



TUGAS AKHIR - EE 184801

**ANALISIS APLIKASI *PREDISTORTER* UNTUK MITIGASI
NONLINIER HPA PADA SISTEM GFDM**

Wildan Bramaji Wahyudianto
NRP 0711144000125

Dosen Pembimbing
Dr. Ir. Titiek Suryani, M.T.

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2019



FINAL PROJECT - EE 184801

***ANALYSIS ON PREDISTORTER APPLICATION FOR
MITIGATING NONLINEAR HPA IN GFDM SYSTEM***

Wildan Bramaji Wahyudianto
NRP 0711144000125

Advisors
Dr. Ir. Titiek Suryani, M.T.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2019

**PERNYATAAN KEASLIAN
TUGAS AKHIR**

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir dengan judul "**Analisis Aplikasi *Predistorter* untuk Mitigasi Nonlinier HPA pada Sistem GFDM**" adalah benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya orang lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2019



Wildan Bramaji Wahyudianto
NRP 07111440000125

[Halaman Ini Sengaja Dikosongkan]

**ANALISIS APLIKASI *PREDISTORTER* UNTUK
MITIGASI NONLINIER HPA PADA SISTEM GFDM**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
pada**

**Bidang Studi Telekomunikasi Multimedia
Departemen Teknik Elektro
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I



ANALISIS APLIKASI *PREDISTORTER* UNTUK MITIGASI NONLINIER HPA PADA SISTEM GFDM

Nama : Wildan Bramaji Wahyudianto
Dosen Pembimbing : 1. Dr. Ir. Titiek Suryani, M.T.

ABSTRAK

Sistem GFDM (*Generalized Frequency Division Multiplexing*) adalah sistem komunikasi *multi-carrier* yang dikembangkan untuk memenuhi kebutuhan teknologi seluler generasi kelima (5G). Sistem GFDM dikembangkan untuk menjadi alternatif dari sistem OFDM. Modulasi OQAM digunakan pada sistem GFDM dikarenakan beberapa kelebihan yakni memiliki sifat mendekati orthogonal sehingga sistem GFDM lebih tahan terhadap ISI dan ICI. *High power amplifier* (HPA) merupakan perangkat yang esensial dalam sistem komunikasi. Namun HPA memiliki titik saturasi yang dapat mengakibatkan penguatan tidak linier atau nonlinier, sehingga berujung dengan munculnya distorsi pada sinyal yang dikirimkan. Salah satu cara untuk memitigasi efek nonlinier HPA adalah dengan menggunakan teknik predistorsi. Sehingga dalam penelitian ini akan dibandingkan dua skema antara sistem GFDM-OQAM sebelum dan sesudah penggunaan *predistorter* dengan menganalisis kinerja BER untuk nilai IBO yang berbeda serta meneliti distorsi residu setelah aplikasi *predistorter* pada sistem.

Dari hasil simulasi dapat diketahui bahwa dengan menggunakan *predistorter*, kinerja sistem GFDM-OQAM menjadi lebih baik. Hal ini dapat diketahui dari kinerja BER dimana sebelum aplikasi *predistorter* untuk mencapai nilai BER 10^{-3} dibutuhkan IBO atau redaman sebesar 15 dB. Sedangkan setelah aplikasi *predistorter* untuk mencapai nilai BER yang sama dibutuhkan IBO atau redaman sebesar 3 dB. Kemudian untuk mengetahui distorsi residu dapat diketahui dari nilai EVM mengalami penurunan drastis dimana sebelum aplikasi *predistorter* untuk mencapai nilai EVM 15%, dibutuhkan IBO atau redaman sebesar 15 dB. Sedangkan setelah aplikasi *predistorter* untuk mencapai nilai EVM yang sama dibutuhkan IBO atau redaman sebesar 3 dB. Apabila pada kanal diberikan *noise* berupa AWGN maka diperlukan kompensasi yakni penambahan IBO atau redaman dari 6 dB menjadi 8 dB sehingga dicapai kinerja BER 10^{-5} .

Kata Kunci : GFDM, *Offset QAM*, HPA, *Predistorter*.

[Halaman Ini Sengaja Dikosongkan]

ANALYSIS ON PREDISTORTER APPLICATION FOR MITIGATING NONLINEAR HPA IN GFDM SYSTEM

Name : Wildan Bramaji Wahyudianto
Advisors : 1. Dr. Ir. Titiek Suryani, M.T.

ABSTRACT

Generalized Frequency Division Multiplexing (GFDM) is a multi carrier communication system developed to fulfill the need of Fifth Generation (5G) cellular technology. GFDM is meant to become the alternate system after OFDM. Offset QAM (OQAM) is being used in GFDM with many benefits such as near orthogonality properties which decrease GFDM susceptibility against ISI and ICI. High power amplifier (HPA) is a device essential in communication system. As essential as it is, HPA has a saturation operating point which lead to nonlinear amplifying causing nonlinear distortion on the output signal. One way to mitigate this effect is using predistortion technique. In order to observe the effect of using predistorter, this study compared two schemes between before and after usage of predistorter on GFDM-OQAM system by analyzing the performance of BER with different IBO and the distortion's residue after using predistorter.

From the results of the simulation, it shows that by using predistorter the GFDM-OQAM system's performance would be better. From BER performance perspective, to reach 10^{-3} the system which doesn't apply predistorter need 15 dB of IBO while after applying predistorter the system only need 3 dB of IBO to reach the same value of BER. Also to identify the distortion's residue, EVM value could be use as identifier. For the system to reach 15% of EVM value without using predistorter, the system need 15 dB of IBO while the system after using predistorter only need 3 dB of IBO to reach the same value of EVM. If AWGN noise are to be added into the signal through the ideal channel, then to get a better system's performance the IBO needed to reach 10^{-5} of BER performance increase from 6 dB into 8 dB.

Keywords : GFDM, Offset QAM, HPA, Predistorter

[Halaman Ini Sengaja Dikosongkan]

KATA PENGANTAR

Segala pujian dan rasa syukur kepada Allah Subhanahu wa Ta'ala, yang telah memberikan rahmat dan karunianya sehingga penulis dapat menyelesaikan laporan Tugas Akhir dengan judul “**Analisis Aplikasi Predistorter Untuk Mitigasi Nonlinier HPA Pada Sistem GFDM**”. Tugas akhir ini disusun untuk memenuhi persyaratan menyelesaikan studi Strata-1 di departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember Surabaya. Selama penyusunan laporan ini penulis mendapatkan banyak dorongan, bimbingan, arahan dan bantuan dari berbagai pihak. Oleh karenanya penulis ingin mengucapkan terima kasih kepada:

1. Kedua orang tua tercinta Bapak Kelik Prasetyo dan Ibu Heny Triwidyastuti serta kedua adik Witsqadianto Wicaksono dan Taqara Rayhan Irfandianto yang dengan tulus memberikan do'a, semangat, dan dukungan penuh,
2. Ibu Dr. Ir. Titiek Suryani, MT. yang sedari awal telah memberikan motivasi, bimbingan, dan arahan hingga penulis mampu menyelesaikan permasalahan dan kendala dalam tugas akhir,
3. Ibu Ari yang telah membantu penulis dalam penelitian dan menyelesaikan tugas akhir,
4. Keluarga e54, dan keluarga Telkom yang selalu ada untuk mengingatkan dan menguatkan dalam pengerjaan tugas akhir.

Semoga Tugas Akhir ini dapat memberikan manfaat bagi pembacanya. Penulis menyadari bahwa dalam tugas akhir ini masih banyak kekurangan sehingga diperlukan masukan berupa kritik dan saran untuk membantu penelitian selanjutnya.

Surabaya, Juni 2019

Wildan Bramaji Wahyudianto

[Halaman Ini Sengaja Dikosongkan]

DAFTAR ISI

Halaman

HALAMAN JUDUL	
PERNYATAAN KEASLIAN TUGAS AKHIR	
HALAMAN PENGESAHAN	
ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xiii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi Penelitian	3
1.6 Sistematika Penulisan	4
1.7 Relevansi	4
BAB 2 TINJAUAN PUSTAKA	5
2.1 Sinyal Baseband dan Passband	5
2.1.1 Sinyal <i>Baseband</i> [3]	5
2.1.2 Modulasi <i>Bandpass</i> [3]	6
2.1.3 Representasi Sinyal <i>Bandpass</i> [6]	6
2.2 <i>Quadrature Amplitude Modulation</i>	7
2.3 Offset Quadrature Amplitude Modulation	9
2.4 <i>Generalized Frequency Division Multiplexing</i>	11
2.4.1 Pemancar GFDM	12
2.4.2 Model Kanal GFDM	14
2.4.3 Penerima GFDM	14
2.4.4 Prinsip Kerja GFDM-OQAM	15
2.4.5 <i>Cyclic Prefix</i>	16
2.4.6 Filter <i>Root Raised Cosine</i>	17
2.5 Kanal Propagasi	18
2.5.1 Kanal AWGN	18
2.6 Elemen Nonlinear Pada Sistem Komunikasi Digital	19
2.6.1 High Power Amplifier [12]	19
2.6.2 Amplifier TWT Model Saleh	23

2.7	Linearisasi High Power Amplifier.....	24
2.7.1	Teknik Linearisasi Predistorsi RF.....	24
2.7.2	Invers Model Saleh [17].....	25
2.8	Teknik Estimasi BER.....	27
2.9	<i>Power Spectral Density</i>	27
2.10	<i>Error Vector Magnitude</i>	28
BAB 3 PEMODELAN DAN SIMULASI SISTEM.....		29
3.1	Parameter Simulasi.....	29
3.2	Pemodelan Sistem.....	31
3.2.1	Bagian <i>Transmitter</i>	31
3.2.2	Bagian <i>Receiver</i>	42
3.3	Validasi Program Simulasi.....	45
3.3.1	BER Sistem GFDM.....	45
3.3.2	Karakteristik AM/AM pada <i>High Power Amplifier</i>	45
3.3.3	<i>Error Vector Magnitude</i>	46
BAB 4 ANALISIS KINERJA SISTEM.....		49
4.1	Analisis Hasil Simulasi Sistem GFDM dengan HPA.....	49
4.1.1	<i>Scatter Plot</i> Simbol di Penerima.....	49
4.1.2	<i>Error Vector Magnitude</i> Sinyal di Penerima.....	53
4.2	Analisis Hasil Simulasi Sistem GFDM dengan HPA dan <i>Predistorter</i>	55
4.2.1	<i>Scatter Plot</i> Simbol di Penerima.....	55
4.2.2	<i>Error Vector Magnitude</i> Sinyal di Penerima.....	60
4.3	Analisis Perbandingan Hasil Simulasi Sistem GFDM dengan HPA Sebelum dan Sesudah Aplikasi <i>Predistorter</i>	62
4.3.1	Perbandingan <i>Bit Error Rate</i> Sebelum dan Sesudah Aplikasi <i>Predistorter</i>	62
4.4	Analisis <i>Power Spectral Density</i> Sistem GFDM Sebelum dan Sesudah Aplikasi <i>Predistorter</i>	65
4.5	Analisis Perbandingan Hasil Simulasi Sistem GFDM dengan HPA Sesudah Aplikasi <i>Predistorter</i> Pada Kanal AWGN.....	67
BAB 5 PENUTUP.....		69
5.1	Kesimpulan.....	69
5.2	Saran.....	69
DAFTAR PUSTAKA.....		70
LAMPIRAN A.....		72
LAMPIRAN B.....		74
LAMPIRAN C.....		78
BIOGRAFI PENULIS.....		106

DAFTAR GAMBAR

Gambar 2.1 Jenis-jenis Modulasi M-QAM [4]	9
Gambar 2.2 Skema representasi <i>time-frequency</i> QAM dan OQAM [7] .	9
Gambar 2.3 Blok Diagram Modulator <i>Offset</i> QAM dengan <i>pulse shaping</i> [9]	10
Gambar 2.4 Blok Diagram Demodulator OQAM dengan <i>pulse shaping</i> [9].....	11
Gambar 2.5 Blok Diagram <i>Transceiver</i> GFDM [11]	11
Gambar 2.6 Partisi Waktu dan Frekuensi antara OFDM dan GFDM [11]	12
Gambar 2.7 Model pemancar <i>baseband</i> GFDM [1].....	13
Gambar 2.8 Blok diagram pada modulator GFDM [11]	14
Gambar 2.9 Model penerima <i>baseband matched filter</i> GFDM [1]	15
Gambar 2.10 <i>Cyclic prefix</i> pada paket GFDM.....	16
Gambar 2.11 (a) Rapat Spektral Daya <i>White Noise</i> (b) Fungsi Otokorelasi <i>White Noise</i> [5].....	19
Gambar 2.12 Grafik AM-AM dan AM-PM [6]	21
Gambar 2.13 Model Sistem <i>Bandpass</i> [6]	21
Gambar 2.14 Model Simulasi Kompleks Ekuivalen [6]	22
Gambar 2.15 Model Alternatif Sistem <i>Bandpass</i> [6].....	22
Gambar 2.16 Model Alternatif Simulasi Kompleks Ekuivalen [6]	22
Gambar 2.17 Skema linearisasi distorsi nonlinear [15].....	25
Gambar 2.18 Cara Kerja <i>Predistorter</i> dalam Blok Diagram [16]	25
Gambar 2.19 Blok diagram <i>Predistorter</i> dan HPA	25
Gambar 2.20 Blok Diagram <i>Predistorter</i> [17]	26
Gambar 2.21 Representasi <i>error vector magnitude</i> [19].....	28
Gambar 3.1 Diagram Alir Sistem GFDM-OQAM dengan HPA	30
Gambar 3.2 Diagram Blok <i>Transceiver</i> Skema Pertama	31
Gambar 3.3 Diagram Blok <i>Transceiver</i> Skema Kedua	31
Gambar 3.4 <i>Bit</i> informasi awal	32
Gambar 3.5 Sinyal <i>Inphase</i> dan <i>Quadrature</i> hasil modulasi 16-QAM. 33	
Gambar 3.6 Konstelasi Sinyal 16-QAM	33
Gambar 3.7 Konversi <i>Serial to Parallel</i>	34
Gambar 3.8 Filter <i>root raised cosine</i>	35
Gambar 3.9 <i>Pulse shaping</i> RRC dengan $\alpha = 0.5$	36
Gambar 3.10 <i>Pulse shaping</i> pada <i>offset</i> QAM	37
Gambar 3.11 Superposisi sebelum proses IFFT	37
Gambar 3.12 Bentuk sinyal <i>subcarrier</i> pada <i>Offset</i> QAM.....	38

Gambar 3.13 Format paket GFDM.....	39
Gambar 3.14 Hasil sinyal setelah ditambahkan CP	39
Gambar 3.15 Perbandingan <i>Output</i> Sinyal	41
Gambar 3.16 <i>Scatter plot</i> simbol diterima pada IBO 5 dB	43
Gambar 3.17 <i>Bit</i> informasi akhir	44
Gambar 3.18 Perbandingan nilai BER pada $E_b/N_0 = 13$ dB	45
Gambar 3.19 Kurva AM/AM pada Sistem GFDM-OQAM	46
Gambar 3.20 Kurva EVM terhadap IBO untuk Sistem GFDM-OQAM dengan HPA saja	47
Gambar 3.21 Kurva EVM terhadap IBO untuk Sistem GFDM-OQAM dengan HPA dan <i>Predistorter</i>	47
Gambar 4.1 <i>Scatterplot</i> Simbol 16-OQAM GFDM dengan HPA untuk IBO: (a) 0 dB, (b) 2 dB, (c) 4 dB, (d) 6 dB, (e) 8 dB, (f) 10 dB, (g) 12 dB, dan (h) 15 dB	50
Gambar 4.2 <i>Scatterplot</i> Simbol 8-PSK GFDM dengan HPA IBO: (a) 0 dB, (b) 2 dB, (c) 4 dB, (d) 6 dB, (e) 8 dB, (f) 10 dB, (g) 12 dB, dan (h) 15 dB	52
Gambar 4.3 Grafik Perbandingan EVM terhadap IBO pada Sistem GFDM-OQAM dengan HPA.....	54
Gambar 4.4 Grafik Perbandingan EVM terhadap IBO pada Sistem GFDM-PSK dengan HPA.....	54
Gambar 4.5 <i>Scatterplot</i> Simbol 16-OQAM GFDM dengan HPA dan <i>Predistorter</i> HPA untuk IBO: (a) 0 dB, (b) 2 dB, (c) 4 dB, (d) 6 dB, (e) 8 dB, (f) 10 dB, (g) 12 dB, dan (h) 15 dB	57
Gambar 4.6 <i>Scatterplot</i> Simbol 8-PSK GFDM dengan HPA dan <i>Predistorter</i> untuk IBO (a) 0 dB (b) 4 dB (c) 8 dB (d) 12 dB	59
Gambar 4.7 Grafik Perbandingan EVM terhadap IBO pada sistem GFDM-OQAM dengan HPA dan <i>Predistorter</i>	60
Gambar 4.8 Grafik Perbandingan EVM terhadap IBO pada sistem GFDM-PSK dengan HPA dan <i>Predistorter</i>	61
Gambar 4.9 Grafik perbandingan BER sistem GFDM-OQAM dengan HPA terhadap IBO.....	62
Gambar 4.10 Grafik perbandingan BER sistem GFDM-OQAM dengan HPA dan <i>Predistorter</i> terhadap IBO.....	64
Gambar 4.11 Grafik <i>Power Spectral Density</i> pada sistem GFDM-OQAM dengan HPA dan <i>Predistorter</i> +HPA	65
Gambar 4.12 Grafik <i>Power Spectral Density</i> pada sistem GFDM-OQAM dengan HPA dan HPA+ <i>predistorter (side lobe)</i>	66

Gambar 4.13 Grafik <i>Power Spectral Density</i> pada sistem GFDM-OQAM dengan HPA dan HPA+ <i>predistorter (main lobe)</i>	66
Gambar 4.14 Grafik perbandingan BER Sistem GFDM_OQAM dengan HPA dan <i>predistorter</i> pada kanal AWGN	67

[Halaman Ini Sengaja Dikosongkan]

DAFTAR TABEL

Tabel 2.1 Parameter optimum untuk persamaan HPA model Saleh [14]	24
Tabel 4.1 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-OQAM dengan HPA saja	51
Tabel 4.2 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-PSK dengan HPA saja.....	53
Tabel 4.3 Nilai EVM terhadap IBO pada sistem GFDM-OQAM dan GFDM-PSK dengan HPA untuk $\alpha = 0.5$	55
Tabel 4.4 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-OQAM dengan HPA dan <i>predistorter</i>	57
Tabel 4.5 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-OQAM dengan HPA dan <i>predistorter</i>	59
Tabel 4.6 Nilai EVM terhadap IBO pada sistem GFDM-OQAM dan GFDM-PSK dengan HPA dan <i>Predistorter</i> untuk $\alpha = 0.5$	61
Tabel 4.7 Nilai BER terhadap IBO pada sistem GFDM-OQAM dengan HPA untuk $\alpha = 0.5$	63
Tabel 4.8 Nilai BER terhadap IBO pada sistem GFDM-OQAM dengan <i>Predistorter</i> dan HPA untuk $\alpha = 0.5$	64
Tabel 4.9 Nilai BER terhadap IBO pada sistem GFDM-OQAM dengan HPA untuk $\alpha = 0.5$	68

[Halaman Ini Sengaja Dikosongkan]

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Sistem komunikasi nirkabel saat ini merupakan komponen penting yang tidak dapat lepas dari kehidupan sehari-hari. Teknologi 4G yang telah banyak digunakan saat ini memiliki laju data yang tinggi jauh melampaui teknologi sebelumnya. Namun tantangan yang muncul untuk kebutuhan sistem komunikasi nirkabel selanjutnya (5G), tidak hanya laju data yang meningkat drastis, namun juga perlu dipertimbangkan aspek-aspek penting lainnya seperti konsumsi daya yang rendah serta latensi yang rendah pula. GFDM (Generalized Frequency Division Multiplexing) adalah skema sistem komunikasi *multi-carrier* yang dikembangkan untuk memenuhi kebutuhan teknologi seluler generasi kelima (5G). Sistem GFDM didasarkan pada modulasi independen tiap blok yang terdiri atas sejumlah *subsymbol* dan sejumlah *subcarrier*. Pulse shaping yang diterapkan pada masing-masing subcarrier menggunakan filter prototipe yang bergeser secara sirkulas pada domain waktu dan frekuensi. Filter prototipe yang digunakan merupakan filter non orthogonal. Kemudian pada GFDM digunakan modulasi *offset QAM* yang dikembangkan dari modulasi QAM, dimana komponen *inphase* dan *quadrature* tidak mengalami transisi pada waktu yang sama. Oleh karenanya penggunaan *offset QAM* dapat memberikan efek orthogonal pada sinyal GFDM.

High power amplifier (HPA) adalah salah satu perangkat atau komponen esensial dalam suatu sistem komunikasi nirkabel yang berfungsi untuk menguatkan sinyal yang akan dikirimkan oleh *transmitter* sehingga bisa diterima oleh *receiver*. *Travelling Wave Tube Amplifier* (TWTA) adalah salah satu jenis amplifier yang terbuat dari tabung hampa udara secara khusus dibuat untuk menguatkan sinyal RF pada rentang gelombang mikro. TWTA termasuk ke dalam *linear beam tubes*. Umumnya *power amplifier* khususnya HPA dapat menimbulkan distorsi nonlinier pada sinyal yang dikirimkan apabila HPA bekerja pada daerah saturasi. Meskipun sistem GFDM memiliki PAPR yang kecil namun masih dapat terkena distorsi nonlinier, khususnya distorsi fasa yang dapat menggeser letak titik konstelasi *offset QAM* sinyal yang diterima sehingga berakibat buruknya informasi yang diterima. Untuk mengurangi distorsi nonlinear dari HPA, dapat dilakukan berbagai

metode. Salah satu dari metode tersebut adalah teknik predistorsi. Teknik predistorsi juga terbagi menjadi sejumlah teknik yang memiliki keterbatasan masing-masing dalam menangani distorsi nonlinier.

Sehingga pada Tugas Akhir ini akan dianalisis aplikasi dari *predistorter* untuk mengatasi distorsi nonlinier yang diakibatkan oleh HPA pada sistem GFDM, khususnya dengan menggunakan modulasi *offset QAM*. Penelitian ini meliputi analisis kinerja BER untuk nilai IBO yang berbeda serta meneliti distorsi residu setelah aplikasi predistorter pada sistem. Tujuan akhir dari penelitian ini adalah untuk meningkatkan ketahanan sistem komunikasi GFDM khususnya terhadap distorsi nonlinier.

1.2 Perumusan Masalah

Permasalahan yang dibahas dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana cara memitigasi efek nonlinier pada sistem GFDM-OQAM karena *high power amplifier* ?
2. Apakah penggunaan predistorter pada sistem GFDM-OQAM dapat meningkatkan kinerja sistem GFDM-OQAM serta menurunkan efek nonlinier beserta distorsi residu yang dapat muncul ?

1.3 Tujuan

Tujuan yang diharapkan tercapai setelah selesainya tugas akhir ini adalah sebagai berikut

1. Mengatasi pengaruh efek nonlinear pada sistem GFDM dengan modulasi OQAM karena *high power amplifier*.
2. Membandingkan kinerja sistem GFDM-OQAM dengan dan tanpa predistorter serta menganalisis kemampuan predistorter dalam mengatasi distorsi residu yang muncul.

1.4 Batasan Masalah

Untuk menyelesaikan permasalahan yang ada, maka hal-hal yang akan dilakukan dalam penelitian ini adalah :

1. Pada sistem GFDM ini akan menggunakan modulasi OQAM dengan filter *pulse shaping raised-cosine*.
2. *High power amplifier* yang digunakan adalah *travelling wave tube amplifier* yang didasarkan pada Model Saleh.
3. *Predistorter* yang digunakan berdasarkan invers dari fungsi HPA Model Saleh.
4. Kanal yang akan digunakan adalah kanal ideal (AWGN).

5. Paramater yang dianalisa adalah *bit error rate* (BER) dan *error vector magnitude* (EVM).

1.5 Metodologi Penelitian

Metode yang digunakan pada penelitian tugas akhir ini dijabarkan sebagai berikut :

1. Studi Literatur

Mencari dan mempelajari beberapa buku, paper, jurnal maupun literatur ilmiah lainnya mengenai teori maupun mengenai hal-hal teknis lain yang dibutuhkan untuk menyelesaikan tugas akhir. Selain itu juga melakukan *internet* research. Pada tahap ini akan dipelajari sistem GFDM, modulasi Offset M-QAM, *high power amplifier* berdasarkan model Saleh, *predistorter*, dan distorsi residu.

