digunakan 4, dengan perbaikan ISI sebesar 22,936 %. Saat orde filter dinaikkan menjadi 6 dan 8, kenaikan performansi equalizer sudah tidak signifikan lagi (kenaikan perbaikan ISI dibanding orde 4 kurang dari 7 %).
2. Pada saat nilai *forgetting factor* (\square) antara 0,996 sampai 0,98577 proses equalisasi berhenti sebelum iterasi ke 100. Proses equalisasi bisa berlangsung sampai iterasi ke 100, saat nilai \square 0,9875.
3. Dengan kondisi kanal yang berubah setiap 50 simbol, perbaikan yang optimal dilakukan ketika perubahan bobot equalizer dilakukan setiap 10 simbol. Ketika perubahan bobotnya diset setiap 20 dan 50 simbol, nilai ISI output equalizer masih cukup besar (diatas 60 %). Hal ini dikarenakan perubahan bobot tidak bisa mengikuti perubahan kanal, sehingga proses adaptasi selalu terlambat.
4. Equalizer adaptif dengan periode perubahan bobot setiap 10 simbol mampu melakukan perbaikan ISI untuk semua periode perubahan kanal yang disimulasikan (yaitu setiap 10, 25, 50 dan 100 simbol) dengan nilai perbaikan ISI yang bervariasi antara 19,6817 % sampai dengan 30,7215 %.
5. Adanya noise AWGN menyebabkan kinerja equalizer dalam menangani ISI menjadi terganggu. Pada saat nilai SNR 0 dB, equalizer

hanya mampu melakukan perbaikan ISI kurang dari 5 %. Pada saat SNR 5 dB perbaikan ISI sekitar 15 %. Saat SNR 10 dB perbaikannya sekitar 20 %, dan saat SNR 15 dB perbaikannya ISInya sangat besar yaitu sekitar 30 %.

6. Pada saat kondisi noise cukup besar (0 dB dan 5 dB), nilai BER untuk demodulator yang menggunakan equalizer justru lebih besar dibandingkan demodulator yang tidak menggunakan equalizer. Perbaikan BER dengan menggunakan equalizer terjadi saat nilai SNR noise 10 dan 15 dB. Pada SNR 15 dB. Nilai BER untuk demodulator yang menggunakan equalizer adalah 6×10^{-3} . Pada kondisi SNR yang sama jika demodulator tidak menggunakan equalizer nilai BERnya adalah 5×10^{-1} .
7. Untuk kondisi $\mu = 0,99$ 0,995 dan 1 algoritma RLS konvergen pada iterasi ke 10. Saat nilai orde filter transversal diubah-ubah orde 2, 4, dan 8 algoritma ini juga konvergen pada iterasi ke 10.
8. Crosstalk menyebabkan kesalahan deteksi pada penerima, sehingga nilai BER meningkat. Hasil simulasi menunjukkan equalizer adaptif yang dipasang pada demodulator QPSK mampu mengurangi efek crosstalk sehingga nilai BER turun dari 0,057 menjadi 0,006.

Daftar Pustaka:

- [1] Haykin, Simon, *Adaptive Filter Teory*. Prentice-Hall International, Inc. New Jersey. 1996.
- [2] R.E Ziemer and R.L Peterson. *Digital Communication and Spread Spectrum System*.
- [3] Haykin, Simon. *Communication System*, John Wiley & Sons, Inc, 2001.
- [4] Valkama, Mikko. *Statistical Signal Processing Principles with Applications to Communication Receiver Signal Processing*.
- [5] Wijanto, Heroe. *Materi Kuliah : Sistem Komunikasi, Spektral Tersebar*. Laboratorium Sistem Komunikasi Jurusan Teknik Elektro STT Telkom.
- [6] Lutfi, Harunul. *Analisis Sistem Equalisasi Adaptif Pada Demodulator QPSK Dengan Algoritma LMS dan Time Varying LMS*. Tugas Akhir Jurusan Teknik Elektro STT Telkom, 2006.
- [7] Pratikno, Wahyu. *Simulasi Equalisasi Adaptif Dengan Menggunakan Algoritma LMS Pada QPSK*, Tugas Akhir Jurusan Teknik Elektro STT Telkom, 2005.
- [8] Habibah, Umu. *Pembangunan Alat Ajar Mata Kuliah Sistem Komunikasi Bagian ASK, FSK, BPSK, dan QPSK*. Tugas Akhir Jurusan Teknik Elektro STT Telkom, 2005.
- [9] Modul Praktikum Elektronika Komunikasi, Laboratorium Elektronika Komunikasi, STT Telkom, Bandung.

- [10] Modul Praktikum Sistem Komunikasi, Laboratorium Sistem Komunikasi, STT Telkom, Bandung.
- [11] Tato, Luís. M and Miranda, Henrique C. *Simulation of an RLS Adaptive Equalizer using Simulink*. Department of Electrical and Computer Engineering School of Engineering, University of Porto.
- [12] Rappaport, Theodore S. *Wireless communication, principles and practice*. Prentice Hall PTR, New Jersey 07458.
- [13] Advanced Digital Signal Processing, Theory and Application.
- [14] Ramasami, Vijaya Chandrani. Fractionally Spaced, Kalman and Lattice Equalizer.