2. Perumusan Masalah dan Batasannya

Menuliskan rumusan masalah berdasarkan permasalahan yang akan diteliti serta memberikan batasan-batasan pada rumusan masalah. Rumusan masalah adalah inti dari suatu permasalahan yang ingin diselesaikan. Sedangkan batasan masalah berfungsi untuk membatasi ruang lingkup tugas akhir.

3. Perancangan sistem GFDM-OQAM menggunakan MATLAB

Pada tahapan ini program GFDM akan dibuat berdasarkan paper terkait serta penambahan komponen HPA dan predistorter pada sistem berdasarkan paper dan buku referensi.

4. Simulasi program GFDM-OQAM

Pada tahap ini dilakukan pengujian sistem GFDM-OQAM dengan memasukkan nilai-nilai parameter yang berbeda-beda sehingga akan didapatkan respon yang berbeda-beda. Parameter-parameter yang dimasukkan antara lain nilai IBO yang diberlakukan pada sinyal input HPA, dan lainnya.

5. Pengolahan hasil simulasi dan analisis data

Tahap ini baru bisa dilakukan ketika pengujian program GFDM-OQAM telah dilakukan serta mendapatkan data-data yang telah valid.

6. Penarikan Kesimpulan

Pada tahap ini dilakukan pengamatan terhadap hasil data yang dihasilkan pada tahap perancangan untuk dianalisis lebih lanjut lagi. Analisis terhadap data meliputi pengaruh aplikasi predistorter terhadap linearitas sinyal serta efek terhadap distorsi residu yang muncul.

7. Penulisan Buku Tugas Akhir

Pada tahap ini dilakukan penulisan buku laporan tugas akhir berdasarkan tahapan sebelumnya.

1.6 Sistematika Penulisan

Pembahasan dalam tugas akhir ini akan dibagi dalam lima bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

Pada bab ini dijelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, metodologi, sistematika pembahasan dan relevansi.

Bab II Tinjauan Pustaka

Pada bab ini dijelaskan mengenai tinjauan pustaka yang mendukung pengembangan sistem GFDM-OQAM dengan tambahan HPA dan predistorter.

Bab III Pemodelan dan Simulasi Sistem

Pada bab ini dijelaskan mengenai pembuatan model sistem serta simulasi sistem GFDM-OQAM dengan tambahan HPA dan predistorter.

Bab IV Analisis Kinerja Sistem

Pada bab ini berisi pengolahan data hasil dari simulasi sistem GFDM-OQAM dengan HPA tanpa predistorter dan dengan predistorter.

Bab V Kesimpulan dan Saran

Pada bab ini berisi kesimpulan dan saran hasil analisis.

1.7 Relevansi

Hasil dari yang didapat pada tugas akhir ini diharapkan dapat memberi manfaat dan solusi sekaligus sebagai referensi untuk implementasi predistorter pada sistem GFDM-OQAM menggunakan *high power amplifier* beserta pengaruh penggunaan predistorter terhadap distorsi nonlinier yang muncul pada sinyal.

BAB 2 TINJAUAN PUSTAKA

2.1 Sinyal Baseband dan Passband

Sinyal pada komunikasi digital dapat dikategorikan menjadi dua, yaitu sinyal *baseband* dan sinyal *passband*. Sinyal *baseband* adalah sinyal yang dimiliki oleh informasi dengan frekuensi mula-mula yang akan dikirimkan sebelum dimodulasi dengan sinyal pembawa. Sinyal ini memiliki frekuensi pita *narrowband* yang terletak disekitar frekuensi nol. Sedangkan sinyal *passband* adalah sinyal dengan frekuensi jauh diatas frekuensi sinyal *baseband*.

2.1.1 Sinyal Baseband [3]

Pada setiap sistem komunikasi akan terdapat sumber informasi baik berupa sumber analog maupun sumber digital. Apabila informasi berupa tekstual maka akan diubah menjadi deret biner digital menggunakan encoder sebelum dilewatkan *pulse modulator*. Jika informasi berupa analog seperti audio atau video, maka akan diubah menjadi sumber digital ekivalen dengan cara dilewatkan melalui tiga tahapan terlebih dahulu yakni *sampling*, *quantizing*, dan *encoding* dengan hasil akhir berupa deret biner (sumber digital). Deret biner ini kemudian akan di transmisikan melalui kanal *baseband*, seperti kabel *coaxial* atau gelombang radio. Namun sebelum dilewatkan kanal tersebut, deret biner perlu diubah menjadi *waveform* yang kompatibel dengan kanal.

Untuk melakukan proses pengambilan sampel sinyal informasi, *waveform* yang cocok adalah sinyal pulsa. Sinyal pulsa tersebut akan digunakan sebagai sinyal *sampling*. Salah satu dari sinyal tersebut adalah fungsi impuls:

$$x_{\delta} = \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \quad 2.1$$

Hasil dari proses pengambilan sampel dinamakan *pulse amplitude modulation* (PAM). Untuk mendapatkan kembali sinyal analog dari PAM, sampel PAM dapat dilewatkan filter *low-pass*. Untuk mendapatkan sinyal informasi kembali dengan baik terdapat parameter penting yang perlu diperhatikan yaitu Nyquist *criterion* dimana:

$$f_s \geq 2f_m \quad 2.2$$

Dengan f_s adalah laju *sampling* dan f_m adalah frekuensi sinyal informasi. Jika laju *sampling* $f_s = 2f_m$, maka laju *sampling* ini dinamakan laju Nyquist (*Nyquist rate*). Nyquist *criterion* merupakan kondisi yang bila

terpenuhi, sebuah sinyal dapat di rekonstruksi kembali dari sinyal terkirim yang disampling secara *uniform*.

2.1.2 Modulasi *Bandpass* [3]

Modulasi digital adalah proses dimana simbol digital diubah menjadi bentuk gelombang yang sesuai dengan karakteristik dari kanal tertentu. Jika bentuk gelombang dari modulasi *baseband* umumnya adalah pulsa-pulsa, maka pada modulasi *bandpass*, pulsa-pulsa tadi memodulasi sinyal sinus yang dinamakan gelombang pembawa (*carrier wave*). Frekuensi dari *carrier wave* umumnya jauh diatas frekuensi sinyal *baseband*. Hal ini menjelaskan mengapa ukuran antena untuk komunikasi berukuran kecil dan tidak sebaliknya. *Carrier wave* atau modulasi *bandpass* juga dapat memisahkan beberapa sinyal yang menggunakan satu kanal yang sama sehingga tidak terjadi tumpang tindih antar sinyal tersebut.

Modulasi *bandpass* adalah proses yang mengubah-ubah amplitudo, frekuensi, atau fasa dari sinyal karier atau kombinasi dari ketiga parameter tersebut berdasarkan sinyal informasi yang akan dikirimkan. Bentuk umum dari sinyal karier adalah:

$$s(t) = A(t) \cos(\theta(t)) \quad 2.3$$

dengan $\theta(t)$ dapat ditulis menjadi

$$\theta(t) = \omega_0 t + \phi(t) \quad 2.4$$

Sehingga persamaan diatas dapat ditulis menjadi

$$s(t) = A(t) \cos[\omega_0 t + \phi(t)] \quad 2.5$$

Ketika sinyal yang dikirimkan diterima maka diperlukan suatu metode untuk mendeteksi sinyal yang diterima sehingga informasi didalamnya dapat diteruskan. Proses deteksi sinyal *bandpass* dapat dibagi menjadi dua, deteksi koheren dan deteksi non-koheren. Deteksi pada penerima dikategorikan koheren apabila penerima mengetahui fasa dari karier dan menggunakannya untuk mendeteksi sinyal yang masuk. Sedangkan deteksi dikatakan non-koheren apabila penerima tidak menggunakan informasi tadi (fasa sinyal karier) untuk mendeteksi sinyal yang masuk.

2.1.3 Representasi Sinyal *Bandpass* [6]

Dalam sebuah sistem komunikasi, modulasi yang dilakukan pada sebuah transmitter untuk menghasilkan sinyal *bandpass* dan demodulasi yang dilakukan pada receiver untuk memulihkan informasi sinyal digital melibatkan pergeseran frekuensi kerja. Untuk mengurangi

kompleksitas dari segi matematis dan simulasi maka digunakan sebuah metode *complex envelope* (selubung kompleks) yang menggunakan konsep *lowpass equivalent* sinyal dan sistem. Hal ini dikarenakan sifat-sifat umum dari sinyal bandpass berasal dari sinyal lowpass (baseband).

Sebuah sinyal yang termodulasi amplitudo dan fasa dapat dituliskan sebagai berikut:

$$x(t) = p(t) \cos(2\pi f_0 t) - q(t) \sin(2\pi f_0 t) \quad 2.6$$

$$X(f) = \frac{1}{2}[P(f - f_0) + jQ(f - f_0) + P(f + f_0) - jQ(f + f_0)] \quad 2.7$$

Dimana persamaan 2.7 merupakan spektrum dari persamaan 2.6. $P(f)$ dan $Q(f)$ adalah spektra dari masing-masing $p(t)$ dan $q(t)$. Jika B adalah bandwidth dari masing-masing $p(t)$ dan $q(t)$ serta $B \leq f_0$,

$$P(f) = 0, \quad |f| > f_0 \quad 2.8$$

$$Q(f) = 0, \quad |f| > f_0 \quad 2.9$$

Maka transformasi Hilbert dari $x(t)$ adalah

$$\hat{x}(t) = p(t) \sin(2\pi f_0 t) + q(t) \cos(2\pi f_0 t) \quad 2.10$$

Kemudian sinyal analitis atau *pre-envelope* dapat dituliskan sebagai berikut:

$$z_x(t) = x(t) + j\hat{x}(t) \quad 2.11$$

Dengan menggunakan rumus Euler untuk menggantikan $\cos(2\pi f_0 t)$ dan $\sin(2\pi f_0 t)$ masing-masing dengan $\frac{1}{2} [e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}]$ dan $\frac{1}{2} [e^{j2\pi f_0 t} - e^{-j2\pi f_0 t}]$ maka persamaan 2.11 dapat dituliskan menjadi

$$z_x(t) = [p(t) + jq(t)]e^{j2\pi f_0 t} \quad 2.12$$

$$Z_x(f) = 2X(f)u(f) = P(f - f_0) + jQ(f - f_0) \quad 2.13$$

Persamaan 2.12 adalah *complex envelope* atau *lowpass equivalent* yang dibutuhkan untuk mengurangi kompleksitas dan waktu simulasi.

2.2 Quadrature Amplitude Modulation

Quadrature amplitude modulation (QAM) adalah salah satu teknik modulasi pada komunikasi digital yang memetakan deretan bit biner menjadi beberapa sinyal basis yang terkait. Sinyal-sinyal basis ini dapat dibedakan dari amplitudonya, fasanya, frekuensinya, atau kombinasi dari dua atau lebih parameter parameter sebuah sinyal. QAM dikembangkan dari PAM [4] dimana dua simbol berbeda yang diambil dari deret informasi dilanjutkan menuju dua karier *quadrature* yaitu $\cos 2\pi f_c t$ dan $\sin 2\pi f_c t$. Pada modulasi M-ary PSK, sinyal yang di

transmisikan memiliki amplitudo yang konstan sehingga konstelasi yang terbentuk berbentuk lingkaran. Jika amplitudo pada sinyal juga di variasikan seperti halnya fasa, maka akan didapatkan modulasi *quadrature amplitude modulation* (QAM). Gambar 2.1 menunjukkan variasi konstelasi dari modulasi QAM.

Bentuk umum dari sinyal *M*-ary QAM dapat dituliskan sebagai berikut [5]:

$$S_i(t) = \sqrt{\frac{2E_{min}}{T_s}} a_i \cos(2\pi f_c t) + \sqrt{\frac{2E_{min}}{T_s}} b_i \sin(2\pi f_c t)$$

$$0 \leq t \leq T \quad i = 1, 2, \dots, M \quad 2.14$$

dimana E_{min} adalah energi minimum dari sinyal pada amplitudo terendah dan a_i , b_i adalah bilangan integer yang menunjukkan lokasi dari titik sinyal pada konstelasi [5].

Dengan mengembangkan persamaan 2.14 menjadi :

$$S_i(t) = a_i \sqrt{E_{min}} \phi_1(t) + b_i \sqrt{E_{min}} \phi_2(t) \quad 2.15$$

maka koordinat dari pesan ke- i ditunjukkan oleh $a_i \sqrt{E_{min}}$ dan $b_i \sqrt{E_{min}}$. a_i dan b_i merupakan elemen dari matriks persegi $L \times L$ dimana $L = \sqrt{M}$. Himpunan dari (a_i, b_i) dapat dituliskan sebagai berikut [6]:

$$(a_i, b_i) \in (\pm 1, \pm 3, \pm 7, \dots, \pm \sqrt{M} - 1) \quad 2.16$$

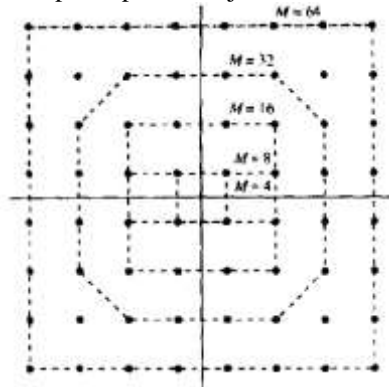
Sehingga jika dituliskan ke dalam matriks $L \times L$ dengan mengambil contoh 16-QAM akan seperti berikut [5]:

$$\{a_i, b_i\} = \begin{bmatrix} (-3,3) & (-1,3) & (1,3) & (3,3) \\ (-3,1) & (-1,1) & (1,1) & (3,1) \\ (-3,-1) & (-1,-1) & (1,-1) & (3,-1) \\ (-3,-3) & (-1,-3) & (1,-3) & (3,-3) \end{bmatrix} \quad 2.17$$

dimana untuk setiap nilai a_i terletak pada sumbu *Inphase* (*I channel*) dan untuk setiap nilai b_i terletak pada sumbu *Quadrature* (*Q channel*).

Terlihat pada modulasi 16-QAM terdapat masing-masing empat nilai untuk komponen *Inphase* dan *Quadrature* sehingga muncul 16 kemungkinan kombinasi dari kedua komponen. Masing-masing simbol pada konstelasi tersusun atas empat bit dimana dua bit diambil dari

komponen I dan dua bit lainnya diambil dari komponen Q. Laju dari setiap simbol adalah seperempat dari laju bit informasi.

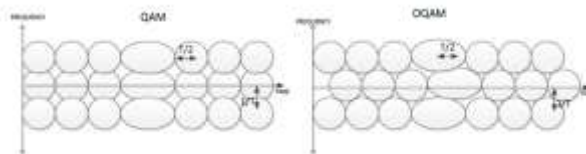


Gambar 2.1 Jenis-jenis Modulasi M-QAM [4]

2.3 Offset Quadrature Amplitude Modulation

Modulasi *Offset* QAM adalah pengembangan dari modulasi QAM yang memiliki sedikit perbedaan namun sangatlah penting. Perbedaan tersebut antara lain digunakannya *guard interval* pada modulasi QAM dan *pulse shaping* pada modulasi OQAM. Kemudian lompatan fasa yang mungkin terjadi pada modulasi QAM adalah $\pm 90^\circ$ dan 180° . Sedangkan pada modulasi *offset* QAM, lompatan fasa yang mungkin terjadi hanya $\pm 90^\circ$. Pada modulasi ini, *offset* hanya diberlakukan pada komponen *quadrature* sinyal. Akibatnya komponen *inphase* dan *quadrature* sinyal tidak akan mengalami transisi pada waktu yang sama [7].

Hal ini akan mengurangi *Inter Carrier Interference* (ICI) karena terjadi pengurangan jarak kanal yang berdekatan pada *subcarrier*. Dengan memberikan *delay* setengah simbol, maka *crossstalk* yang mungkin terjadi akan digeser ke sampel genap sehingga simbol kompleks yang dikirimkan dapat diterima tanpa *crossstalk* dari sampel ganjil [7].



Gambar 2.2 Skema representasi *time-frequency* QAM dan OQAM [7]

Offset QAM dapat mencapai kondisi yang diperlukan untuk mendekati operasi bebas *crosstalk*. Pada *matched filter*, saluran pusat memperoleh nilai maksimal pada titik sampling. Angka ini jelas menunjukkan bahwa ada sisa *crosstalk* dari saluran tambahan imajiner $j\pm 1$ tanpa *offset*. Di sisi lain, *crosstalk* dapat dihilangkan jika saluran tambahan imajiner tertunda sebesar $T/2$ [8]

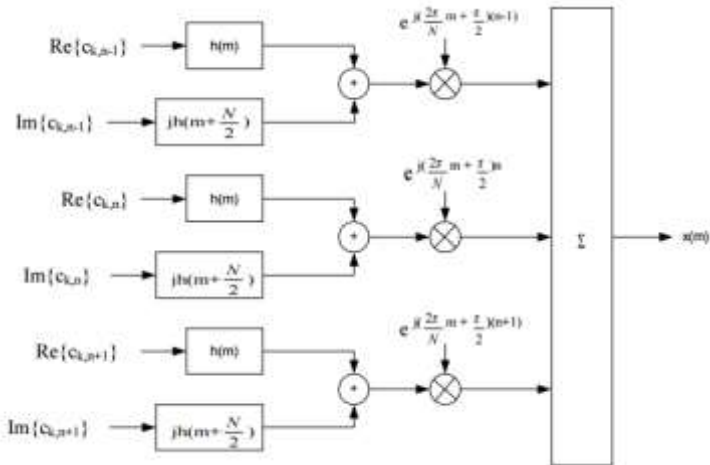
Blok diagram dari modulator OQAM ditunjukkan pada Gambar 2.3 yang dilengkapi dengan *pulse shaping*. Pada setiap kanal, bagian riil dan imajiner disimbolkan dengan $c_{k,n}$, yang kemudian di-filter dengan *pulse shaping* $h(m)$ dan $h\left(m + \frac{N}{2}\right)$. Selanjutnya kedua bagian tersebut dijumlahkan dan digeser pada frekuensi yang telah ditentukan dengan menggunakan modulasi *baseband*.

Sinyal bagian riil dapat dituliskan sebagai berikut [9]:

$$a_{k,n} = \text{Re} \left\{ \sum_m h(m) \times (kN - m) e^{j\left(\frac{2\pi}{N}m - \frac{\pi}{2}\right)n} \right\} \quad 2.18$$

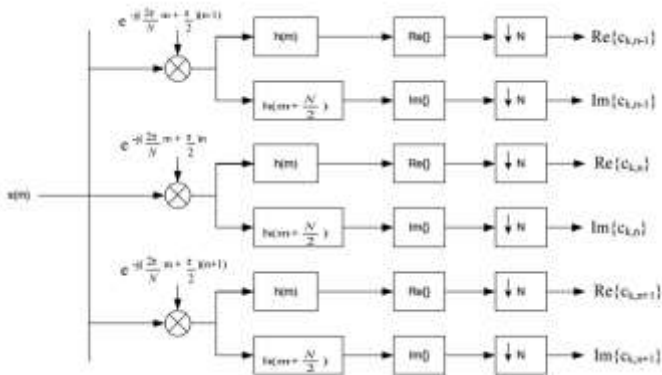
Sedangkan sinyal bagian imajiner dituliskan sebagai berikut [9]:

$$b_{k,n} = \text{Im} \left\{ \sum_m h\left(m - \frac{N}{2}\right) \times (kN - m) e^{j\left(\frac{2\pi}{N}m - \frac{\pi}{2}\right)n} \right\} \quad 2.19$$



Gambar 2.3 Blok Diagram Modulator *Offset QAM* dengan *pulse shaping* [9]

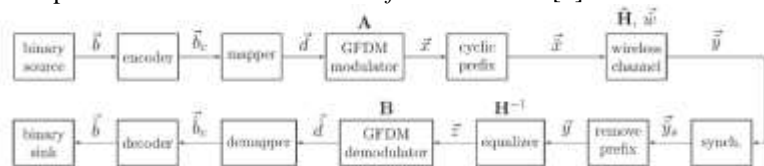
Selanjutnya blok diagram dari demodulator OQAM ditunjukkan pada Gambar 2.4 yang dimana prosesnya merupakan kebalikan dari bagian modulator OQAM sebelumnya.



Gambar 2.4 Blok Diagram Demodulator OQAM dengan *pulse shaping* [9]

2.4 Generalized Frequency Division Multiplexing

Generalized Frequency Division Multiplexing atau GFDM adalah sistem *multicarrier* yang mengimplementasikan pendekatan klasik pita filter [10]. Sistem ini mengadopsi prinsip OFDM dimana sinyal GFDM dikonstruksi dari penjumlahan *tones*, yaitu beberapa sinyal yang membawa simbol data tertentu pada *timeslot* dan *subcarrier* tertentu. GFDM didasarkan pada modulasi blok-blok independen dengan tiap bloknya tersusun atas sejumlah *subcarrier* dan *subsymbols* serta penggunaan *pulse shaping* yang fleksibel. *Subcarriers* di-filter dengan filter prototipe yang bergeser secara sirkular pada domain waktu dan domain frekuensi. Metode ini dapat mengurangi emisi *out of band* (OOB), meningkatkan efisiensi spektrum, dan mengurangi *interchannel interference* (ICI) dan *intersymbol interference* (ISI). GFDM juga merupakan salah satu kandidat *waveform* untuk 5G [1].

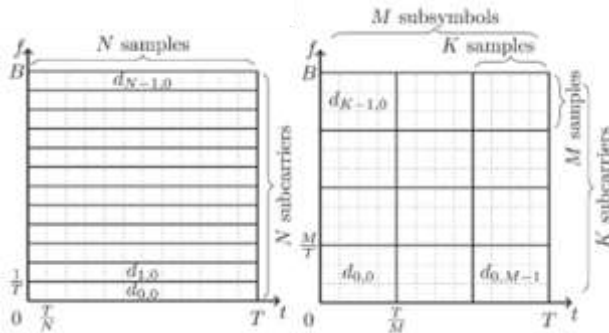


Gambar 2.5 Blok Diagram *Transceiver* GFDM [11]

Pada blok diagram Gambar 2.5, deretan data biner \vec{b} (konversi analog ke sinyal digital) yang dihasilkan oleh sumber data dikodekan menjadi \vec{b}_c . Data yang telah dikodekan menjadi sinyal biner (*bit*) akan

dipetakan menjadi deretan simbol di dalam blok *mapper*. *Mapper* yang digunakan di dalam penelitian Tugas Akhir ini adalah *Quadrature Amplitude Modulation* (QAM). Modulasi M-QAM merupakan teknik pengkodean M-ary QAM yang memiliki kemungkinan kombinasi simbol yang berbeda sejumlah M simbol dengan masing-masing *state* terdiri atas $n = \sqrt[3]{M}$ bit data. Hasil dari blok *mapper* adalah vektor \vec{d} , yang secara matematis dapat juga ditulis sebagai $\vec{d} = d_0, d_1, d_2, \dots, d_{N-1}$ dimana N adalah jumlah seluruh simbol. Vektor kemudian diubah menjadi data dengan kelajuan rendah dan didekomposisi menjadi blok GFDM dengan ukuran $K \times M$ di dalam blok GFDM modulator dimana K dan M secara berturut turut menotasikan jumlah sampel *subcarrier* dan *subsymbol* untuk setiap blok GFDM. Vektor hasil dekomposisi dapat dituliskan $\vec{d} = (d_{0,0}, d_{1,0}, \dots, d_{K-1,M-1})$. Sedangkan penjelasan detail dari blok GFDM ditunjukkan pada Gambar 2.6.

Gambar 2.6 menjelaskan bahwa sinyal GFDM merupakan penjumlahan dari beberapa blok dengan ukuran $K \times M$ yang terpisah dengan interval M/T yang berisi beberapa *subsymbol* (M) dan *subcarrier* (K) yang berbeda. Berdasarkan hal tersebut maka GFDM dapat membawa deretan simbol yang lebih banyak dibandingkan dengan OFDM dalam sekali transmisi data. Apabila nilai $N = KM$, ketika nilai $M = 1$ maka sinyal GFDM akan menjadi sinyal OFDM.

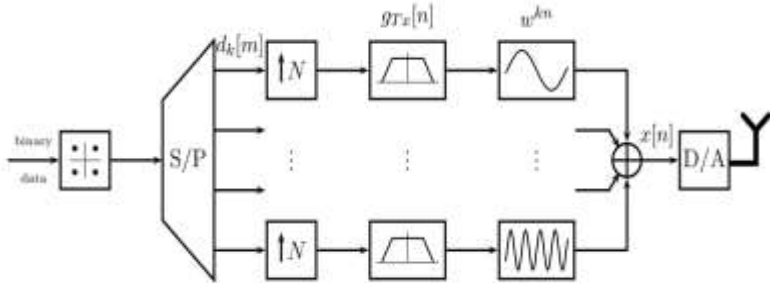


Gambar 2.6 Partisi Waktu dan Frekuensi antara OFDM dan GFDM [11]

2.4.1 Pemancar GFDM

Skema pada Gambar 2.7 memodelkan pemancar GFDM pada *baseband* yang mendistribusikan *data symbol* bernilai kompleks $d_k[m]$ pada K *subcarrier* dan M *symbol*. Setiap *subcarrier* berbentuk pulsa

dengan sebuah filter pengirim $g_{Tx}[n]$ dan dimodulasi dengan sebuah *subcarrier* frekuensi pusat $e^{-j2\pi\frac{kn}{N}}$.



Gambar 2.7 Model pemancar *baseband* GFDN [1]

Untuk memenuhi kriteria *Nyquist*, setiap *symbol* disampling sebanyak N kali, yang mengarah ke MN sampel per *subcarrier*. Sinyal yang ditransmisikan dituliskan seperti persamaan 2.20

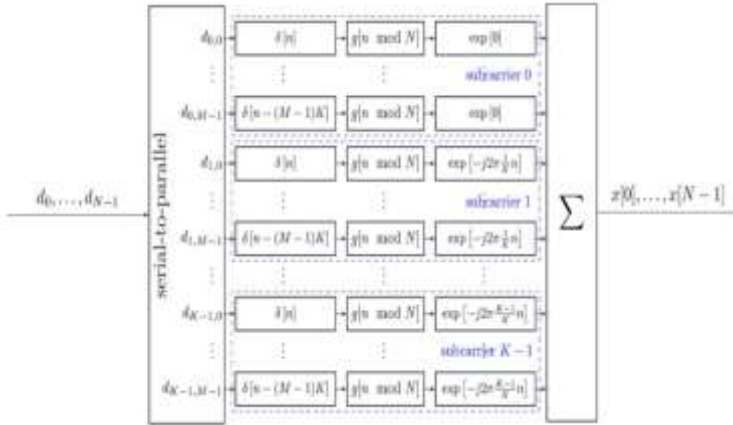
$$x[n] = \sum_{m=0}^{M-1} \sum_{k=0}^{K-1} d_k[m] \otimes g_{Tx}[n - mN] e^{-j2\pi\frac{kn}{N}} \quad 2.20$$

dimana operasi \otimes menunjukkan operasi *circular convolution*. Sinyal tersebut diputar secara sirkular untuk mencegah ekspansi sinyal di luar sinyal GFDN yang dikarenakan proses *filtering* dengan *pulse shaping filter* $g_{Tx}[n]$, dan kemudian digeser menggunakan K/N frekuensi dengan $1/N$ sebagai interval *subcarrier*. Sehingga setiap sinyal yang ditransmisikan merupakan hasil penjumlahan dari semua sinyal-sinyal *subcarrier* seperti pada persamaan 2.20. Filter $g_{Tx}[n]$ memiliki periode $n \bmod MN$. Sedangkan data simbol $d_k[m]$ telah melalui operasi *upsampling*.

Untuk proses *pulse shaping*, *subcarrier up-conversion* dan superposisi dapat direpresentasikan oleh operasi matriks dengan model persamaan sebagai berikut [1]:

$$\mathbf{x} = \mathbf{A}\mathbf{d} \quad 2.21$$

dengan \mathbf{x} adalah sampel waktu transmit $x[n]$ dan \mathbf{A} adalah $MN \times MN$ matriks modulasi. Sedangkan \mathbf{d} adalah sebuah vektor kolom data dengan ukuran $KM \times 1$ yang berisikan tumpukan semua elemen dari unit pemetaan. Blok diagram dari modulator GFDN dapat dilihat pada Gambar 2.8 berikut



Gambar 2.8 Blok diagram pada modulator GFDM [11]

2.4.2 Model Kanal GFDM

Setelah melewati modulator GFDM maka akan diperoleh sinyal GFDM yakni $x_{GFDM} = x[0], x[1], \dots, x[N-1]$. $\mathbf{h} = [h_0, \dots, h_{N_{ch}-1}]$ merupakan kanal respon impuls dengan panjang N_{ch} . Kemudian transmisi yang dilakukan melalui kanal nirkabel dimodelkan dengan persamaan 2.22 [23].

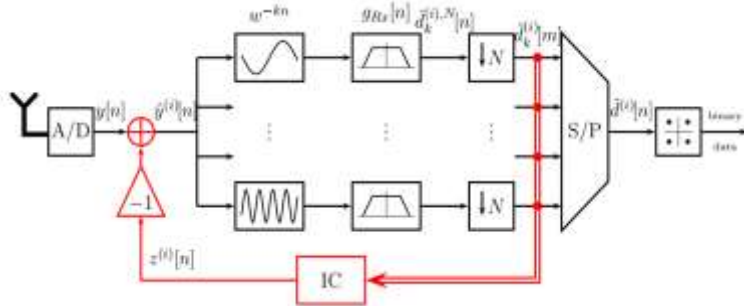
$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w} = \mathbf{H}\mathbf{A}\mathbf{d} + \mathbf{w} \quad 2.22$$

dimana $\mathbf{H} = \text{circ}\{\hat{\mathbf{h}}\}$ adalah matriks saluran konvolusi *circulant* dan $\hat{\mathbf{h}}$ adalah *zero padded* versi \mathbf{h} dengan panjang yang sama dengan \mathbf{x} . $\mathbf{w} \sim \mathcal{CN}(0, \sigma_w^2 \mathbf{I}_{KM})$ menyatakan derau *additive white Gaussian* (AWGN) dan \mathbf{I}_{KM} adalah identitas matriks dari orde KM . Elemen individual $y(n)$ dari \mathbf{y} adalah $y(n) = x(n) * h(n) + w(n)$, dimana $*$ menunjukkan operasi konvolusi dan $h(n)$ menunjukkan respon impuls dari kanal. Apabila $h(n)$ bernilai 1 maka model kanal adalah kanal ideal.

2.4.3 Penerima GFDM

Terdapat tiga cara untuk memperoleh kembali sinyal yang diterima yaitu *zero forcing* (ZF), *matched filter* (MF), dan *minimum mean square error* (MMSE). Sebuah matriks yang mempunyai invers disebut *invertible* atau matriks non singular, sedangkan matriks yang tidak mempunyai invers disebut matriks singular. Jika \mathbf{A} dan \mathbf{B} adalah matriks persegi, maka berlaku $\mathbf{A}\mathbf{B} = \mathbf{B}\mathbf{A} = \mathbf{I}$ apabila \mathbf{A} dan \mathbf{B} saling invers dan \mathbf{I} adalah matriks identitas. Pada penerima ZF, matriks penerima \mathbf{B} yang digunakan adalah \mathbf{A}^{-1} . Penggunaan *receiver* ZF sepenuhnya

menghilangkan *self-interference* namun mengakibatkan penguatan *noise* pada sinyal.



Gambar 2.9 Model penerima *baseband matched filter* GFDM [1]

Persamaan 2.23 dibawah ini menjelaskan *zero forcing* pada *receiver*. [1]

$$\hat{d}_{ZF} = \mathbf{A}^{-1}z \quad 2.23$$

Cara kedua adalah dengan menerapkan sebuah *matched filter* \mathbf{A}^H pada *receiver*. Penggunaan MF dapat memaksimalkan SNR tiap *subcarrier* di penerima, namun dapat timbul *self-interference* ketika sinyal yang dikirim menggunakan pulsa *non-orthogonal*. Persamaan 2.24 dibawah ini menjelaskan *matched filter* pada *receiver* [1]

$$\hat{d}_{MF} = \mathbf{A}^H z \quad 2.24$$

Cara ketiga adalah dengan menggunakan *minimum mean square error* (MMSE) *receiver* seperti pada persamaan 2.25. \mathbf{R}_w^2 menyatakan kovarian matriks suatu *noise*. [11]

$$\hat{d}_{MMSE} = ((\mathbf{R}_w^2 + \mathbf{A}^H \mathbf{H}^H \mathbf{H} \mathbf{A})^{-1} \mathbf{A}^H \mathbf{H}^H) z \quad 2.25$$

Receiver $((\mathbf{R}_w^2 + \mathbf{A}^H \mathbf{H}^H \mathbf{H} \mathbf{A})^{-1} \mathbf{A}^H \mathbf{H}^H)$ memberikan efek *trade-off* antara *self-interference* dan penguatan *noise*. Selanjutnya *output* dari demodulator GFDM \hat{d} akan diproses *demapper* untuk menghasilkan deretan bit di penerima yaitu \hat{b}_c . Kemudian dilakukan *decoding* untuk mengubah \hat{b}_c menjadi \hat{b} .

2.4.4 Prinsip Kerja GFDM-OQAM

Skema GFDM-OQAM memiliki komponen yang sama seperti GFDM namun dengan beberapa perbedaan. Perbedaannya adalah menggunakan pemetaan QAM yang diikuti dengan pergeseran sampel $K/2$ dalam domain waktu antara komponen *inphase* dan *quadrature* dari data kompleks QAM.

Sebagai ganti penggunaan simbol-simbol kompleks *baseband*, dalam skema GFDM-OQAM simbol bernilai riil yang dimodulasi oleh *offset* QAM ditransmisikan pada setiap *subcarrier* dengan sintesis fungsi dasar yang kemudian diperoleh dengan menerjemahkan format *time-frequency* dari filter prototipe fungsi ini yang dituliskan pada persamaan 2.26 :

$$g_{k,m}(n) = g \left[\left(n - \frac{mK}{2} \right) \bmod KM \right] e^{\frac{j2\pi k L_p - 1}{K} n} \quad 2.26$$

Dimana $n = 0, 1, \dots, KM - 1$, $g_{k,m}(n)$ adalah filter prototipe $g(n)$ yang digeser pada domain waktu dan frekuensi. Operasi modulasi menggambarkan pergeseran melingkar saat $g_{k,m}$ berada pada titik ke $-n$. Kompleks eksponensial merupakan komponen fasa dari filter prototipe yang mengalami *delay* yang juga dilakukan operasi pergeseran sesuai frekuensi. L_p adalah panjang dari filter prototipe. Sehingga superposisi dari semua simbol yang ditransmisikan mengarah ke *discrete time baseband*. *Output* modulator dari GFDM-OQAM dapat ditulis sebagai berikut:

$$x(n) = \sum_{k=0}^{K-1} \sum_{m=0}^{M-1} a_k(m) g_{k,m}(n) e^{j\phi_{m,k}} \quad 2.27$$

dimana $a_k(m)$ adalah data kompleks yang diperoleh dari bagian kompleks hasil pemetaan QAM dan $\phi_{m,k} = \frac{(k+m)\pi}{2}$ sehingga $e^{j\phi_{m,k}}$ memastikan bahwa perbedaan fasa adalah $\frac{\pi}{2}$ pada domain waktu dan frekuensi antar titik $a_k(m)$.

2.4.5 Cyclic Prefix

Cyclic prefix atau CP adalah deretan bit yang dibentuk untuk menyalin ulang bagian akhir bit-bit suatu simbol yang kemudian ditempatkan di awal simbol tersebut. Untuk sistem GFDM penggunaan CP diletakkan pada masing-masing blok GFDM yang terdiri atas beberapa *subsymbol* [11]. Gambar 2.10 menunjukkan penggunaan CP pada sistem GFDM.

Cyclic Prefix	$s_0[0]$	$s_0[1]$	$s_0[2]$	\vdots	$s_0[M-1]$
	$s_1[0]$	$s_1[1]$	$s_1[2]$		$s_1[M-1]$
	$s_2[0]$	$s_2[1]$	$s_2[2]$		$s_2[M-1]$
	\vdots	\vdots	\vdots		\vdots
	$s_{N-1}[0]$	$s_{N-1}[1]$	$s_{N-1}[2]$		$s_{N-1}[M-1]$

Gambar 2.10 *Cyclic prefix* pada paket GFDM

CP berfungsi untuk menjaga keorthogonalan antar *subcarrier* sehingga tidak terjadi interferensi antar simbol (ISI). CP juga berfungsi sebagai *guard interval* yang memastikan semua simbol GFDM terkirim secara lengkap dalam interval FFT dengan waktu tunda yang sama. CP ditambahkan saat setelah melewati proses IFFT dan akan dibuang kembali sebelum dilakukan proses FFT di penerima.

2.4.6 Filter Root Raised Cosine

Inter symbol interference (ISI) yang muncul pada sinyal yang diterima dapat dihilangkan apabila sinyal yang dikirimkan melalui tahap *pulse shaping*. Filter yang digunakan harus memenuhi kriteria *Nyquist* sehingga dapat menghilangkan ISI. Suatu pembentukan sinyal akan menghasilkan sinyal bebas ISI jika memiliki *respon impulse*:

$$h_{eff}(t) = \frac{\sin\frac{\pi t}{T}}{\pi t} \cdot z(t) \quad 2.25$$

Filter yang memenuhi kriteria *Nyquist* disebut dengan filter *Nyquist*. Diasumsikan bahwa distorsi pada kanal transmisi dapat dihilangkan dengan *equalizer* yang mempunyai fungsi transfer berupa invers dari respon kanal sehingga fungsi transfer $H_{eff}(f)$ dapat diperoleh dengan mengalikan fungsi transfer $\sqrt{H_{eff}(f)}$ pada *transmitter* dan *receiver*. Akibatnya adalah respon *matched filter* yang dapat meminimalkan baik *bandwidth* (*bandlimit*) maupun ISI.

Sinyal yang dikirimkan pada sistem GFDM perlu melalui proses *pulse shaping* menggunakan filter *square root raised cosine* dengan cara mengkonvolusikan simbol dengan filter respon impuls. Filter *pulse shaping root raised cosine* (RRC) dikembangkan dari *raised cosine* (RC) yang diterapkan di pemancar dan penerima. *Pulse shaping RC* didesain pada domain waktu didasarkan pada *roll of factor* α pada persamaan 2.26.

$$g_{RC}(t) = \begin{cases} 1, & \text{for } |t| \leq (1 - \alpha)\frac{T}{2} \\ \frac{1}{2} \left[1 + \cos\left(\pi\left(\frac{t - (1 - \alpha)\frac{T}{2}}{\alpha T}\right)\right) \right], & \text{for } (1 - \alpha)\frac{T}{2} < |t| \leq (1 + \alpha)\frac{T}{2} \\ 0, & \text{others} \end{cases} \quad 2.26$$

dengan:

$g_{RC}(t)$: *pulse shaping RC*

α : *roll of factor*, dengan nilai $0 \leq \alpha \leq 1$

T : periode simbol GFDM

Untuk memperoleh *half-Nyquist RRC* atau $g_{RRC}(t)$ adalah dengan cara mengakar kuadrat fungsi $g_{RC}(t)$. Pulsa *meyer RRC* juga menggunakan fungsi *meyer auxiliary* $v(t)$ pada domain waktu seperti pada persamaan 2.27.

$$g_{RRC}(t) = \begin{cases} 1, & \text{for } |t| \leq (1 - \alpha) \frac{T}{2} \\ \sqrt{\frac{1}{2}} \left[1 + \cos\left(\pi v \left(\frac{t - (1 - \alpha) \frac{T}{2}}{\alpha T}\right)\right) \right], & \text{for } (1 - \alpha) \frac{T}{2} < |t| \leq (1 + \alpha) \frac{T}{2} \\ 0, & \text{others} \end{cases} \quad 2.27$$

Kemudian $v(t) = t^4(35 - 84t + 70t^2 - 20t^3)$ adalah fungsi *meyer auxiliary* dengan jarak yang mendefinisikan $0 \leq \alpha \geq 1$. Dengan mengatur nilai α maka bentuk filter dapat bervariasi sesuai dengan kebutuhan.

2.5 Kanal Propagasi

Kanal adalah media yang terletak diantara pemancar dan penerima. Pada kanal ideal, sinyal yang diterima merupakan sinyal tunggal yang merupakan sinyal yang dikirimkan atau dengan kata lain sinyal yang diterima dapat direkonstruksi secara sempurna di penerima. Skenario ini tidak mungkin terjadi dikarenakan sinyal yang dikirimkan memerlukan waktu untuk dapat mencapai penerima. Sebagai tambahan, redaman yang terjadi pada *loss free environment* adalah proporsional dengan kuadrat jarak antara pengirim dan penerima. Sehingga kanal ideal yang dimaksud adalah kanal yang hanya memberikan delay dan redaman pada sinyal yang dikirimkan.

Pada kenyataannya, sinyal yang dikirimkan akan mengalami pantulan, pembelokan, dan hamburan dikarenakan kondisi halur yang dilewati oleh sinyal. Selain itu terdapat pula *noise* yang dapat menyebabkan perubahan karakteristik sinyal yang dikirim baik perubahan amplitudo maupun pergeseran frekuensi sinyal pembawa. Untuk itu dalam mensimulasikan sistem komunikasi diperlukan pemodelan kanal yang sesuai dengan kondisi lapangan. Model kanal yang sering digunakan pada sistem komunikasi *wireless* antara lain model kanal AWGN dan kanal *multipath fading* terdistribusi *Rayleigh* atau terdistribusi *Rician*.

2.5.1 Kanal AWGN

Kanal ini merupakan pemodelan yang menggambarkan dimana gangguan yang terjadi pada sinyal yang dikirimkan berasal dari *noise thermal* dan tidak terdapat gangguan yang lain seperti *fading*. Dengan kata lain kanal AWGN dapat dianggap sebagai kanal ideal. AWGN atau *Additive White Gaussian Noise* memiliki sifat aditif atau menambahkan pada sinyal yang dikirimkan. Sifat *white* dimana daya dari *noise* bernilai sama pada keseluruhan pita frekuensi sehingga *power spectral density* dari *white noise* bernilai konstan untuk rentang $-\infty$ hingga $+\infty$. Gaussian dikarenakan distribusi probabilitas dari sampel *noise* adalah Gaussian yang memiliki rataan (*mean*) nol dan standar deviasi satu.

Sinyal yang diterima dalam sistem komunikasi pada selang waktu $0 < t < T$ dapat dituliskan dalam persamaan dibawah ini:

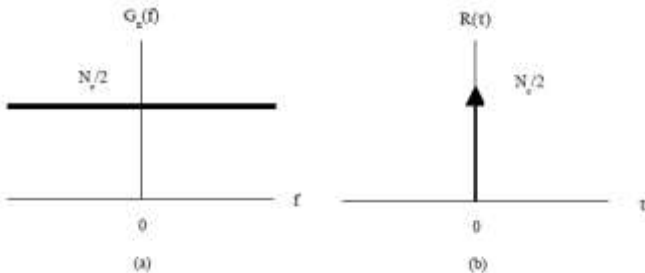
$$r(t) = s(t) + n(t), 0 < t < T \quad 2.28$$

dengan $n(t)$ mewakili *noise* kanal sebagai sebuah proses acak *zero mean Gaussian*. Jika dimisalkan notasi *psd Gaussian* adalah $G_n(f)$ maka dapat ditulis persamaannya dengan [9]:

$$G_n(f) = \frac{N_0}{2} \text{ watt/Hz} \quad 2.29$$

Dimana faktor dua menunjukkan bahwa $G_n(f)$ memiliki rapat spektral daya pada dua sisi. Lalu untuk memperoleh fungsi autokorelasi *white noise* dapat menggunakan operasi *Inverse Fast Fourier Transform (IFFT)* pada rapat spektral dayanya yang dinyatakan sebagai berikut [9]:

$$R_n = F^{-1}\{G_n(f)\} = \frac{N_0}{2} \cdot \delta(\tau) \quad 2.30$$



Gambar 2.11 (a) Rapat Spektral Daya *White Noise* (b) Fungsi Otokorelasi *White Noise* [5]

2.6 Elemen Nonlinear Pada Sistem Komunikasi Digital

Dalam sistem komunikasi digital, umumnya sinyal yang diterima akan tidak sama dengan sinyal yang dikirimkan. Hal ini dikarenakan elemen-elemen pada sistem komunikasi yang memberikan efek distorsi baik pada sisi pengirim, kanal komunikasi, dan sisi penerima. Sumber utama penyebab distorsi ini berasal dari perangkat *transmitter* dan kanal. Distorsi yang berasal dari perangkat umumnya disebabkan karakteristik operasional dari komponen perangkat seperti *power amplifier* atau dikarenakan cacat produksi. Sedangkan distorsi yang terjadi pada kanal dapat disebabkan oleh *fading* atau propagasi *multi-path* pada media nirkabel [12].

2.6.1 High Power Amplifier [12]

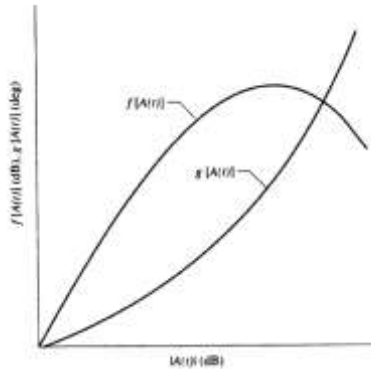
Distorsi nonlinear adalah distorsi yang terjadi pada sinyal yang terkirim disebabkan oleh salah satu elemen pada sisi *transmitter* yang bekerja pada daerah nonlinear. Elemen-elemen tersebut antara lain *high*

power amplifier dan *digital-to-analog* dan *analog-to-digital* converter [13]. *Power amplifier* adalah perangkat yang digunakan pada *transmitter* untuk menaikkan daya sehingga sinyal yang dikirimkan dapat mencapai sisi penerima. *Power amplifier* yang bekerja secara ideal akan meningkatkan daya sinyal secara proporsional sesuai dengan *gain* dari *power amplifier*, kenaikan daya input akan berakibat dengan kenaikan pada daya output. Namun secara praktik, *power amplifier* akan mengalami saturasi penguatan pada daya input tertentu. Hal ini disebabkan keterbatasan daerah operasional dari *power amplifier* yang berujung pada *clipping* atau distorsi pada sinyal. Efek distorsi yang disebabkan oleh *power amplifier* dapat dilihat pada domain frekuensi dimana sinyal informasi memiliki tambahan komponen frekuensi yang kita sebut dengan *sideband*. *Sideband* dapat menyebabkan interferensi dengan sinyal yang berdekatan sehingga mempengaruhi informasi yang dibawa sinyal tersebut .

Cara mencegah *power amplifier* dari bekerja pada daerah saturasi antara lain dengan memasang *input back-off* yang tinggi sehingga memaksa *amplifier* untuk bekerja di daerah linear. Namun kelemahan dari teknik ini adalah menurunnya efisiensi kerja dari *power amplifier*. Jika *power amplifier* bekerja dengan *input back-off* yang rendah, maka sinyal akan bekerja dengan efisiensi tinggi namun rentan terhadap distorsi. Sehingga dalam pengoperasiannya akan ada *trade-off* antara *input back-off* dengan efisiensi dimana *power amplifier* akan bekerja mendekati daerah saturasi untuk mencapai penguatan yang maksimal dan distorsi minimum atau dengan kata lain performa yang optimal.

Sebuah sinyal dapat dibedakan berdasarkan frekuensinya, amplitudonya, dan fasanya. Ketiga parameter ini dapat dipengaruhi oleh distorsi nonlinear. Sedangkan untuk sinyal *multi-carrier*, pengaruh dari distorsi nonlinear di definisikan oleh *amplitude distortion* dan *phase distortion*. *Amplitude distortion* di definisikan dengan konversi AM-AM (*Amplitude Modulation – Amplitude Modulation*) dari sinyal yang dapat dilihat pada sebuah grafik. Grafik ini membandingkan antara nilai amplitudo sinyal *output power amplifier* yang mengalami distorsi terhadap nilai amplitudo dari sinyal *input*. *Phase distortion* di definisikan dengan konversi AM-PM (*Amplitude Modulation – Phase Modulation*) dari sinyal yang dapat dilihat pada sebuah grafik. Grafik ini membandingkan antara nilai fasa sinyal *output power amplifier* yang terdistorsi terhadap nilai amplitudo dari sinyal *input*.

Pada Gambar 2.12, $f[A(t)]$ merepresentasikan fungsi konversi AM-AM dan $g[A(t)]$ merepresentasikan fungsi konversi AM-PM. Dari grafik kita dapat melihat untuk fungsi AM-AM pada nilai sinyal *input* tertentu mengalami penurunan *output*. Nilai *input* tadi disebut sebagai titik saturasi dari fungsi AM-AM dan hal ini menandakan bekerjanya *power amplifier* pada daerah nonlinear (saturasi).



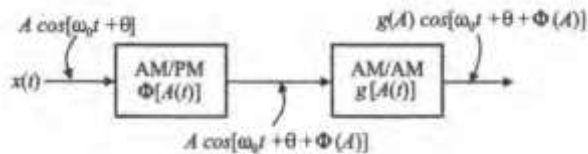
Gambar 2.12 Grafik AM-AM dan AM-PM [6]

Begitu pula dengan fungsi konversi AM-PM dimana *power amplifier* dapat menggeser fasa dari sinyal *input*. Efek lain yang dapat dilihat juga pada konstelasi sinyal yang diterima. Berikut adalah perbandingan konstelasi sinyal input sebelum dilewatkan *power amplifier* dan setelah diterima di *receiver*.

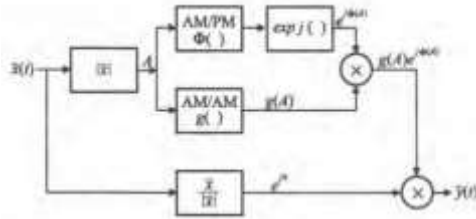
Jika sebuah sinyal bandpass $x(t)$ dilewatkan melalui *power amplifier* sehingga didapatkan $y(t)$ maka dapat digambarkan ke dalam blok diagram [6].

$$y(t) = g[A(t)]\cos\{2\pi f_c t + \theta(t) + \Phi[A(t)]\} \quad 2.31$$

Gambar 2.13 merupakan pemodelan HPA dalam sistem *bandpass*. Sedangkan Gambar 2.14 merupakan pemodelan HPA dengan menggunakan simulasi.



Gambar 2.13 Model Sistem *Bandpass* [6]



Gambar 2.14 Model Simulasi Kompleks Ekvivalen [6]

Persamaan 2.31 dapat dikembangkan berdasarkan identitas trigonometri menjadi

$$y(t) = S_p(t) \cos[2\pi f_c t + \theta(t)] - S_q(t) \sin[2\pi f_c t + \theta(t)] \quad 2.32$$

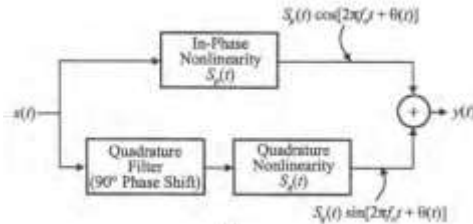
dimana

$$S_p(t) = g[A(t)] \cos \Phi[A(t)] \quad 2.33$$

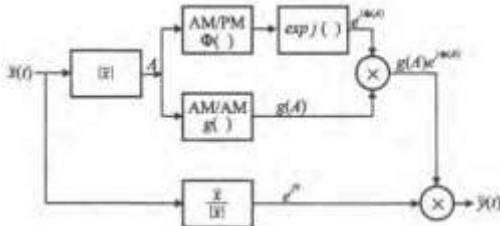
dan

$$S_q(t) = g[A(t)] \sin \Phi[A(t)] \quad 2.34$$

Berdasarkan persamaan 2.32, maka dapat dirancang pemodelan alternatif yang dapat dilihat pada Gambar 2.15 dan 2.16.



Gambar 2.15 Model Alternatif Sistem *Bandpass* [6]



Gambar 2.16 Model Alternatif Simulasi Kompleks Ekvivalen [6]

Kemudian untuk melengkapi persamaan 2.32, diperlukan suatu metode yang dapat merepresentasikan variabel $g(A)$ dan $\Phi(A)$. Terdapat dua metode untuk merepresentasikan kedua variabel tersebut. Metode

pertama adalah dengan merancang tabel dan nilai *lookup* selama simulasi berlangsung. Metode ini memerlukan interpolasi yang akan valid apabila nilai-nilai yang didapatkan memiliki jarak yang berdekatan. Metode kedua adalah dengan menggunakan representasi analitis yang dapat menggantikan peran pengukuran dengan sebuah fungsi yang mudah untuk di evaluasi. Jika fungsi yang dirancang memiliki hasil yang baik, maka *error* dapat diminimalisir dan model bisa lebih efisien [6].

2.6.2 Amplifier TWT Model Saleh

Dalam mensimulasikan kinerja dari *power amplifier*, diperlukan suatu model simulasi kompleks ekivalen *bandpass* seperti yang telah dituliskan pada persamaan 2.32. Namun dalam persamaan tersebut, variabel $g(A)$ dan $\Phi(A)$ belum representasikan. Maka dalam tugas akhir ini akan digunakan metode representasi analitis untuk kedua variabel tersebut. Representasi analitis didasarkan pada penelitian dari Adel Saleh [14], dimana beliau mengajukan suatu persamaan untuk merepresentasikan kedua variabel tersebut:

$$g(A) = \frac{\alpha_g A}{1 + \beta_g A^2} \quad 2.35$$

dan

$$\Phi(A) = \frac{\alpha_\Phi A^2}{1 + \beta_\Phi A^2} \quad 2.36$$

Sebagai catatan untuk nilai A yang sangat besar, nilai $g(A)$ sebanding dengan $1/r$ dan nilai $\Phi(A)$ mendekati konstan. Sedangkan dalam bentuk *inphase* dan *quadrature*, $S_p(A)$ dan $S_q(A)$ dapat dituliskan pada persamaan 2.37 dan 2.38

$$S_p(A) = \frac{\alpha_p A}{1 + \beta_p A^2} \quad 2.37$$

dan

$$S_q(A) = \frac{\alpha_q A^3}{(1 + \beta_q A^2)^2} \quad 2.38$$

Hal yang perlu diperhatikan selanjutnya adalah untuk persamaan $g(A)$, $\Phi(A)$, $S_p(A)$, dan $S_q(A)$ memiliki parameter yang didapatkan berdasarkan pengujian persamaan 2.35, 2.36, 2.37, dan 2.38 terhadap data uji coba. Dalam penelitiannya, Adel Saleh telah mendapatkan nilai-nilai optimum untuk masing-masing parameter berdasarkan data dari beberapa uji coba amplifier TWT. Nilai-nilai tersebut dituliskan didalam Tabel 2.1.

Tabel 2.1 Parameter optimum untuk persamaan HPA model Saleh [14]

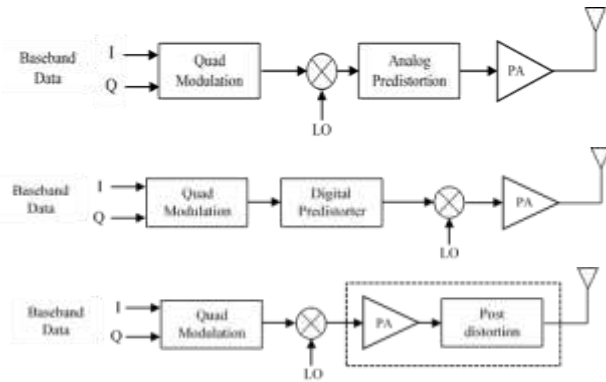
Model	Persamaan	α	β	Error RMS
1	$g(A)$	1.9638	0.9945	0.012
	$\Phi(A)$	2.5293	2.8168	0.478
2	$S_p(A)$	1.6397	0.0618	0.057
	$S_q(A)$	0.2038	0.1332	0.023
	$g(A)$	1.6623	0.0552	0.041
	$\Phi(A)$	0.1533	0.3456	0.508
3	$S_p(A)$	2.0922	1.2466	0.015
	$S_q(A)$	5.5290	2.7088	0.009
	$g(A)$	2.1587	1.1517	0.01
	$\Phi(A)$	4.0033	9.1040	0.469

2.7 Linearisasi High Power Amplifier

Untuk menghilangkan efek distorsi dari RF *power amplifier* (PA) diperlukan sebuah teknik yang dinamakan linearisasi. Untuk sistem komunikasi radio yang menggunakan skema modulasi digital *advance* yang umumnya memiliki nilai *peak-to-average ratio* tinggi, teknik linearisasi sangat dibutuhkan sehingga dicapai efisiensi yang tinggi [15]. Inti dari perbaikan sinyal yang terdistorsi akibat RF *power amplifier* adalah *power amplifier* dioperasikan sedekat mungkin dengan daerah saturasi untuk mencapai amplifikasi maksimum dengan daya masuk yang efisien dan digabungkan dengan penggunaan teknik linearisasi tertentu untuk menekan distorsi yang muncul saat PA dioperasikan di dekat daerah saturasi. Jenis- jenis dari teknik linearisasi yang sering digunakan antara lain *Back-Off* RF PA, predistorsi RF, *feedback*, dan *feedforward*.

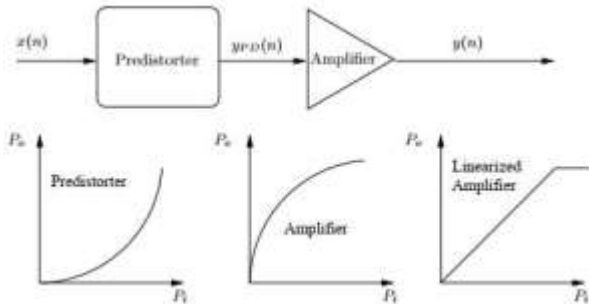
2.7.1 Teknik Linearisasi Predistorsi RF

Teknik predistorsi RF adalah teknik yang secara aktif melacak dan menerapkan invers pada nonlinearitas dari *amplifier*. Jika sebuah *amplifier* menerapkan *gain compression* pada sinyal maka predistorsi akan di desain untuk menerapkan *gain expansion* pada sinyal sehingga efek dari *amplifier* dapat ditekan. Teknik predistorsi RF menggunakan elemen nonlinear analog aktif maupun pasif untuk dapat menghasilkan sinyal yang telah ter-predistorsi. Elemen predistorsi RF dapat diterapkan pada sinyal *passband*, sinyal *baseband*, atau diterapkan pasca elemen nonlinear seperti PA. Sistem predistorsi RF menggunakan elemen nonlinear aktif atau pasif yang bekerja di frekuensi RF untuk menghasilkan sinyal yang telah di predistorsi.



Gambar 2.17 Skema linearisasi distorsi non-linear [15]

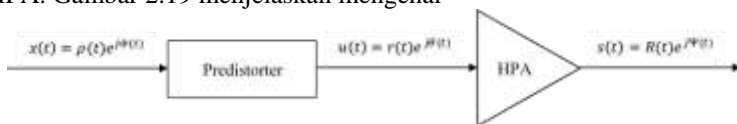
Implementasi lain dari sistem predistorsi adalah dengan melakukan predistorsi pada sinyal *baseband* dimana invers dari *transfer function* sinyal diberikan sebelum dilakukan modulasi pada sinyal karier. Predistorsi yang dilakukan pada sinyal *baseband* termasuk kedalam predistorter digital.



Gambar 2.18 Cara Kerja *Predistorter* dalam Blok Diagram [16]

2.7.2 Invers Model Saleh [17]

Dengan membuat invers dari Model Saleh, maka sebuah predistorter dapat dikembangkan untuk mengatasi distorsi non-linear dari HPA. Gambar 2.19 menjelaskan mengenai



Gambar 2.19 Blok diagram *Predistorter* dan HPA

Untuk sinyal keluaran dari HPA maka kondisi dibawah ini perlu dipenuhi dimana:

$$R(t) = A[r(t)] \quad 2.39$$

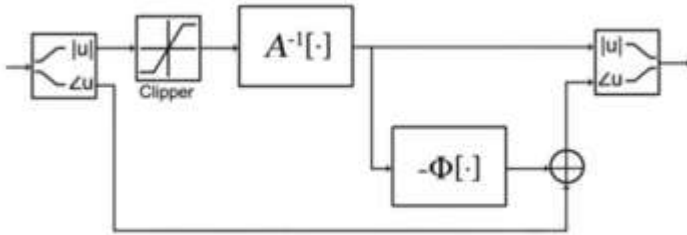
$$\phi(t) = \theta(t) + \Phi[r(t)] \quad 2.40$$

dimana $r(t)$ dan $\theta(t)$ adalah amplitudo dan fasa dari sinyal *input* HPA, sedangkan $A[\cdot]$ dan $\Phi[\cdot]$ adalah fungsi transfer AM/AM dan AM/PM dari HPA seperti pada persamaan 2.35 dan 2.36 . Jika melihat dari gambar 2.18 maka *predistorter* dapat mengkompensasi distorsi nonlinear secara menyeluruh apabila $x(t) = s(t)$ sehingga

$$r(t) = A^{-1}[\rho(t)] \quad 2.41$$

$$\theta(t) = \phi(t) - \Phi[r(t)] \quad 2.42$$

Invers dari persamaan 2.39 hanya berlaku apabila $\rho(t) \leq \sqrt{\frac{\alpha_a^2}{4\beta_a}}$ sehingga sinyal input dari predistorter perlu dibatasi.



Gambar 2.20 Blok Diagram *Predistorter* [17]

Apabila Gambar 2.19 dituliskan kedalam persamaan maka akan didapatkan sebagai berikut:

$$r(t) = \begin{cases} \frac{\alpha_a - \sqrt{\alpha_a^2 - 4\beta_a \rho(t)^2}}{2\beta_a \rho(t)}, & 0 < \rho(t) < \sqrt{\frac{\alpha_a^2}{4\beta_a}} \\ A^{-1}\left(\sqrt{\frac{\alpha_a^2}{4\beta_a}}\right) = \frac{1}{\sqrt{\beta_a}}, & \rho(t) \geq \sqrt{\frac{\alpha_a^2}{4\beta_a}} \end{cases} \quad 2.43$$

$$\theta(t) = \phi(t) - \frac{\alpha_\phi r^2(t)}{1 + \beta_\phi r^2(t)} \quad 2.44$$

dimana $r(t)$ adalah amplitudo sinyal output dari predistorter dan $\theta(t)$ adalah fasa sinyal output dari predistorter.

2.8 Teknik Estimasi BER

Performansi dari simulasi sistem komunikasi umumnya dilihat dengan cara menghitung laju error yang berasal dari kesalahan urutan data acak yang melewati suatu model kanal dikarenakan penambahan *noise* aditif. Terdapat beberapa metode untuk mengestimasi BER, salah satunya adalah metode simulasi *Monte Carlo*. Metode simulasi *Monte Carlo* (MC) merupakan metode simulasi estimasi BER yang relatif sederhana namun memerlukan waktu.

Simulasi BER dengan menggunakan metode simulasi MC pada MATLAB® dapat dilakukan dengan mensimulasikan pengiriman data acak melalui model kanal tertentu yang kemudian data hasil pengiriman akan dibandingkan dengan data yang dikirimkan [18]. Jika dalam sistem komunikasi terdapat sebanyak N *bit* yang diproses dan ada n *bit* yang salah/error, maka BER dapat dihitung dengan mudah yaitu dengan membandingkan total *bit* yang error dengan total *bit* yang dikirimkan, sesuai dengan persamaan (2.40). Sedangkan dalam simulasi MATLAB® dapat digunakan fungsi *biterr* untuk menghitung BER dengan membandingkan data sebelum dikirimkan dengan data setelah diterima.

$$BER = \frac{\text{jumlah bit error}}{\text{jumlah bit yang diterima}} \quad 2.45$$

2.9 Power Spectral Density

Untuk dapat memodelkan distorsi pada sistem komunikasi digital, PSD dari sinyal yang diterima perlu di estimasi. Fungsi autokorelasi dari *complex envelope* suatu sinyal acak didefinisikan pada persamaan 2.46

$$R_{\tilde{x}\tilde{x}} = E[\tilde{x}(t)\tilde{x}(t + \tau)] \quad 2.46$$

PSD dapat dicari dengan melakukan transformasi Fourier pada fungsi autokorelasi diatas sesuai dengan teorema Wiener-Khinchin (Papoulis,1994) [18] sehingga menghasilkan:

$$S_{\tilde{x}\tilde{x}} = \int_{-\infty}^{\infty} R_{\tilde{x}\tilde{x}}(\tau)e^{-j\omega\tau}d\tau \quad 2.47$$

Secara umum fungsi auto korelasi dari suatu sinyal input $x(t)$ yang memiliki frekuesni tengah f_0 memiliki keterkaitan dengan fungsi autokorelasi *complex envelope* sinyal dimana:

$$\begin{aligned} R_{xx}(\tau) &= E[x(t + \tau)x(t)] \\ &= \Re\{E[\tilde{x}(t + \tau)\tilde{x}^*(t)]e^{j2\pi f_0\tau}\} \\ &= \Re\left\{\frac{1}{2}R_{\tilde{x}\tilde{x}}(\tau)e^{j2\pi f_0\tau}\right\} \end{aligned} \quad 2.48$$

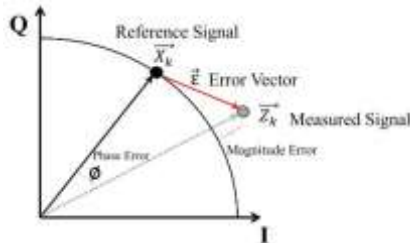
Sedangkan untuk sinyal *bandpass* yang memiliki frekuensi tengah f_0 , memiliki keterkaitan dengan PSD dari *complex envelope* dengan menggunakan salah satu sifat dari transformasi Fourier (Peebles, 1987):

$$S_{xx}(f) = \frac{1}{4} [S_{\bar{x}\bar{x}}(f - f_0) + S_{\bar{x}\bar{x}}(f + f_0)] \quad 2.49$$

2.10 Error Vector Magnitude

Error vector magnitude (EVM) adalah salah satu *figure of merit* yang umum digunakan untuk mengukur standar linearitas sistem komunikasi digital. Sinyal yang dikirimkan oleh *transmitter* ideal akan secara tepat tiba di lokasi ideal yang telah ditentukan di awal. Namun berbagai jenis gangguan pada sistem komunikasi seperti nonlinearitas dari HPA menyebabkan titik konstelasi dari sinyal diterima bergeser dari lokasi idealnya. EVM didefinisikan sebagai nilai perbandingan antara akar RMS dari selisih antara kumpulan simbol terukur dengan simbol-simbol ideal dengan akar dari rata-rata daya sinyal. Dari definisi tersebut EVM dapat dituliskan sebagai berikut [19]:

$$EVM = \sqrt{\frac{E\{|z(t) - x_k(t)|^2\}}{E\{|x_k(t)|^2\}}} \quad 2.50$$



Gambar 2.21 Representasi *error vector magnitude* [19]

BAB 3

PEMODELAN DAN SIMULASI SISTEM

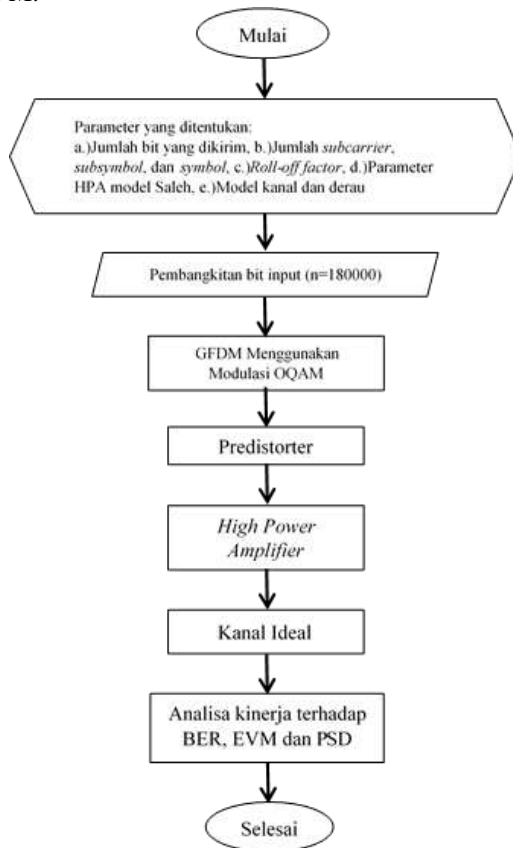
Pada bab tiga ini dijelaskan mengenai tahapan-tahapan dalam pembuatan tugas akhir. Tugas akhir ini membahas mengenai perbandingan kinerja sistem GFDM-OQAM dengan HPA sebelum dan sesudah penggunaan *predistorter*. Parameter yang akan diperhatikan adalah kurva *Bit Error Rate* (BER) dan kurva *Error Vector Magnitude* (EVM). Kedua parameter tersebut akan dijadikan acuan dalam melihat peningkatan kinerja sistem serta kinerja *predistorter* dalam mengatasi efek nonlinier pada sistem. Pemodelan sistem menggunakan model simulasi *baseband* kompleks dikarenakan model simulasi ini dapat mewakili model simulasi *passband* yang bekerja di frekuensi tinggi dan membutuhkan komputasi yang kompleks, sedangkan model simulasi *baseband* kompleks lebih mudah dan sederhana. Dalam pemodelan simulasi ini digunakan *software Matlab* versi R2015a.

3.1 Parameter Simulasi

Analisis simulasi sistem GFDM-OQAM menggunakan HPA dan HPA+Predistorter dilakukan setelah menjalankan simulasi pemodelan sistem sesuai dengan Gambar 3.1. Pada simulasi ini parameter dan asumsi yang digunakan antara lain:

- a) Input data biner yang dibangkitkan oleh *pseudo-random generator* adalah sebanyak 180000 bit dimana data tersebut merupakan data acak.
- b) Jenis modulasi yang digunakan adalah modulasi 16 *Offset-QAM*.
- c) Jumlah *subcarrier* yang digunakan pada simulasi adalah 5
- d) Jumlah *subsymbol* yang digunakan pada simulasi adalah 9
- e) Total *symbol* yang digunakan adalah 1000
- f) *Roll-off factor* yang digunakan adalah 0,5
- g) *Pulse shaping* yang digunakan adalah *root raised cosine filter*
- h) Model HPA dalam simulasi didasarkan pada HPA model Saleh dengan parameter $\alpha_f = 2,1587$; $\beta_f = 1,1517$; $\alpha_g = 4,0033$; dan $\beta_g = 9,104$
- i) Teknik predistorsi merupakan invers dari fungsi HPA model Saleh.
- j) Parameter yang dibahas adalah pengaruh penggunaan *predistorter* terhadap kinerja sistem dilihat dari BER serta dalam

mengurangi efek nonlinier pada sinyal yang diterima dilihat dari EVM.

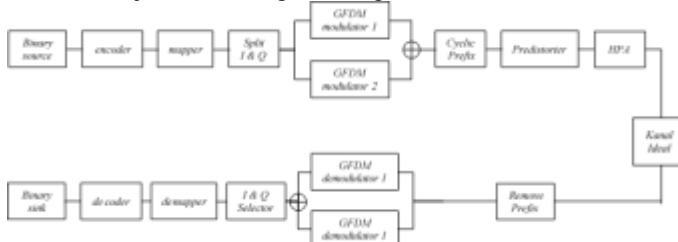


Gambar 3.1 Diagram Alir Sistem GFDM-OQAM dengan HPA

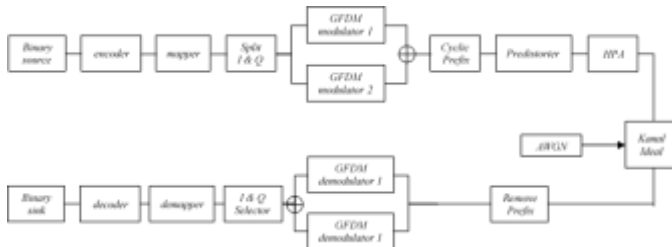
Dalam simulasi yang dilakukan akan digunakan dua skema pengukuran. Pada skema pertama adalah simulasi sistem GFDM-OQAM dengan HPA sebelum dan sesudah penggunaan *predistorter* pada kanal ideal dimana pengaruh *noise* dianggap tidak berpengaruh terhadap sinyal yang dikirimkan. Sedangkan pada skema kedua adalah simulasi sistem GFDM-OQAM dengan HPA sesudah penggunaan *predistorter* pada kanal ideal yang diberikan pengaruh *thermal noise* atau *additive white gaussian noise* (AWGN).

3.2 Pemodelan Sistem

Sistem komunikasi pada simulasi dimodelkan kedalam dua skema seperti yang telah dijelaskan pada sub bab sebelumnya. Gambar 3.2 menjelaskan mengenai diagram blok skema pertama, sedangkan Gambar 3.3 menjelaskan mengenai diagram blok skema kedua.



Gambar 3.2 Diagram Blok Transceiver Skema Pertama



Gambar 3.3 Diagram Blok Transceiver Skema Kedua

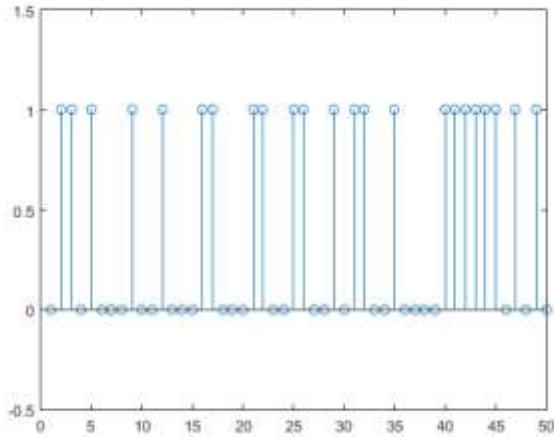
3.2.1 Bagian Transmitter

Pada bagian *transmitter* terdiri dari beberapa bagian seperti data *bit* input yang dibangkitkan oleh *pseudo-random generator*, proses *mapping*, proses *pulse shaping*, proses *Inverse Fast Fourier Transform* (IFFT), proses penambahan *cyclic prefix*, proses predistorsi sinyal informasi, penguatan sinyal dengan HPA yang kemudian ditransmisikan melalui kanal ideal. *Bit* data terlebih dahulu dibangkitkan secara acak sejumlah 180000, kemudian *bit-bit* tadi akan dipetakan. Pada tugas akhir ini proses pemetaan menggunakan modulasi 16-QAM *baseband* kompleks. Hasil dari proses ini akan menghasilkan simbol-simbol kompleks.

3.2.1.1 Informasi Input

Langkah pertama yang perlu dilakukan adalah membangkitkan sinyal secara acak dengan fungsi *randi* pada MATLAB. Data acak yang dibangkitkan memiliki nilai 0 dan 1 yang juga merupakan representasi dari sinyal digital. Pada Tugas Akhir ini, jumlah *subcarrier* dan

subsymbol yang digunakan masing masing adalah 5 dan 9 dengan jumlah simbol adalah 1000 sehingga jumlah *bit* yang dibangkitkan berjumlah 180000 *bit* informasi. Gambar 3.4 menunjukkan hasil pembangkitan sinyal input dengan pengambilan sampel sebanyak 50 bit.

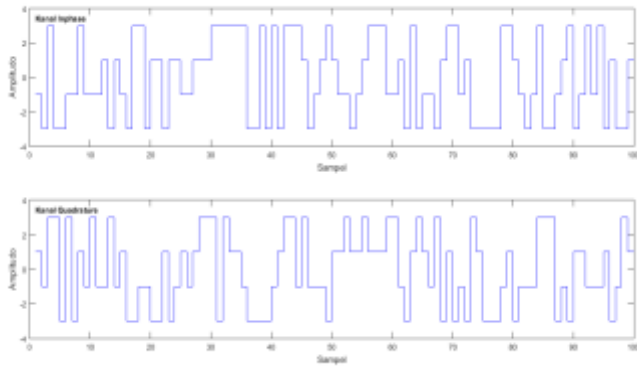


Gambar 3.4 Bit informasi awal

3.2.1.2 Proses Modulasi

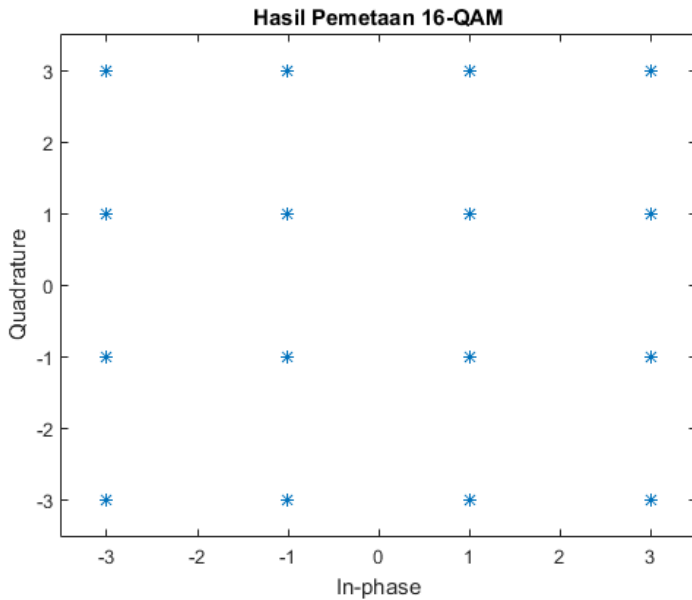
Bit acak yang telah dibangkitkan kemudian akan melalui proses modulasi. Pada Tugas Akhir ini modulasi yang digunakan adalah 16-QAM. Namun sebelum diberikan *offset*, perlu dilakukan pemetaan *bit* acak tadi dengan modulasi QAM. Hasil dari proses pemetaan ini adalah bilangan kompleks yang memiliki bagian riil dan imajiner. Dalam proses pemetaan 16-QAM setiap empat bit yang memasuki S/P converter, dua *bit* akan dikeluarkan melalui kanal I (*Inphase*) dan dua *bit* lainnya dikeluarkan melalui kanal Q (*Quadrature*). Dengan kata lain, *bit* informasi input dipecah menjadi empat kanal yaitu I, I', Q, Q' dengan masing masing kanal memiliki laju *bit* seperempat dari laju *bit* informasi input.

Setelah dipecah menjadi empat kanal, untuk kanal I dan Q akan menentukan polaritas output dimana untuk setiap *bit* 0 sama dengan negatif dan *bit* 1 sama dengan positif. Untuk kanal I' dan Q' akan menentukan magnitudo output dimana untuk setiap *bit* 0 bernilai 3V dan *bit* 1 bernilai 1V. Gambar 3.5 menunjukkan hasil pemetaan 16-QAM pada kanal I dan Q.



Gambar 3.5 Sinyal *Inphase* dan *Quadrature* hasil modulasi 16-QAM

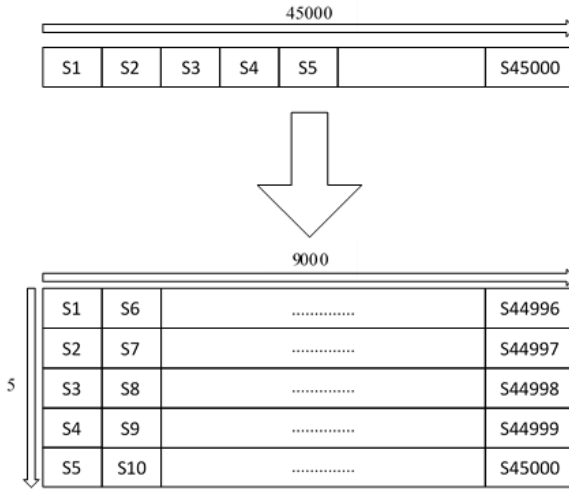
Sedangkan konstelasi dari hasil pemetaan 16-QAM dapat dilihat pada Gambar 3.6. Dari hasil pemetaan didapatkan 16 simbol yang mewakili setiap kemungkinan 4 bit informasi.



Gambar 3.6 Konstelasi Sinyal 16-QAM

3.2.1.3 Inisiasi Simbol GFDM

Setelah selesai dipetakan, ukuran dari simbol-simbol *mapping* 16-QAM adalah 1×45000 atau seperempat dari *bit* informasi. Langkah selanjutnya adalah dengan mengubah bentuk serial simbol 16-QAM menjadi matriks paralel dengan ukuran 5×9000 . Dalam simulasi Matlab, proses konversi dari serial menuju paralel dapat dilakukan dengan menggunakan fungsi *reshape*. Gambar 3.7 menunjukkan proses konversi *Serial to Parallel* pada



Gambar 3.7 Konversi *Serial to Parallel*

Selanjutnya dibuat matriks d yang mewakili satu blok GFDM dengan ukuran 8×10 yang berisikan *zero padding*. Matriks tersebut akan diisi dengan deretan simbol *mapping* berukuran $K \times M$ atau 5×9 . Sisa matriks dari blok GFDM yang tidak berisikan simbol 16-QAM atau bernilai *zero* bersama dengan input deretan simbol *mapping* akan diterapkan *circular convolution*. Matriks berukuran 8×10 ini selanjutnya diubah menjadi 80×1 untuk mewakili satu buah simbol GFDM. Masing-masing simbol GFDM kemudian di *upsampling* dengan faktor N untuk mengubahnya menjadi sinyal impuls.

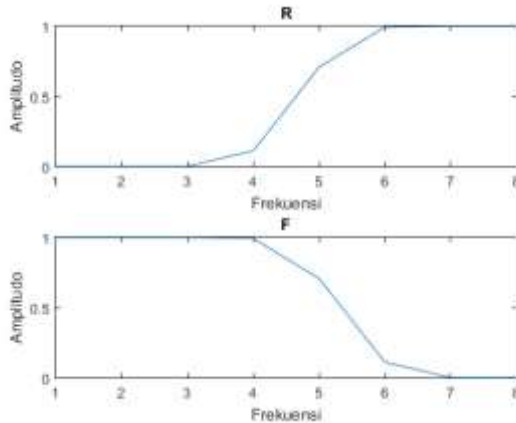
Matriks d yang terbentuk dari hasil *upsampling* memiliki ukuran 80×1000 atau $KM \times N$, sehingga d dapat disebut sebagai kumpulan blok GFDM. Matriks d tersebut kemudian dipecah menjadi di dan dq . Matriks di merupakan bagian riil dari simbol QAM dan matriks dq merupakan bagian imajiner dari simbol QAM.

3.2.1.4 Pembentukan Pulsa (*Pulse Shaping*)

Proses selanjutnya adalah *pulse shaping*, dimana hasil keluaran dari inialisasi simbol GFDM akan dikalikan dengan *pulse shapes* yang dibentuk pada domain waktu. Proses *pulse shaping* menggunakan filter *root-raised cosine* dengan *roll-off factor* sebesar 0.5. Setelah didapatkan pulsa RRC dari persamaan Meyer *root-raised cosine* yang diberikan dengan notasi g , pulsa $g[n]$ digeser secara sirkular pada domain waktu dan frekuensi untuk menghasilkan *pulse shaping*. Filter prototipe tersebut berfungsi untuk membedakan *subsymbol* berdasarkan slot waktu dan frekuensi. Karenanya hasil dari pemfilteran tersebut adalah sebuah sub-sinyal GFDM yang memiliki frekuensi dan waktu yang berbeda-beda akibat proses pemfilteran sirkular. Semua sub-sinyal yang terbentuk kemudian dijumlahkan sehingga menjadi sebuah sinyal GFDM. *Pulse shape* dari setiap *subcarrier* dan *subsymbol* dapat dikelompokkan pada sebuah matriks modulasi sebagai berikut:

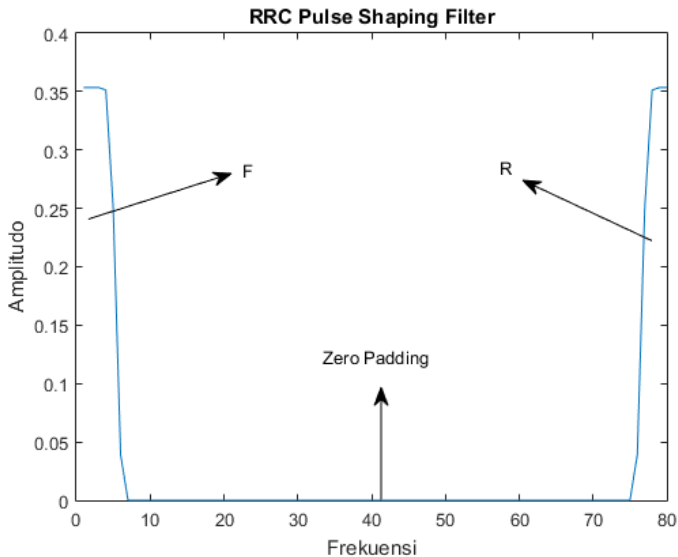
$$A = [g_{0,0} \dots g_{K-1,0} \ g_{0,1} \dots g_{K-1,M-1}] \quad 3.1$$

Dimana $g_{k,m}$ adalah sebuah vektor kolom yang berisi sampel dari $g_{k,m}[n]$. Tujuan pemberian *pulse shaping* RRC ini adalah untuk memberikan efisiensi *bandwidth* yang lebih dibandingkan dengan menggunakan *rectangular pulse*. Dalam program Matlab, *pulse shaping* g terdiri atas tiga komponen yaitu komponen F, komponen R, dan *zero padding*. Bentuk filter RRC komponen F dan R dapat dilihat pada gambar 3.8.



Gambar 3.8 Filter *root raised cosine*

Sedangkan bentuk dari *pulse shaping* g dengan α bernilai 0.5 pada domain frekuensi dapat dilihat pada gambar 3.9



Gambar 3.9 *Pulse shaping* RRC dengan $\alpha = 0.5$

3.2.1.5 Proses *Offset* QAM

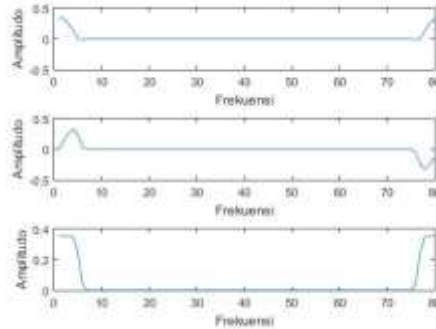
Langkah berikutnya adalah memberikan *offset* kepada simbol QAM menggunakan *pulse shaping* $g[n]$ yang telah dibangkitkan sebelumnya. Bagian *inphase* dan *quadrature* dari simbol QAM ditransmisikan menggunakan *pulse shaping* yang telah di *offset* atau digeser sebesar $K/2$ sampel pada bagian *quadrature*. Serta dilakukan rotasi *phase* sebesar $\pi/2$ diantara *subcarrier* dan *subsymbol* yang berdekatan. Proses *pulse shaping* pada *offset* QAM dapat diekspresikan pada persamaan 3.1 berikut:

$$g_{k,m}^{(i)}[n] = j^k g_{k,m}[n] \tag{3.2}$$

$$g_{k,m}^{(q)}[n] = j^{k+1} g_{k,m+\frac{1}{2}}[n]$$

Representasi hasil *pulse shape* g_i dan g_q dengan modulasi *Offset* QAM dapat dilihat pada gambar 3.10. Proses diatas pada Matlab dapat dilakukan dengan menggunakan fungsi *circshift*. Fungsi ini digunakan

untuk menggeser secara sirkular *pulse shaping* g pada proses *Offset QAM*.

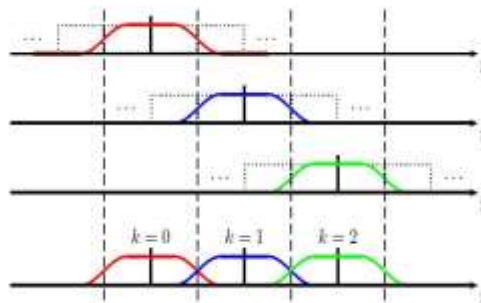


Gambar 3.10 *Pulse shaping* pada *offset QAM*

Konstelasi dari hasil *offset* 16-QAM akan berbeda dengan konstelasi dari 16-QAM dikarenakan pergeseran sampel pada domain waktu antara komponen *inphase* dan *quadrature*.

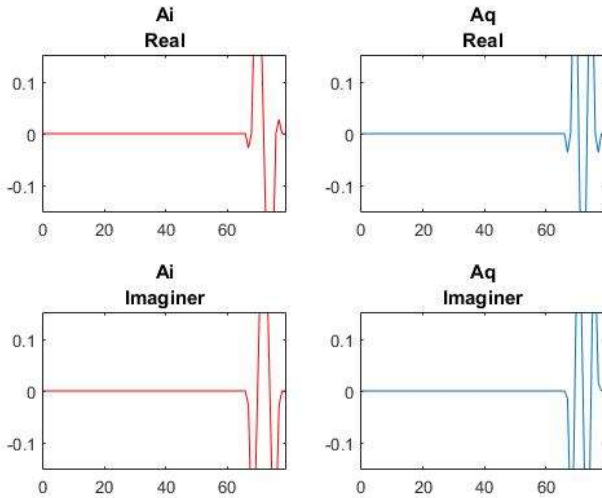
3.2.1.6 IFFT (*Inverse Fast Fourier Transform*)

Proses selanjutnya adalah proses IFFT yaitu proses untuk mengubah sinyal dari domain frekuensi ke dalam domain waktu. Dari gambar 3.11 dapat diamati bahwa setiap *subcarrier* digeser pada domain frekuensi dengan menggunakan operasi *circular convolution* [21]. Pembuatan matriks A (A_i dan A_q) menggunakan operasi ini. Sedangkan pengaplikasiannya pada program Matlab dapat menggunakan fungsi *circshift*.



Gambar 3.11 Superposisi sebelum proses IFFT

Pembuatan matriks A ini dibuat dengan melakukan operasi *circular convolution* pada *pulse shaping g*. Gambar 3.12 menunjukkan bentuk dari matriks A_i dan A_q pada bagian real dan imajiner.



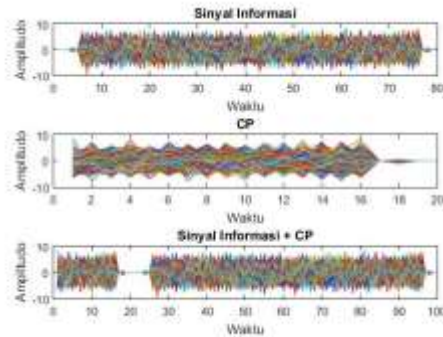
Gambar 3.12 Bentuk sinyal *subcarrier* pada *Offset QAM*

3.2.1.7 Penambahan *cyclic prefix*

Penggunaan *cyclic prefix* pada sistem komunikasi memiliki tujuan untuk mengurangi efek *intersymbol interference* antara dua simbol yang berdekatan. Prinsip penempatan *cyclic prefix* pada GFDM menyerupai penambahan *cyclic prefix* pada OFDM. Pada OFDM, CP berasal dari 25% ukuran satu simbol OFDM. CP pada OFDM diambil dari salinan akhir dari masing-masing simbol yang kemudian diletakkan diawal simbol. Sedangkan untuk GFDM, CP diletakkan pada setiap blok M simbol dari satu *frame* GFDM. Pada Tugas Akhir ini, blok GFDM yang berukuran 80×1000 akan menggunakan *cyclic prefix* sebesar 20. Simbol GFDM yang disalin adalah 20 *subsymbol* terakhir dari simbol GFDM dan diletakkan pada awal simbol sehingga total *subsymbol* yang dikirimkan adalah 100 setelah ditambahkan *cyclic prefix* sehingga blok GFDM berukuran 100×1000 . Gambar 3.13 merepresentasikan penambahan *cyclic prefix* pada blok GFDM. Sedangkan sinyal sebelum dan sesudah ditambahkan CP dapat dilihat pada gambar 3.14.

cyclic prefix	$s_0[0]$	$s_0[1]$	$s_0[2]$...	$s_0[M-1]$
	$s_1[0]$	$s_1[1]$	$s_1[2]$		$s_1[M-1]$
	$s_2[0]$	$s_2[1]$	$s_2[2]$		$s_2[M-1]$
	\vdots	\vdots	\vdots		\vdots
	$s_{N-1}[0]$	$s_{N-1}[1]$	$s_{N-1}[2]$		$s_{N-1}[M-1]$

Gambar 3.13 Format paket GFDM



Gambar 3.14 Hasil sinyal setelah ditambahkan CP

3.2.1.8 *Predistorter*

Selanjutnya sinyal yang telah ditambahkan dengan CP akan dilewatkan melalui *predistorter*. *Predistorter* yang digunakan pada simulasi adalah invers dari HPA model Saleh. Sebelum blok GFDM dilewatkan *predistorter*, blok GFDM yang awalnya memiliki ukuran 100x1000 akan diubah dari bentuk paralel menjadi serial dengan ukuran 1x100000 dengan menggunakan fungsi *reshape* pada simulasi Matlab. Sinyal yang masuk akan dibagi menjadi dua yakni amplitudo dan fasa dimana masing-masing akan diproses dengan fungsi invers AM/AM dan AM/PM. Setelah itu komponen amplitudo dan fasa akan digabungkan menjadi satu untuk dilewatkan menuju HPA. Pada tahap ini IBO juga diberlakukan pada sinyal yang akan dikirim. IBO atau *input back-off* merupakan cara untuk mengurangi nilai input sinyal beberapa level satuan dB. IBO digunakan dalam sistem komunikasi khususnya saat penggunaan HPA supaya sinyal keluaran dari HPA tidak mendekati daerah operasi

nonlinier. Dalam Tugas Akhir ini dikarenakan penggunaan predistorter dan HPA secara berurutan maka dapat dikatakan sama dengan menggunakan *linearized* HPA sehingga IBO diletakkan pada komponen *predistorter*.

Berikut sintak program untuk *predistorter*:

```
xxx=reshape(xcpOQAM,1,[]); %Sinyal sebelum
normalisasi
Pin=mean(mean(xcpOQAM.*conj(xcpOQAM)));Ex=sqrt
(Pin);
yx=xxx/Ex; %Sinyal setelah normalisasi

IBO=-15:1;%dB
kk=10.^(IBO/10); %IBO linier

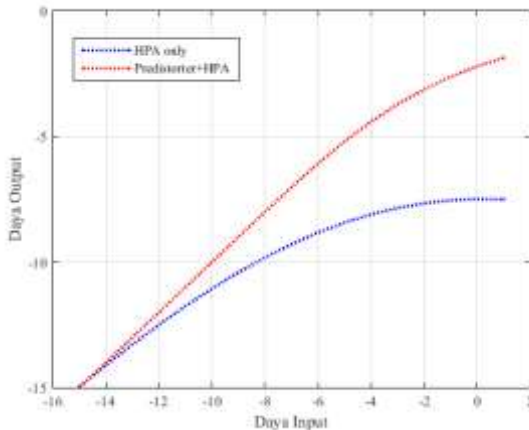
y=yx*sqrt(kk(1,k)); %Penerapan IBO pada sinyal
nx=y; %Sinyal input predistorter
xpred = abs(nx); phi = angle(nx); %xpred
adalah amplitudo sinyal input, phi adalah fasa
sinyal input
af=2.1587; bf=1.1517; ag=4.0033; bg=9.104;
c = (af^2)/(4*bf); Lim = sqrt(c); r =
zeros(1,length(xpred)); psi0 = [];
for i = 1:1:length(xpred)
    if xpred(1,i) > 0 && xpred(1,i) < Lim
        r(1,i) = (af-sqrt((af^2)-
4*bf*xpred(1,i)^2))./(2*bf*xpred(1,i));
    elseif xpred(1,i) >= Lim
        r(1,i) = 1/sqrt(bf);
    end
    psi0 = [psi0
(ag*r(1,i)^2)/(1+bg*r(1,i)^2)];
end
theta = phi - psi0;
rr = r.*exp(1i*theta); %Output predistorter
```

Perlu diperhatikan untuk sinyal sebelum memasuki *predistorter* perlu diberlakukan normalisasi sehingga memiliki daya rata-rata bernilai satu. Hal ini juga memudahkan pengaruh pemberian IBO pada sinyal yang masuk. Untuk melakukan normalisasi pada sinyal sebelum

memasuki *predistorter* dan HPA, perlu dicari daya rata-rata sinyal yang masuk. Selanjutnya sinyal tersebut akan dibagi dengan akar dari daya rata-rata yang telah dicari sebelumnya. Untuk pemberian IBO pada sinyal yang telah di normalisasi, sinyal hanya perlu dikalikan dengan nilai IBO linier.

3.2.1.9 High Power Amplifier

Setelah melalui *predistorter*, sinyal akan memasuki HPA. HPA yang digunakan pada Tugas Akhir ini adalah *high power amplifier* model Saleh. Prosesnya sama dengan saat sinyal memasuki *predistorter* yakni sinyal akan dibagi menjadi dua yakni amplitudo sinyal dan fasa sinyal yang masing-masing akan diproses dengan fungsi AM/AM dan AM/PM. Gambar 3.14 menunjukkan hasil perbandingan antara penggunaan HPA saja dan penggunaan HPA yang didahului *predistorter*.



Gambar 3.15 Perbandingan Output Sinyal

Parameter yang digunakan untuk fungsi HPA model Saleh adalah $\alpha_a = 2.1587$, $\beta_a = 1.1517$, $\alpha_\phi = 4.033$, $\beta_\phi = 9.1040$. Sedangkan kanal yang digunakan adalah kanal ideal.

Berikut sintak program untuk high power amplifier:

```
xx=rr; %Sinyal Input HPA
teta=angle(xx); %Fasa sinyal input
Rt=abs(xx); %Amplitudo sinyal input
alfa_p=2.1587;beta_p=1.1517;alfa_q=4.033;beta_
q=9.104;
```

```

PR=(alfa_p.*Rt)./(1+beta_p.*Rt.^2);
QR=(alfa_q.*Rt.^2)./(1+beta_q.*Rt.^2);
sI=PR.*cos(QR); sQ=PR.*sin(QR);
yy=(sI+li*sQ).*exp(li*teta); %Sinyal Output
HPA
yyy=yy*Ex/sqrt(kk(1,k)); %Sinyal Output
setelah normalisasi

```

3.2.2 Bagian Receiver

Pada sisi penerima terdiri dari proses *remove cyclic prefix*, *Fast Fourier Transform* (FFT), *demapping*, hingga sinyal diterima kedalam bentuk *bit* seperti saat dibangkitkan di sisi *transmitter*. Parameter yang digunakan untuk melihat apakah terjadi kesalahan pada *bit* yang diterima dinamakan *bit error rate*.

3.2.2.1 Serial to Parallel dan Remove Cyclic Prefix

Langkah pertama yang perlu dilakukan untuk mengolah sinyal yang diterima di *receiver* adalah dengan mengembalikan sinyal yang sebelumnya di normalisasi dan kemudian dilanjutkan dengan menghilangkan *cyclic prefix* pada sinyal. Selanjutnya sinyal atau blok GFDM perlu mengubah matriks sinyal dari serial menjadi paralel yaitu ukuran 1x100000 menjadi 100x1000 dengan menggunakan fungsi *reshape*. Kemudian dengan menghilangkan dua puluh baris pertama sehingga tersisa matriks dengan ukuran 80x1000 yang berisi data informasi.

Berikut sintak program untuk *remove cyclic prefix*:

```

yr=reshape(yyy,size(xcpOQAM));
yrr=yrr(min(size(x_cpOQAM))+1:min(size(yr)),:);

```

3.2.2.2 FFT (Fast Fourier Transform)

Langkah kedua adalah demodulasi sinyal dengan melakukan proses FFT pada sinyal. Demodulasi pada sistem GFDM disebut juga dengan demodulasi *multicarrier*. Proses demodulasi ini bertujuan untuk memisahkan data informasi dengan sinyal *carrier*. Dalam proses FFT, sinyal yang semula berada di domain waktu diubah kembali ke domain frekuensi. Proses FFT pada sistem GFDM dapat dilakukan dengan mengalikan transpose dari matriks A_i dan A_q hasil IFFT dengan sinyal yang ditransmisikan setelah pengurangan CP. Persamaan dibawah ini menyatakan proses FFT.

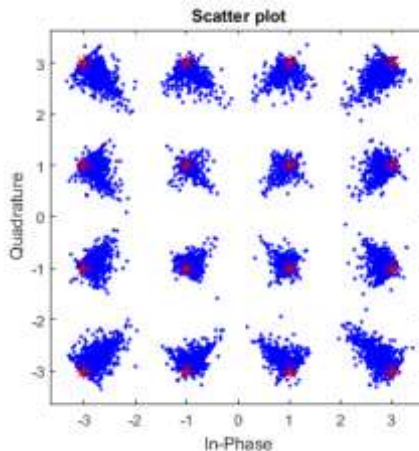
$$\hat{\mathbf{d}} = \Re\{(\tilde{A}^{(i)})^H \mathbf{y}\} + j\Re\{(\tilde{A}^{(q)})^H \mathbf{y}\} \quad 3.3$$

Berikut sintak program untuk Proses FFT:

```
diiE=(Ai) '*yrr;diE=(real(diiE));  
dqQE=(AqQAM) '*yrr;dqE=(real(dqQE));  
dE=diE+j*dqE;  
dE=reshape(dE,K,[]);  
dEc=circshift(dE,1);  
ddE=dEc(1:(K-3*K/8),:);  
for k2=1:jsimbol  
dddE(:,(k2-1)*(M-1)+(1:(M-1)))=ddE(:,(k2-  
1)*M+(2:M));  
end
```

3.2.2.3 Parallel to Serial

Setelah sinyal melalui proses FFT maka sinyal berbentuk paralel, namun tidak semua baris pada tahap ini dipakai untuk diproses lebih lanjut. Pada Tugas Akhir ini simulasi menggunakan lima *subcarrier* dan sembilan *subsymbol*, sehingga ukuran matriks yang digunakan 5x9000. Berikut adalah salah satu contoh simbol-simbol diterima setelah melalui proses FFT yang ditunjukkan pada Gambar 3.16 dengan menggunakan perintah *scatterplot*.



Gambar 3.16 Scatter plot simbol diterima pada IBO 5 dB

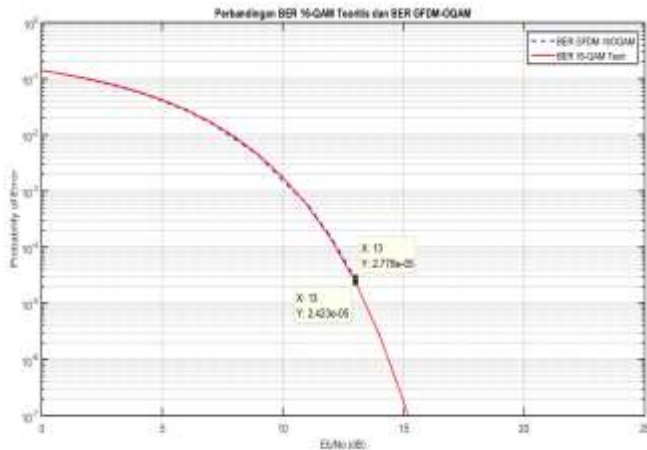
Kemudian matriks diubah menjadi serial untuk memasuki proses demodulasi. Pada tahap ini seperti halnya pada pemancar, *parallel to*

3.3 Validasi Program Simulasi

Hal yang perlu diperhatikan dalam melakukan simulasi suatu sistem adalah validasi hasil simulasi program yang telah dijalankan. Dalam Tugas Akhir ini akan diperhatikan sejumlah parameter yang akan menjadi perhatian dalam melakukan validasi yakni diantaranya: nilai BER GFDM, grafik PSD, *Gain*, dan EVM.

3.3.1 BER Sistem GFDM

Untuk mengecek ketepatan program simulasi GFDM pada simulasi, akan dilakukan validasi dengan membandingkan nilai *bit error rate* antara sistem GFDM-16 OQAM dengan fungsi teoritis dari modulasi 16-QAM. Gambar 3.18 memperlihatkan grafik nilai BER terhadap nilai E_b/N_0 . Dapat diperhatikan untuk nilai $E_b/N_0 = 13$ dB, nilai BER untuk sistem GFDM-16 OQAM adalah $2,778 \times 10^{-5}$ dan nilai BER untuk modulasi 16-QAM teoritis adalah $2,423 \times 10^{-5}$. Dapat disimpulkan bahwa simulasi program GFDM dengan modulasi 16-OQAM telah berjalan dengan baik dikarenakan memiliki kurva BER mendekati modulasi 16-QAM teoritis.

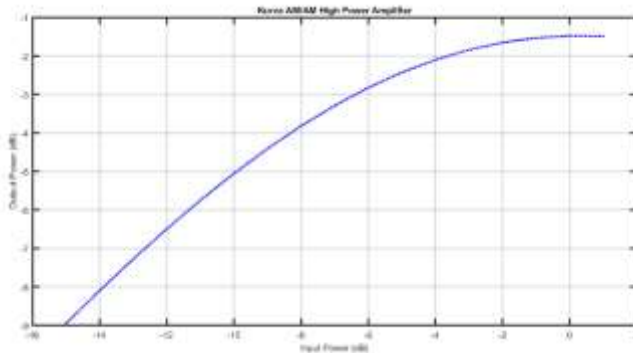


Gambar 3.18 Perbandingan nilai BER pada $E_b/N_0 = 13$ dB

3.3.2 Karakteristik AM/AM pada *High Power Amplifier*

Untuk mengecek ketepatan model HPA yang digunakan pada simulasi, selanjutnya akan dilakukan validasi dengan memperhatikan kurva AM/AM pada sistem GFDM-OQAM dengan HPA. Gambar 3.19

menunjukkan kurva karakteristik AM/AM dari HPA yang sesuai dengan kurva AM/AM HPA pada umumnya.



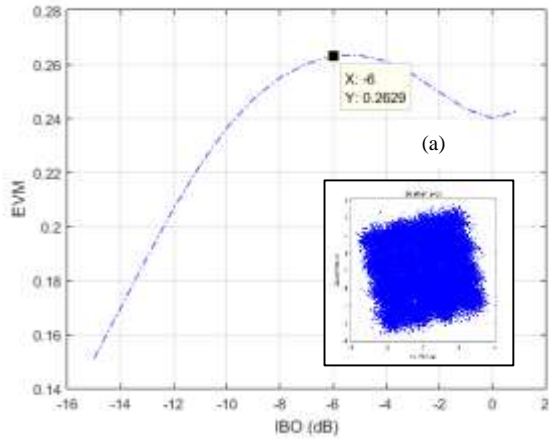
Gambar 3.19 Kurva AM/AM pada Sistem GFDM-OQAM

3.3.3 Error Vector Magnitude

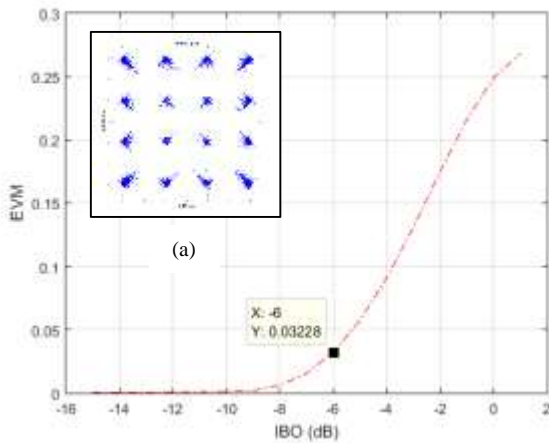
Untuk mengecek ketepatan model *predistorter* yang digunakan pada simulasi, validasi selanjutnya adalah dengan mengamati nilai *error vector magnitude* sebelum dan sesudah penggunaan *predistorter*. Nilai EVM dapat dihitung dengan mengolah simbol yang diterima pada *receiver* dengan simbol ideal. Persamaan 3.4 menyatakan EVM.

$$EVM = \sqrt{\frac{E\{|z(t)-x(t)|^2\}}{E\{|x(t)|^2\}}} \quad 3.4$$

Dengan menggunakan persamaan diatas, akan didapatkan kurva EVM terhadap IBO sebelum dan sesudah penerapan *predistorter*. Gambar 3.20 dan Gambar 3.21 menunjukkan kurva EVM terhadap nilai IBO. Untuk sistem GFDM-OQAM menggunakan HPA saja, khususnya pada nilai IBO 6 dB EVM nya bernilai 0.2629 atau 26.29%. Sedangkan pada sistem GFDM-OQAM menggunakan HPA dan *predistorter*, khususnya pada nilai IBO 6 dB EVM nya bernilai 0.03228 atau 3,228%. Dengan menggunakan *predistorter* maka konstelasi dari sinyal yang diterima mengalami perbaikan dari semula memiliki kesalahan sebesar 26% turun menjadi 3%. Dapat diartikan bahwa *predistorter* yang digunakan dalam simulasi telah berjalan sesuai dengan tujuannya. Untuk melihat perbaikan pada konstelasi sinyal diterima dapat dilihat secara visual pada Gambar 3.20 (a) dan Gambar 3.21 (a). Dengan melihat secara visual dapat diperhatikan pengaruh dari penggunaan *predistorter* pada konstelasi sinyal yang diterima khususnya pada nilai IBO 6 dB.



Gambar 3.20 Kurva EVM terhadap IBO untuk Sistem GFDM-OQAM dengan HPA saja



Gambar 3.21 Kurva EVM terhadap IBO untuk Sistem GFDM-OQAM dengan HPA dan *Predistorter*

[Halaman Ini Sengaja Dikosongkan]

BAB 4

ANALISIS KINERJA SISTEM

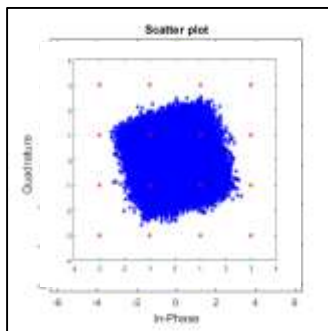
Pada bab ini akan ditampilkan serta analisis hasil simulasi dari sistem GFDM-OQAM dengan menggunakan HPA saja dan HPA + *Predistorter*. Modulasi 8-PSK akan digunakan sebagai pembanding dalam analisis. Hasil dari simulasi akan dianalisis untuk mengetahui dan membandingkan kinerja dari pengaplikasian *predistorter* dalam sistem nonlinier. Seluruh simulasi dilakukan dengan menggunakan Matlab versi R2015a (8.5.0.197613).

4.1 Analisis Hasil Simulasi Sistem GFDM dengan HPA

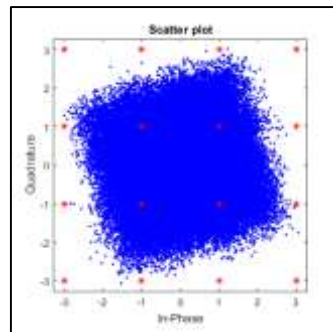
Pada sub bab ini akan dijelaskan mengenai kinerja sistem GFDM-OQAM dengan tambahan HPA. Tujuannya adalah untuk mengamati efek distorsi nonlinier yang diberikan oleh HPA terhadap sinyal yang diterima. Aspek-aspek yang diperhatikan antara lain konstelasi simbol di penerima dan nilai *error vector magnitude*. Penggunaan modulasi 8-PSK pada simulasi ini sebagai modulasi pembanding terhadap modulasi 16-OQAM.

4.1.1 Scatter Plot Simbol di Penerima

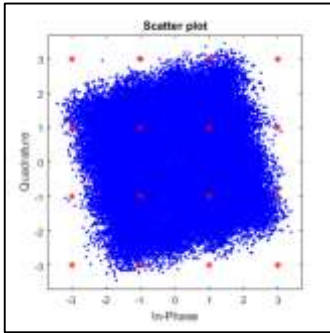
Untuk mengamati konstelasi simbol di penerima maka akan digunakan fungsi *scatterplot* pada MATLAB. Parameter yang digunakan dalam simulasi telah dijelaskan pada bab sebelumnya. *Scatter plot* bagian pertama yang ditampilkan menggunakan modulasi 16-OQAM dengan IBO 0, 2, 4, 6, 8, 10, 12, dan 15 dB pada Gambar 4.1. Untuk *scatter plot* bagian kedua yang ditampilkan hanya berdasarkan modulasi 8-PSK dengan IBO 0, 2, 4, 6, 8, 10, 12, dan 15 dB pada Gambar 4.2.



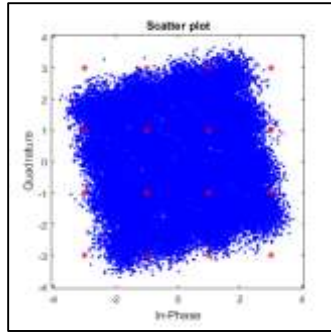
(a)



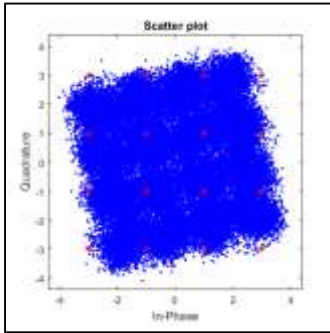
(b)



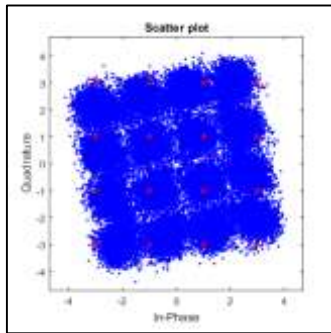
(c)



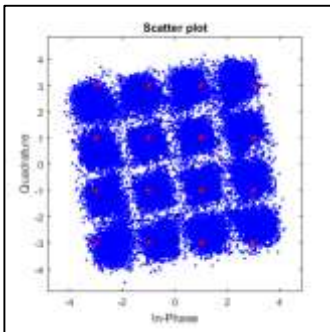
(d)



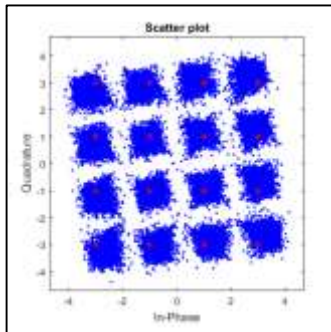
(e)



(f)



(g)



(h)

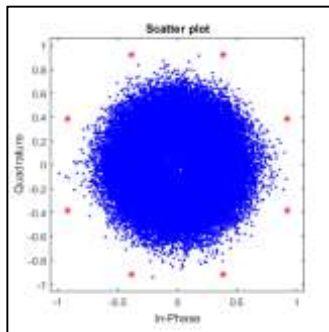
Gambar 4.1 Scatterplot Simbol 16-QAM GFDM dengan HPA untuk IBO: (a) 0 dB, (b) 2 dB, (c) 4 dB, (d) 6 dB, (e) 8 dB, (f) 10 dB, (g) 12 dB, dan (h) 15 dB

Dari hasil simulasi sistem GFDM-OQAM dengan HPA, dapat diperhatikan bahwa dengan menaikkan nilai IBO maka *scatter plot* yang menampilkan konstelasi sinyal diterima mengalami perbaikan. Plot berwarna biru adalah simbol-simbol yang diterima sedangkan plot berwarna merah adalah simbol-simbol yang dikirimkan atau konstelasi referensi. Saat IBO bernilai 0 dB, simbol-simbol yang diterima mengalami kompresi sehingga jatuh pada radius satu. Saat IBO bernilai 2 dB hingga 15 dB, simbol-simbol yang diterima mengalami pengembangan sehingga simbol-simbol tersebut jatuh di titik referensi masing-masing. Nilai rata-rata amplitudo terukur dan fasa terukur dapat dilihat pada Tabel 4.1 yang menunjukkan perbaikan nilai seiring dengan kenaikan nilai IBO.

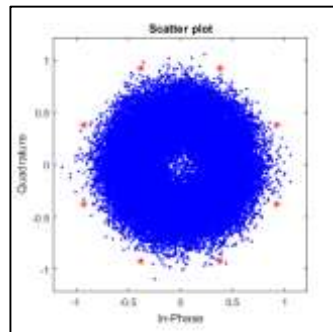
Tabel 4.1 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-OQAM dengan HPA saja

IBO (dB)	Σ Amplitudo Terukur	Σ Amplitudo Referensi	Σ Fasa Terukur	Σ Fasa Referensi
0	1.1051	2.9997	14.5138°	0°
2	1.4111		13.8521°	
4	1.7336		12.9963°	
6	2.0495		11.9235°	
8	2.3373		10.6336°	
10	2.5819		9.1627°	
12	2.7770		7.5886°	
15	2.9816		5.2739°	

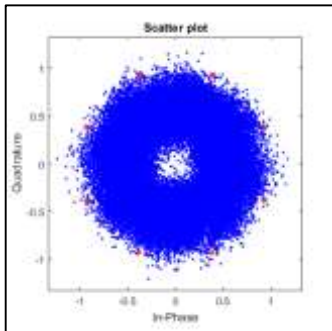
Dapat disimpulkan bahwa dengan menaikkan nilai IBO maka kesalahan pada sinyal akan mengalami perbaikan yang ditandai dengan nilai rata-rata amplitudo mendekati nilai amplitudo referensi dan nilai rata-rata fasa terukur mendekati nilai fasa referensi.



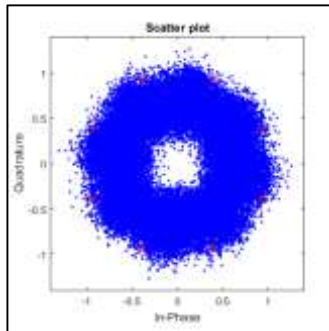
(a)



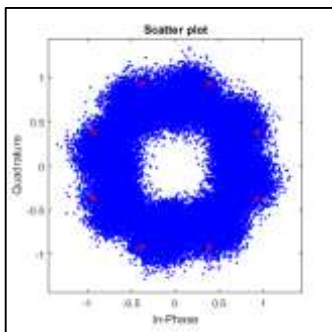
(b)



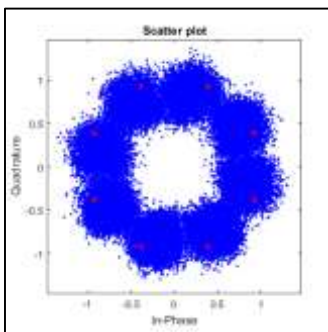
(c)



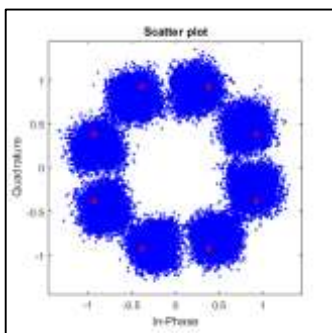
(d)



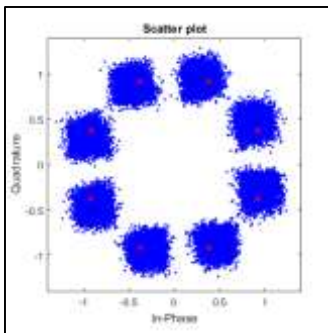
(e)



(f)



(g)



(h)

Gambar 4.2 Scatterplot Simbol 8-PSK GFDM dengan HPA IBO: (a) 0 dB, (b) 2 dB, (c) 4 dB, (d) 6 dB, (e) 8 dB, (f) 10 dB, (g) 12 dB, dan (h) 15 dB

Selanjutnya Gambar 4.2 menunjukkan hasil *scatter plot* dari sistem GFDM-8 PSK dengan HPA dapat diperhatikan bahwa dengan menaikkan nilai IBO maka konstelasi dari sinyal yang diterima jatuh pada titik referensi. Sama seperti saat menggunakan modulasi 16-OQAM, ketika nilai IBO 0 dB maka simbol diterima jatuh pada radius 0.5 sehingga antara simbol yang berbeda tidak terdapat ruang bebas. Hal ini dapat mempengaruhi nilai BER yang terukur. Saat nilai IBO 2 dB hingga 15 dB antar simbol yang berbeda mulai muncul jarak atau dengan kata lain kompresi dari HPA mulai berkurang sehingga simbol-simbol tersebut jatuh mendekati titik referensi. Sedangkan untuk fasa pada modulasi ini memiliki nilai terkecil yakni 5.4356° saat IBO bernilai 5 dB.

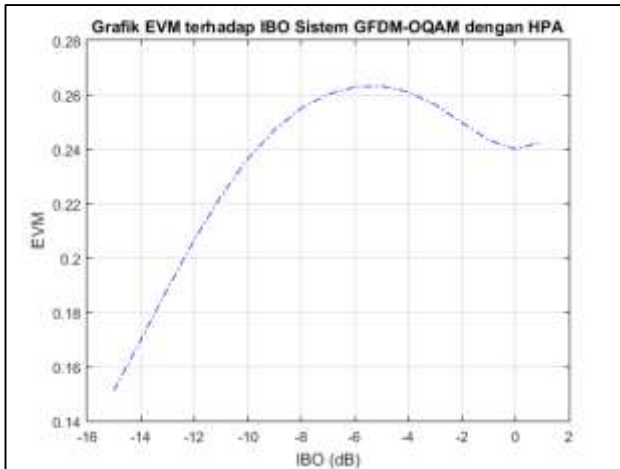
Tabel 4.2 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-PSK dengan HPA saja

IBO (dB)	Σ Amplitudo Terukur	Σ Amplitudo Referensi	Σ Fasa Terukur	Σ Fasa Referensi
0	0.3364	1	14.5354°	0°
2	0.4689		13.9123°	
4	0.5772		13.0936°	
6	0.6834		12.0550°	
8	0.7802		10.7935°	
10	0.8624		9.3408°	
12	0.9278		7.7709°	
15	0.9962		5.4356°	

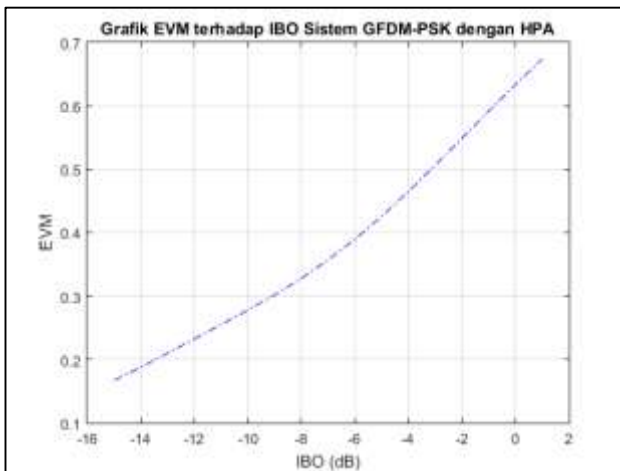
Dapat disimpulkan pengaruh penambahan nilai IBO pada modulasi 8-PSK dimana untuk amplitudo mengalami peningkatan sedangkan pergeseran fasa mengalami penurunan walaupun rata-rata fasa masih diatas 0° . Dari kedua modulasi tersebut dapat dikatakan bahwa penggunaan modulasi memiliki pengaruh yang kecil terhadap perbaikan menggunakan IBO dalam mengatasi distorsi nonlinear khususnya distorsi amplitudo dan distorsi fasa.

4.1.2 *Error Vector Magnitude Sinyal di Penerima*

Tingkat distorsi pada sinyal diterima dapat diketahui dengan menggunakan parameter EVM. Gambar 4.3 akan menampilkan nilai EVM terhadap nilai IBO rentang 0-15 dB pada penggunaan modulasi 16-OQAM. Gambar 4.4 akan menampilkan nilai EVM terhadap nilai IBO 0-15 dB pada penggunaan modulasi 8-PSK. Perlu diketahui bahwa EVM dapat digunakan sebagai indikator numerik pengaruh distorsi terhadap sinyal yang diterima. Sedangkan dengan melihat konstelasi kita dapat mengetahui pengaruh distorsi pada sinyal secara visual.



Gambar 4.3 Grafik Perbandingan EVM terhadap IBO pada Sistem GFDM-OQAM dengan HPA



Gambar 4.4 Grafik Perbandingan EVM terhadap IBO pada Sistem GFDM-PSK dengan HPA

Nilai EVM pada kedua modulasi mengalami penurunan nilai seiring dengan kenaikan nilai IBO. Penurunan nilai ini dapat dikaitkan dengan perbaikan konstelasi sinyal diterima yang dapat diamati pada

Gambar 4.1 dan 4.2. Sama seperti bagian sebelumnya, penggunaan modulasi 16-OQAM dan 8-PSK memiliki pengaruh yang kecil terhadap nilai EVM seiring dengan kenaikan nilai IBO.

Tabel 4.3 Nilai EVM terhadap IBO pada sistem GFDM-OQAM dan GFDM-PSK dengan HPA untuk $\alpha = 0.5$

IBO (dB)	EVM	
	16-OQAM	8-PSK
0	0.2400	0.6336
1	0.2433	0.5916
2	0.2497	0.5488
3	0.2563	0.5062
4	0.2611	0.4648
5	0.2632	0.4257
6	0.2629	0.3896
7	0.2602	0.3570
8	0.2550	0.3279
9	0.2471	0.3018
10	0.2364	0.2779
11	0.2227	0.2550
12	0.2066	0.2323
13	0.1886	0.2097
14	0.1696	0.1877
15	0.1511	0.1670

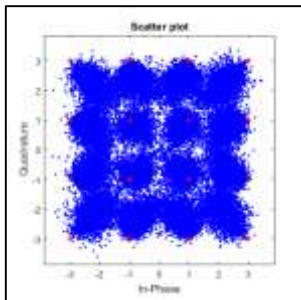
4.2 Analisis Hasil Simulasi Sistem GFDM dengan HPA dan *Predistorter*

Pada sub bab ini akan ditampilkan hasil simulasi sistem GFDM dengan HPA dan *predistorter*. Tujuannya adalah untuk mengamati pengaruh aplikasi *predistorter* dalam sistem terhadap sinyal yang diterima khususnya terhadap distorsi nonlinier yang muncul dikarenakan penggunaan HPA. Aspek-aspek yang diperhatikan antara lain konstelasi simbol di penerima dan nilai *error vector magnitude*. Penggunaan modulasi 8-PSK pada simulasi ini sebagai modulasi pembanding terhadap modulasi 16-OQAM.

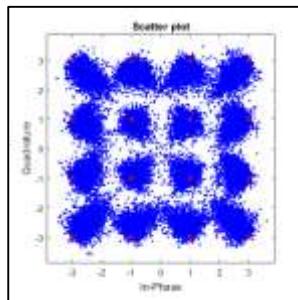
4.2.1 *Scatter Plot* Simbol di Penerima

Untuk mengamati konstelasi dari sinyal yang diterima maka dapat digunakan fungsi *scatterplot* pada MATLAB. Parameter yang digunakan telah dijelaskan pada bab sebelumnya. *Scatter plot* bagian

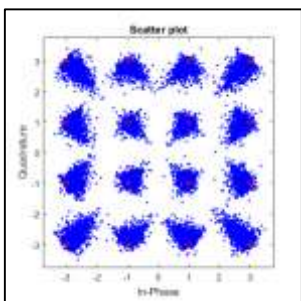
pertama akan menampilkan hasil dari modulasi 16-QAM dengan nilai IBO 0 dB, 2 dB, 4 dB, 6 dB, 8 dB, 10 dB, 12 dB, dan 15 dB pada Gambar 4.5. Sedangkan bagian kedua akan menampilkan hasil dari modulasi 8-PSK dengan nilai IBO 0 dB, 2 dB, 4 dB, 6 dB, 8 dB, 10 dB, 12 dB, dan 15 dB pada Gambar 4.6.



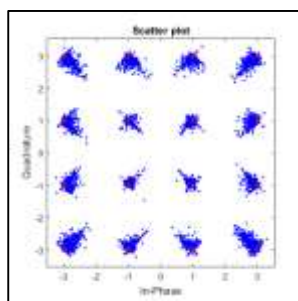
(a)



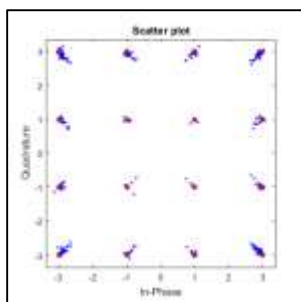
(b)



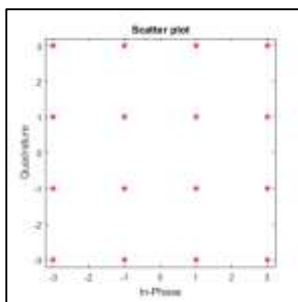
(c)



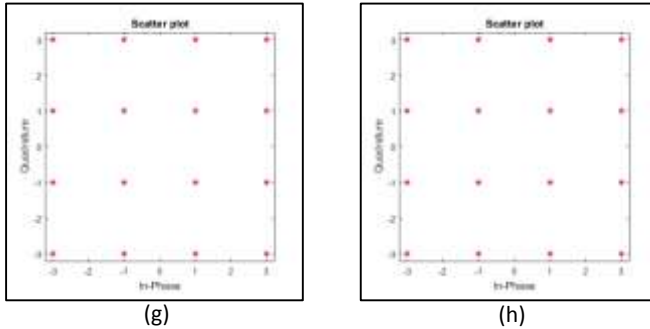
(d)



(e)



(f)



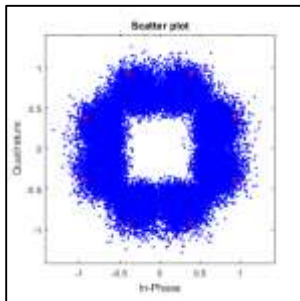
Gambar 4.5 Scatterplot Simbol 16-QAM GFDM dengan HPA dan *Predistorter* HPA untuk IBO: (a) 0 dB, (b) 2 dB, (c) 4 dB, (d) 6 dB, (e) 8 dB, (f) 10 dB, (g) 12 dB, dan (h) 15 dB

Dari hasil simulasi sistem GFDM dengan HPA dan *Predistorter* yang menggunakan modulasi 16-QAM dapat diperhatikan bahwa seiring dengan kenaikan IBO, simbol-simbol yang diterima mengalami perbaikan dimana mereka berada pada lokasi simbol-simbol referensi atau sinyal yang dikirimkan. Plot berwarna biru adalah sinyal yang diterima, sedangkan plot berwarna merah adalah sinyal yang dikirimkan. Dapat diperhatikan saat IBO bernilai 0 dB, persebaran masing-masing simbol berada diantara titik referensi dan titik pusat. Pola ini berbeda dengan saat menggunakan HPA saja. Saat IBO bernilai 2 dB hingga 15 dB, persebaran dari masing-masing simbol mengalami penurunan sehingga banyak yang jatuh tepat pada titik referensi. Nilai rata-rata amplitudo terukur dan fasa terukur dapat dilihat pada Tabel 4.4 yang menunjukkan perbaikan nilai seiring dengan kenaikan nilai IBO.

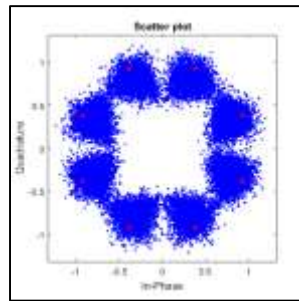
Tabel 4.4 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-OQAM dengan HPA dan *predistorter*

IBO (dB)	\sum Amplitudo Terukur	\sum Amplitudo Referensi	\sum Fasa Terukur	\sum Fasa Referensi
0	2.2927	2.9997	0.2585°	0°
2	2.6177		0.1433°	
4	2.8564		0.1652°	
6	2.9724		0.1616°	
8	2.9983		0.1517°	
10	2.9997		0.1384°	
12	2.9997		0.1283°	
15	2.9997		0.1178°	

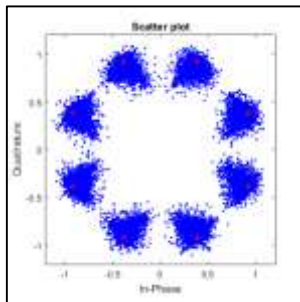
Dari hasil diatas dapat ditarik kesimpulan bahwa dengan menggunakan *predistorter*, distorsi nonlinier khususnya distorsi amplitudo dan distorsi fasa dapat dikurangi seiring dengan kenaikan nilai IBO.



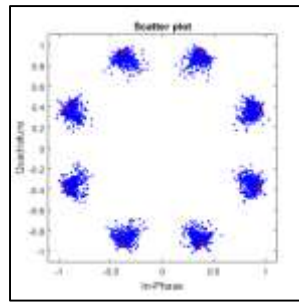
(a)



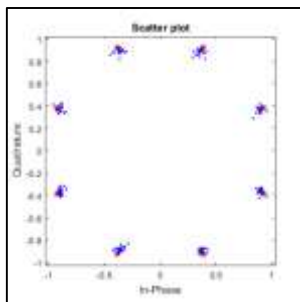
(b)



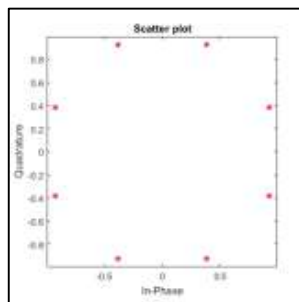
(c)



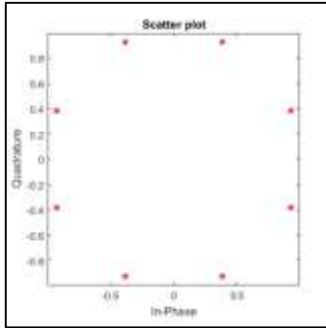
(d)



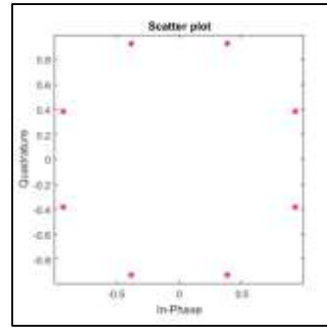
(e)



(f)



(g)



(h)

Gambar 4.6 Scatterplot Simbol 8-PSK GFDM dengan HPA dan *Predistorter* untuk IBO (a) 0 dB (b) 4 dB (c) 8 dB (d) 12 dB

Selanjutnya pada untuk penggunaan modulasi 8-PSK pada sistem GFDM menggunakan HPA dan *predistorter*, dapat diperhatikan bahwa seiring dengan kenaikan nilai IBO maka simbol-simbol yang diterima mengalami perbaikan. Untuk nilai IBO 0 dB, dapat diperhatikan bahwa persebaran dari masing-masing simbol yang berbeda masih saling berpotongan. Sedangkan untuk nilai IBO 2 dB hingga 15 dB dapat diperhatikan bahwa simbol-simbol yang diterima memiliki persebaran yang terpusat pada titik referensi. Nilai rata-rata amplitudo terukur dan fasa terukur dapat dilihat pada Tabel 4.5 yang menunjukkan perbaikan nilai seiring dengan kenaikan nilai IBO.

Tabel 4.5 Nilai rata-rata amplitudo dan fasa sinyal diterima sistem GFDM-OQAM dengan HPA dan *predistorter*

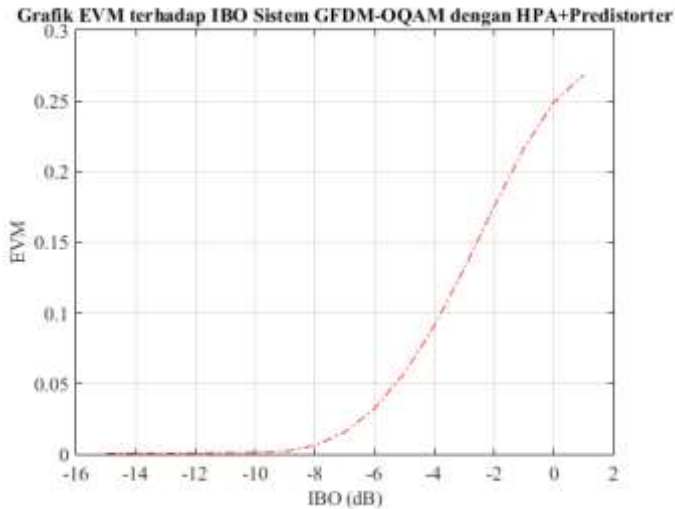
IBO (dB)	Σ Amplitudo Terukur	Σ Amplitudo Referensi	Σ Fasa Terukur	Σ Fasa Referensi
0	0.7676	1	0.4916°	0°
2	0.8758		0.5728°	
4	0.9546		0.5279°	
6	0.9924		0.5165°	
8	0.9998		0.5032°	
10	1		0.4898°	
12	1		0.4794°	
15	1		0.4686°	

Dari Tabel 4.5 dapat dikatakan bahwa seiring dengan kenaikan IBO maka rata-rata amplitudo mendekati nilai ideal yakni 1.

Dapat disimpulkan bahwa penggunaan modulasi sedikit memengaruhi kinerja dari *predistorter* dalam mengatasi distorsi nonlinier khususnya distorsi amplitudo dan distorsi fasa.

4.2.2 Error Vector Magnitude Sinyal di Penerima

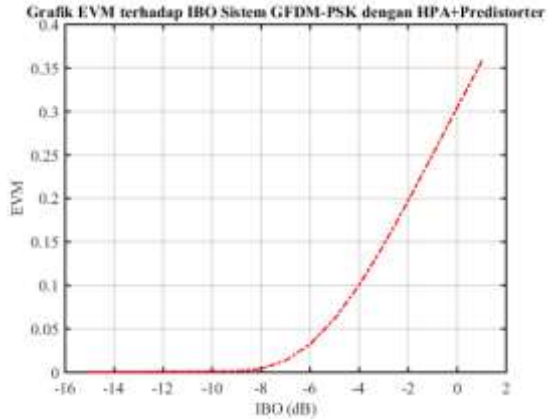
Untuk mengetahui tingkat distorsi pada sinyal yang diterima dapat menggunakan parameter EVM. Gambar 4.7 akan menampilkan nilai EVM terhadap IBO pada penggunaan modulasi 16-OQAM.



Gambar 4.7 Grafik Perbandingan EVM terhadap IBO pada sistem GFDM-OQAM dengan HPA dan *Predistorter*

Sedangkan gambar 4.8 akan menampilkan nilai EVM terhadap IBO pada penggunaan modulasi 8-OPSK.

Nilai EVM pada kedua modulasi mengalami penurunan seiring dengan kenaikan IBO terkait dengan rata-rata daya sinyal referensi atau sinyal yang dikirimkan. Seperti terlihat jelas pada gambar 4.5 (d) dan 4.6 (d), simbol yang diterima jatuh tepat pada konstelasi sinyal yang dikirimkan sehingga memengaruhi nilai EVM. Kedua modulasi memiliki nilai EVM kurang dari 0.05 atau 0.5 % pada IBO 6 dB sehingga dapat dikatakan bahwa pada nilai IBO ini masih terdapat distorsi residu namun relatif rendah dibandingkan saat tidak menggunakan *predistorter*. Tabel 4.6 berisikan perbandingan nilai EVM dari kedua modulasi pada sistem GFDM-OQAM yang menggunakan *predistorter* dan HPA.



Gambar 4.8 Grafik Perbandingan EVM terhadap IBO pada sistem GFDM-PSK dengan HPA dan *Predistorter*

Tabel 4.6 Nilai EVM terhadap IBO pada sistem GFDM-OQAM dan GFDM-PSK dengan HPA dan *Predistorter* untuk $\alpha = 0.5$

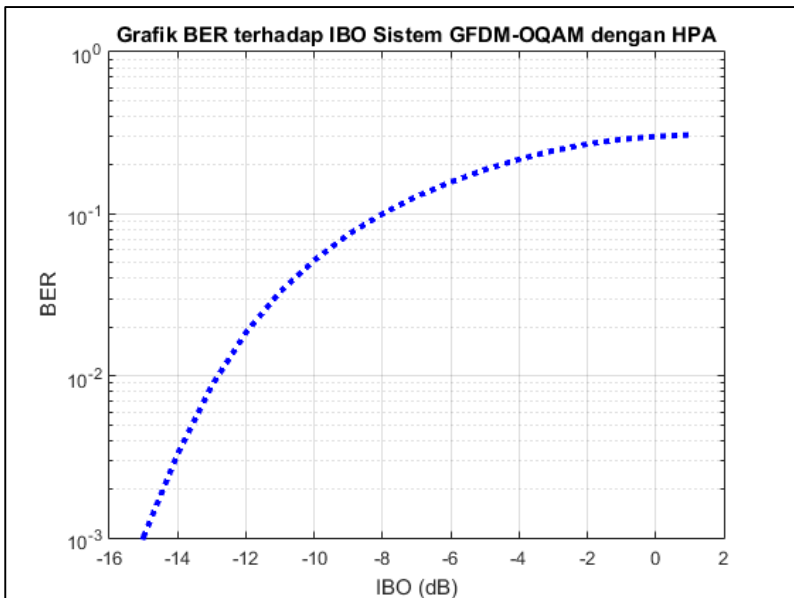
IBO (dB)	<i>Error Vector Magnitude</i>	
	16-OQAM	8-PSK
0	0.2490	0.3045
1	0.2164	0.2506
2	0.1751	0.1970
3	0.1314	0.1462
4	0.0910	0.1003
5	0.0574	0.0617
6	0.0323	0.0324
7	0.0155	0.0135
8	0.0061	0.0042
9	0.0020	0.0013
10	9.6714×10^{-4}	0.0011
11	8.1817×10^{-4}	8.9907×10^{-4}
12	6.8648×10^{-4}	7.5491×10^{-4}
13	5.7152×10^{-4}	6.2890×10^{-4}
14	4.7244×10^{-4}	5.2020×10^{-4}
15	3.8805×10^{-4}	4.2761×10^{-4}

4.3 Analisis Perbandingan Hasil Simulasi Sistem GFDM dengan HPA Sebelum dan Sesudah Aplikasi *Predistorter*

Pada sub bab ini akan ditampilkan hasil simulasi yang menunjukkan sebelum dan sesudah penggunaan *predistorter* dalam sistem GFDM yang menggunakan HPA. Tujuannya adalah untuk mengamati aplikasi *predistorter* dalam mengatasi distorsi nonlinier dilihat dari parameter *bit error rate*.

4.3.1 Perbandingan *Bit Error Rate* Sebelum dan Sesudah Aplikasi *Predistorter*

Pada bagian ini akan menampilkan probabilitas bit yang diterima salah. Caranya adalah dengan menghitung BER pada *bit-bit* yang diterima. Penghitungan BER ini dilakukan untuk sistem GFDM dengan modulasi 16-QAM serta dengan tambahan HPA dan HPA+*predistorter*. *Roll-off-factor* yang digunakan adalah 0.5. Gambar 4.9 menunjukkan kurva BER terhadap nilai IBO dari sistem GFDM-OQAM.



Gambar 4.9 Grafik perbandingan BER sistem GFDM-OQAM dengan HPA terhadap IBO

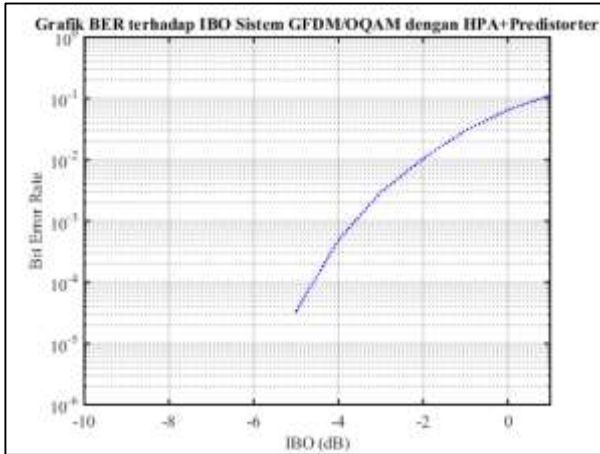
Hasil yang didapatkan adalah untuk GFDM-OQAM dengan *roll-of-factor* 0.5 didapatkan nilai BER sekitar 0.3 untuk nilai IBO mulai dari 0 dB hingga nilai BER sekitar 0.01 untuk 15 dB. Artinya adalah saat IBO bernilai 0 dB, maka dari 1000 bit yang dikirimkan terdapat 300 bit yang salah saat diterima. Sedangkan saat IBO bernilai 15 dB, maka dari 1000 bit yang dikirimkan terdapat 1 bit yang salah saat diterima. Hal ini menunjukkan dengan bertambahnya IBO maka kinerja dari sistem mengalami peningkatan yang ditandai dengan menurunnya nilai BER. Tabel 4.7 menunjukkan penurunan nilai BER seiring dengan kenaikan nilai IBO.

Tabel 4.7 Nilai BER terhadap IBO pada sistem GFDM-OQAM dengan HPA untuk $\alpha = 0.5$

IBO (dB)	BER
0	0.3003
1	0.2885
2	0.2700
3	0.2457
4	0.2171
5	0.1875
6	0.1572
7	0.1283
8	0.1001
9	0.0747
10	0.0517
11	0.0330
12	0.0185
13	0.0087
14	0.0033
15	0.0010

Selanjutnya adalah hasil penghitungan BER untuk sistem GFDM-OQAM dengan HPA+*predistorter*. Gambar 4.10 menampilkan kurva BER terhadap kenaikan nilai IBO dari sistem GFDM-OQAM dengan HPA+*predistorter*. Hasil yang didapat adalah setelah *predistorter* diterapkan pada sistem didapatkan nilai BER sekitar 0.06 untuk IBO bernilai 0 dB hingga 3.3333×10^{-5} untuk IBO bernilai 5 dB. Artinya adalah saat IBO bernilai 0 dB, maka dari 100000 bit yang dikirimkan terdapat 6000 bit yang salah saat diterima. Sedangkan saat IBO bernilai 15 dB,

maka dari 100000 bit yang dikirimkan terdapat 3 bit yang salah saat diterima.



Gambar 4.10 Grafik perbandingan BER sistem GFDM-OQAM dengan HPA dan *Predistorter* terhadap IBO

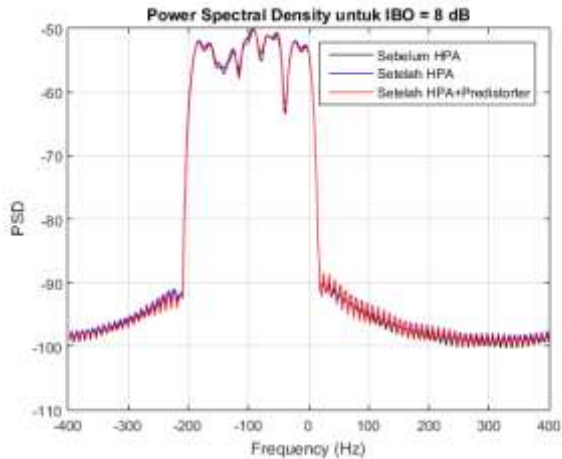
Tabel 4.8 Nilai BER terhadap IBO pada sistem GFDM-OQAM dengan *Predistorter* dan HPA untuk $\alpha = 0.5$

IBO (dB)	BER
0	0.0650
1	0.0299
2	0.0105
3	0.0029
4	4.8889x10 ⁻⁴
5	3.3333x10 ⁻⁵
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0

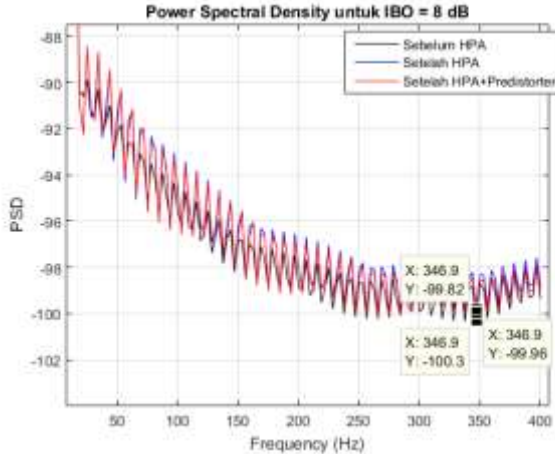
Kemudian untuk mendapatkan kinerja yang sama dalam hal ini nilai BER 0.01, tanpa menggunakan *predistorter* dibutuhkan IBO sebesar 15 dB sedangkan dengan menggunakan *predistorter* hanya dibutuhkan IBO sebesar 3 dB. Dari kedua perbandingan diatas dapat disimpulkan bahwa *predistorter* mampu memitigasi efek distorsi nonlinier pada sinyal yang dikirimkan.

4.4 Analisis *Power Spectral Density* Sistem GFDM Sebelum dan Sesudah Aplikasi *Predistorter*

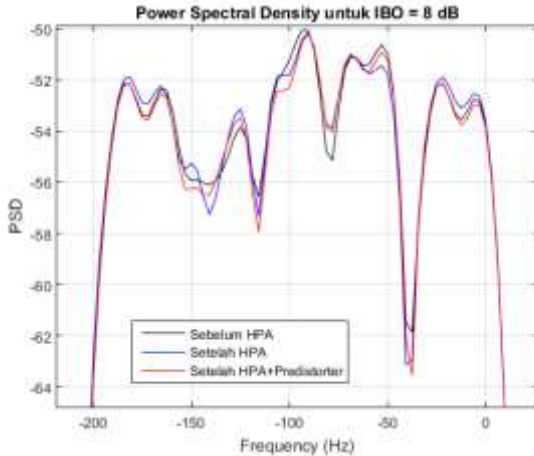
Pada sub bab ini akan menampilkan hasil spektrum sinyal pada sistem GFDM baik sebelum maupun sesudah penggunaan *predistorter*. Tujuannya adalah untuk mengamati pengaruh dari penggunaan *predistorter* pada sistem GFDM-OQAM dengan HPA. Untuk menghasilkan grafik PSD pada Matlab digunakan fungsi *pwelch* yang menggunakan metode Welch untuk mengestimasi PSD dari suatu sinyal. Hasil grafik *Power Spectral Density* pada Sistem GFDM dapat dilihat pada gambar 4.14, 4.15, dan 4.16.



Gambar 4.11 Grafik *Power Spectral Density* pada sistem GFDM-OQAM dengan HPA dan Predistorter+HPA



Gambar 4.12 Grafik *Power Spectral Density* pada sistem GFDM-OQAM dengan HPA dan HPA+predistorter (*side lobe*)



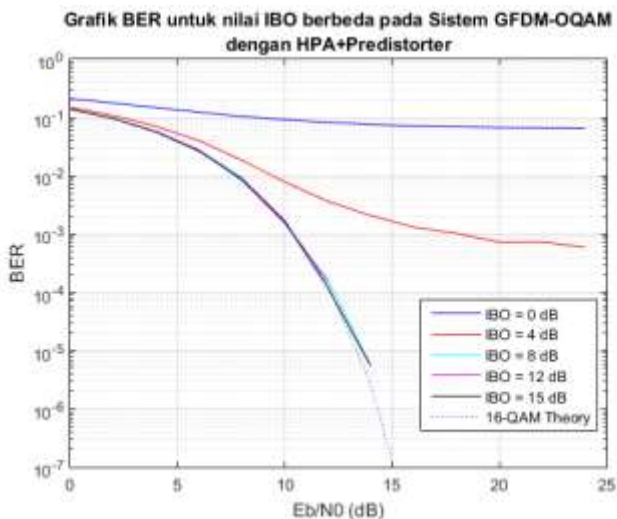
Gambar 4.13 Grafik *Power Spectral Density* pada sistem GFDM-OQAM dengan HPA dan HPA+predistorter (*main lobe*)

Dapat diperhatikan bahwa dengan penggunaan *predistorter* pun, *side lobe* PSD dari sinyal GFDM setelah dilewatkan HPA dan

predistorter memiliki nilai yang mendekati PSD sinyal sebelum dilewatkan HPA dan *predistorter*. Sedangkan untuk *main lobe* dapat dilihat bahwa PSD dari sinyal yang dilewatkan HPA dan *predistorter* memiliki nilai relatif lebih baik dibandingkan dengan sinyal sebelum dilewatkan HPA dan *predistorter*.

4.5 Analisis Perbandingan Hasil Simulasi Sistem GFDM dengan HPA Sesudah Aplikasi *Predistorter* Pada Kanal AWGN

Pada sub bab ini akan ditampilkan hasil simulasi yang menunjukkan penggunaan *predistorter* pada sistem GFDM-OQAM yang menggunakan HPA. Namun pada sub bab ini skema yang digunakan adalah skema kedua dimana sinyal yang melalui kanal ideal akan diberikan *noise* AWGN. Tujuannya adalah untuk mengamati pengaruh penambahan *noise* terhadap kinerja *predistorter*, khususnya pada parameter *bit error rate*.



Gambar 4.14 Grafik perbandingan BER Sistem GFDM_OQAM dengan HPA dan *predistorter* pada kanal AWGN

Dari hasil simulasi didapatkan grafik perbandingan BER terhadap nilai E_b/N_0 pada sistem GFDM menggunakan *predistorter* dan HPA untuk nilai IBO 0 dB, 4 dB, 8 dB, 12 dB, dan 15 dB yang juga

dibandingkan dengan nilai BER teori 16-QAM. Hasil yang didapatkan dengan nilai *roll-off factor* 0.5 dapat dilihat pada Tabel 4.9.

Tabel 4.9 Nilai BER terhadap IBO pada sistem GFDM-OQAM dengan HPA untuk $\alpha = 0.5$

IBO	Eb/N0						
	0	4	8	12	16	20	24
0	0.2119	0.1487	0.1048	0.08305	0.0718	0.0677	0.0665
4	0.1504	0.0709	0.01855	0.0036	0.0013	0.00072	0.0006
8	0.1395	0.0571	0.00916	0.000167	0	0	0
12	0.1380	0.057467	0.00852	0.000156	5.56E-06	0	0
15	0.13972	0.0569	0.00873	0.00012	0	0	0

Dari tabel diatas dapat diperhatikan untuk nilai redaman atau IBO 0 dB kinerja BER nya mencapai nilai 10^{-1} . Untuk nilai redaman atau IBO 4 dB kinerja BER nya mencapai nilai 10^{-3} . Kemudian untuk nilai redaman atau IBO 8 dB hingga 15 dB kinerja BER nya mencapai nilai 10^{-5} . Sehingga dapat disimpulkan bahwa dengan meningkatkan redaman atau IBO kinerja BER dari sistem akan membaik.

BAB 5

PENUTUP

5.1 Kesimpulan

Dari pembahasan dan analisis berupa konstelasi simbol yang diterima, grafik BER terhadap IBO, dan grafik EVM terhadap IBO maka melalui penelitian tugas akhir ini dapat ditarik kesimpulan sebagai berikut:

1. Dengan menggunakan *predistorter*, residu dari distorsi amplitudo masih dapat terlihat pada rentang IBO 0 dB hingga 6 dB yang berupa persebaran tidak merata dari masing-masing titik simbol.
2. Sedangkan residu dari distorsi fasa tidak terlihat setelah penggunaan *predistorter*.
3. Kinerja BER mengalami peningkatan drastis setelah penggunaan *predistorter* dimana untuk mencapai kinerja BER 10^{-3} dibutuhkan IBO sebesar -15 dB untuk sistem tanpa *predistorter*, sedangkan dengan menggunakan *predistorter* hanya dibutuhkan IBO sebesar -3 dB. Hal ini menunjukkan bahwa dengan menggunakan *predistorter*, sistem dapat mengurangi IBO yang dibutuhkan untuk menekan *gain* dari HPA.
4. Sama halnya seperti BER, dengan menggunakan *predistorter* maka distorsi nonlinier dapat ditekan. Apabila sistem tanpa *predistorter* memerlukan IBO sebesar -15 dB untuk mencapai nilai EVM 15% maka dengan menggunakan *predistorter* nilai EVM 15% dapat dicapai dengan IBO sebesar -3 dB.
5. Sistem dapat memiliki kinerja BER 10^{-5} setelah aplikasi *predistorter* apabila menggunakan IBO sebesar 6 dB pada kanal ideal. Apabila *noise* AWGN ditambahkan pada sinyal yang dikirimkan maka dibutuhkan IBO sebesar 8 dB untuk mencapai kinerja BER yang sama.

5.2 Saran

Pada pengerjaan tugas akhir ini, saran yang dapat diberikan penulis untuk pengembangan penelitian lebih lanjut untuk topik GFDM/OQAM antara lain :

1. Penggunaan parameter model Saleh dapat divariasikan sehingga mampu menunjukkan variasi pengaruh efek nonlinear pada sinyal.
2. Efek nonlinearitas yang diberikan pada simulasi dapat ditambahkan efek memori.

DAFTAR PUSTAKA

- [1] N. Michailow, R. Datta, S. Krone, M. Lentmaier, and G. Fettweis, "Generalized frequency division multiplexing: A flexible multi-carrier modulation scheme for 5th generation cellular networks," in *Proceedings of the German microwave conference (GeMiC'12)*, 2012, pp. 1–4.
- [2] R. Ajeng Rochmatika, "Implementasi dan Analisis Teknik Reduksi PAPR OFDM Menggunakan Metode PTS pada WARP."
- [3] Bernard Sklar, *Digital communications: fundamentals and applications*, Prentice-Hall, Inc., Upper Saddle River, New Jersey, 1988
- [4] J. G. Proakis, M. Salehi, *Digital communications Fifth Edition*, Mc Graw-Hill Co, New York, 2008
- [5] Theodore Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall PTR, Upper Saddle River, NJ, 2001
- [6] Michel C. Jeruchim , Philip Balaban , K. Sam Shanmugan, *Simulation of Communication Systems: Modeling, Methodology and Techniques*, Kluwer Academic Publishers, Norwell, MA, 2000
- [7] S. Randel, A. Sierra, X. Liu, S. Chandrasekhar, and P. Winzer, "Study of multicarrier offset-QAM for spectrally efficient coherent optical communications," in *European Conference and Exposition on Optical Communications*, 2011, p. Th–11.
- [8] J. Zhao, "Offset-QAM multicarrier technology for optical systems and networks," in *Optical Communications and Networks (ICOON), 2014 13th International Conference on*, 2014, pp. 1–4.
- [9] S. Wulandari, "Gabungan Modulasi OQAM dan Pulse Shaping pada Sistem OFDM," *J. Tek. POMITS*, 2005.
- [10] Pooria Varahram , Somayeh Mohammady , Borhanuddin Mohd Ali , Nasri Sulaiman, *Power Efficiency in Broadband Wireless Communications*, CRC Press, Inc., Boca Raton, FL, 2014
- [11] N. Michailow *et al.*, "Generalized Frequency Division Multiplexing for 5th Generation Cellular Networks," *IEEE Trans. Commun.*, vol. 62, no. 9, pp. 3045–3061, Sep. 2014.
- [12] Manjunath, Sharath, "Analysis and Compensation of Power Amplifier Distortions in Wireless Communication Systems" (2015).

- [13] Teresa Araújo, Rui Dinis, *Analytical Evaluation of Nonlinear Distortion Effects on Multicarrier Signals*, CRC Press, Boca Raton, FL, 2015
- [14] A. A. M. Saleh, "Frequency-Independent and Frequency-Dependent Nonlinear Models of TWT Amplifiers," in *IEEE Transactions on Communications*, vol. 29, no. 11, pp. 1715-1720, November 1981.
- [15] Z. El-Khatib, L. MacEachern, S. A. Mahmoud, *Distributed CMOS Bidirectional Amplifiers: Broadbanding and Linearization Techniques*, Springer-Verlag, New York, 2012
- [16] F. H. Gregorio, "Analysis and compensation of nonlinear power amplifier effects in multi-antenna OFDM systems," Ph.D. dissertation, Helsinki Univ. Technol, Espoo, Finland, Nov. 2007.
- [17] M. M. Shammasi and S. M. Safavi, "Performance of a predistorter based on Saleh model for OFDM systems in HPA nonlinearity," 2012 14th International Conference on Advanced Communication Technology (ICACT), PyeongChang, 2012, pp. 148-152.
- [18] K. M. Gharaiberh, "Nonlinear Distortion in Wireless Systems", Wiley-IEEE Press, 2012.
- [19] Ali Cheaito, Jean-François Hélar, Matthieu Crussière, Yves Louët. EVM derivation of multicarrier signals to determine the operating point of the power amplifier considering clipping and predistortion. *EURASIP Journal on Wireless Communications and Networking*, SpringerOpen, 2016, 2016 (1)
- [20] I. Gaspar, M. Matthe, N. Michailow, L. Leonel Mendes, D. Zhang, and G. Fettweis, "Frequency-Shift Offset-QAM for GFDM," *IEEE Commun. Lett.*, vol. 19, no. 8, pp. 1454–1457, Aug. 2015.
- [21] N. Michailow, I. Gaspar, S. Krone, M. Lentmaier, and G. Fettweis, "Generalized frequency division multiplexing: Analysis of an alternative multi-carrier technique for next generation cellular systems," in *Wireless Communication Systems (ISWCS), 2012 International Symposium on*, 2012, pp. 171–175.
- [22] V. Selvakumar, "Analysis of LTE Radio Frame by eliminating Cyclic Prefix in OFDM and comparison of QAM and Offset-QAM," *Linnaeus Univ.*
- [23] N. Michailow *et al.*, "Generalized Frequency Division Multiplexing for 5th Generation Cellular Networks," in *IEEE Transactions on Communications*, vol. 62, no. 9, pp. 3045-3061, Sept. 2014.

LAMPIRAN A

PENGESAHAN TUGAS AKHIR

Departemen Teknik Elektro
Fakultas Teknologi Elektro – ITS

EE 184001 TUGAS AKHIR – 6 SKS

Nama Mahasiswa : Wilan Ibrahim Wahyudianto
Nomor Pokok : 0711144000125
Bidang Studi : Telekomunikasi Multimedia
Tugas Diberikan : Semester Genap Th. 2018/2019
Dosen Pembimbing : 1. Dr. Ir. Titiek Suryani, MT.

12 FEB 2019

Judul Tugas Akhir : Analisis Aplikasi Predistorsi untuk Mitigasi Nonlinear HPA pada Sistem GFDM
(*Analysis on Predistorter Application for Mitigating Nonlinear HPA in GFDM System*)

Uraian Tugas Akhir :

Sistem GFDM (*Generalized Frequency Division Multiplexing*) adalah sistem komunikasi *multi-carrier* yang dikembangkan untuk memenuhi kebutuhan teknologi seluler generasi kelima (5G). Sistem GFDM dikembangkan untuk menjadi alternatif dari sistem OFDM. Modulasi OQAM digunakan pada sistem GFDM dikarenakan beberapa kelebihan yakni memiliki sifat mendekati ortogonal sehingga sistem GFDM lebih tahan terhadap ISI dan ICI. *High power amplifier* (HPA) merupakan perangkat yang esensial dalam sistem komunikasi. Namun HPA memiliki titik saturasi yang dapat mengakibatkan penguatan tidak linier atau nonlinier, sehingga berujung dengan munculnya distorsi pada sinyal yang dikirimkan. Salah satu cara untuk memitigasi efek nonlinier HPA adalah dengan menggunakan teknik predistorsi.

Oleh karena itu pada Tugas Akhir ini, akan dianalisis aplikasi dari predistorsi untuk mengatasi distorsi nonlinier yang diakibatkan oleh HPA pada sistem GFDM yang menggunakan modulasi OQAM. Penelitian ini meliputi analisis kinerja BER untuk nilai EBO yang berbeda serta memfili distorsi residu setelah aplikasi predistorsi pada sistem. Tujuan akhir dari penelitian ini adalah untuk meningkatkan ketahanan pada sistem GFDM khususnya yang disebabkan oleh distorsi nonlinier. Sedangkan metode yang digunakan dalam penelitian ini adalah dengan melakukan simulasi menggunakan Matlab.

Dosen Pembimbing,

Dr. Ir. Titiek Suryani, MT.
NIP : 196411301989032001



Muhammad C. Riawan, ST, M.Eng, Ph.D.
NIP : 197311192000031001

Menyetujui,
Kepala Laboratorium Komunikasi Multimedia

Dr. Ir. Endrosono, DEA
NIP : 196504041991021001

[Halaman Ini Sengaja Dikosongkan]

LAMPIRAN B

Hasil *Bit Error Rate* Simulasi

Tabel 1. BER pada sistem GFDM/OQAM dengan *Roll-of-factor* =0.5 sebelum dan sesudah aplikasi *predistorter*

IBO	BER		Prosentase penuruan BER (%)
	<i>Predistorter</i>		
	Sebelum	Sesudah	
0	0.3003	0.0650	78.35498
1	0.2885	0.0299	89.63605
2	0.2700	0.0105	96.11111
3	0.2457	0.0029	98.8197
4	0.2171	4.8889×10^{-4}	99.77481
5	0.1875	3.3333×10^{-5}	99.9822
6	0.1572	0	100
7	0.1283	0	100
8	0.101	0	100
9	0.0747	0	100
10	0.0517	0	100
11	0.0330	0	100
12	0.0185	0	100
13	0.0087	0	100
14	0.0033	0	100
15	0.0010	0	100

Tabel 2. BER pada sistem GFDM/OQAM dengan *Roll-of-factor* =0.5 sebelum aplikasi *predistorter* melalui kanal AWGN

	SNR												
	0	2	4	6	8	10	12	14	16	18	20	22	24
-15	0.138728	0.132272	0.128067	0.125294	0.124844	0.124506	0.125189	0.125167	0.124394	0.124433	0.1246	0.124367	0.124567
-14	0.139828	0.131306	0.127472	0.123494	0.122311	0.121439	0.120744	0.120389	0.120417	0.12025	0.11985	0.120122	0.120039
-13	0.140028	0.132011	0.124544	0.121678	0.120267	0.118894	0.117983	0.116811	0.116267	0.116022	0.116433	0.116117	0.116033
-12	0.140417	0.131083	0.124639	0.12035	0.117911	0.116133	0.115283	0.114406	0.113994	0.11355	0.113361	0.113783	0.113222
-11	0.141194	0.131022	0.123633	0.119411	0.116761	0.114572	0.112917	0.11235	0.1121	0.111567	0.111683	0.111122	0.111167
-10	0.144033	0.132328	0.124506	0.117928	0.115344	0.112894	0.1119	0.111267	0.110711	0.110033	0.109867	0.109539	0.109422
-9	0.14705	0.13335	0.124861	0.118761	0.1142	0.111672	0.1108	0.109833	0.108894	0.108372	0.108494	0.108328	0.107933
-8	0.151428	0.137156	0.123339	0.119306	0.114294	0.112606	0.109472	0.108761	0.1081	0.107994	0.107456	0.107022	0.106861
-7	0.155667	0.140828	0.128922	0.120978	0.116528	0.112478	0.110672	0.109189	0.108361	0.107983	0.107383	0.106894	0.107194
-6	0.165078	0.145994	0.1339	0.1246	0.120044	0.115072	0.112972	0.111417	0.110389	0.109472	0.109322	0.108811	0.108778
-5	0.175322	0.1554	0.140822	0.131228	0.124394	0.120472	0.117283	0.115678	0.114506	0.113883	0.113261	0.113028	0.112767
-4	0.185433	0.165478	0.150739	0.140294	0.133061	0.127844	0.125222	0.122456	0.121756	0.120789	0.120328	0.12035	0.119594
-3	0.202633	0.181017	0.162417	0.151617	0.144006	0.139411	0.136561	0.135172	0.133167	0.131961	0.131472	0.13075	0.130794
-2	0.220922	0.198533	0.180667	0.168272	0.160706	0.155067	0.151072	0.149422	0.148506	0.147061	0.146717	0.146056	0.145772
-1	0.240639	0.2183	0.200744	0.18835	0.179844	0.175417	0.17105	0.167772	0.167078	0.165911	0.165717	0.165261	0.165078
0	0.263889	0.24035	0.224078	0.212067	0.2032	0.19761	0.19425	0.192417	0.191022	0.190322	0.18925	0.188972	0.188761
1	0.288411	0.265089	0.24835	0.237083	0.229772	0.22455	0.221356	0.219194	0.218822	0.217328	0.216806	0.216822	0.216522

Tabel 3. BER pada sistem GFDM/OQAM dengan *Roll-of-factor* =0.5 setelah aplikasi *predistorter* melalui kanal AWGN

	SNR												
	0	2	4	6	8	10	12	14	16	18	20	22	24
IBO													
-15	0.1397	0.09668	0.0569	0.0267	0.00874	0.00166	0.00012	5.56E-06	0	0	0	0	0
-14	0.1391	0.09618	0.05767	0.0267	0.00939	0.00151	0.00014	0	0	0	0	0	0
-13	0.1406	0.09648	0.05719	0.0277	0.00902	0.00151	0.00013	0	0	0	0	0	0
-12	0.1381	0.09639	0.05747	0.0276	0.00852	0.00156	0.00016	0	5.56E-06	0	0	0	0
-11	0.1374	0.09619	0.05696	0.0268	0.00896	0.00166	6.67E-05	0	0	0	0	0	0
-10	0.139	0.09649	0.05645	0.0263	0.00886	0.00159	0.00013	0	0	0	0	0	0
-9	0.1395	0.09593	0.05873	0.0276	0.0088	0.00176	9.44E-05	0	0	0	0	0	0
-8	0.1396	0.09737	0.05717	0.0279	0.00917	0.0015	0.00017	5.56E-06	0	0	0	0	0
-7	0.1404	0.09732	0.05713	0.0277	0.00943	0.00178	0.00017	1.67E-05	5.56E-06	0	0	0	0
-6	0.1421	0.09731	0.06018	0.0282	0.01018	0.00241	0.00034	7.22E-05	1.11E-05	0	0	0	0
-5	0.1458	0.10341	0.06318	0.032	0.01292	0.00432	0.00118	0.0005	0.000244	0.00015	8.33E-05	7.22E-05	6.11E-05
-4	0.1505	0.10852	0.07097	0.0403	0.01855	0.00757	0.00366	0.002094	0.001317	0.00103	0.000728	0.00073	0.00059
-3	0.1617	0.12092	0.08314	0.0521	0.02903	0.017	0.0099	0.00695	0.005033	0.00438	0.00338	0.00331	0.00318
-2	0.1761	0.13703	0.1005	0.0697	0.04758	0.03323	0.02369	0.018633	0.015694	0.01376	0.012344	0.01182	0.01131
-1	0.1919	0.15634	0.12263	0.0945	0.07338	0.05853	0.04764	0.041128	0.037033	0.0345	0.032878	0.03155	0.031
0	0.212	0.17775	0.14873	0.1244	0.1048	0.0925	0.08305	0.076517	0.071861	0.06974	0.067711	0.06703	0.0665
1	0.2342	0.20063	0.17512	0.1562	0.14207	0.13058	0.12486	0.119833	0.11775	0.11606	0.114661	0.1139	0.11422

Tabel 4. Nilai EVM terhadap IBO pada sistem GFDM/OQAM dengan *Roll-of-factor* =0.5 sebelum dan sesudah aplikasi *predistorter*

IBO	EVM		Prosentase penurunan EVM (%)
	<i>Predistorter</i>		
	Sebelum	Sesudah	
0	0.2400	0.2490	-3.75
1	0.2433	0.2164	11.0563
2	0.2497	0.1751	29.8758
3	0.2563	0.1314	48.7319
4	0.2611	0.0910	65.1474
5	0.2632	0.0574	78.1914
6	0.2629	0.0323	87.7139
7	0.2602	0.0155	94.0430
8	0.2550	0.0061	97.6078
9	0.2471	0.0020	99.1906
10	0.2364	9.6714×10^{-4}	99.5908
11	0.2227	8.1817×10^{-4}	99.6326
12	0.2066	6.8648×10^{-4}	99.6677
13	0.1886	5.7152×10^{-4}	99.6969
14	0.1696	4.7244×10^{-4}	99.7214
15	0.1511	3.8805×10^{-4}	99.7431

LAMPIRAN C

SINTAKS PROGRAM

SINTAKS GFDM-16 OQAM TANPA PREDISTORTER

```
clear all
% close all

%-----GFDM nonlinier tanpa predistorter-----%

% 16-OQAM
%Parameter
K=8;%number of subcarriers (used to be 8)
M=10;%number of subsymbols (used to be 10)
%Kindex = 1:K;
%s=M*K*8;
%Fs=16;
r=2.5;%length of the cyclic prefix (CP) in
multiples of 'subsymbols'
CP=r*K;
a=0.5;%roll-off
jsymbol=1000;
jumlahbit=4*(K-3*K/8)*(M-1)*jsymbol;
rng(2,'v4');
b=randi([0 1],1,jumlahbit);
sp=reshape(b,4,[]);

%mapping
xI1=2*sp(1,:)-1;
xI2=2-(2*sp(2,:)-1);
xQ1=2*sp(3,:)-1;
xQ2=2-(2*sp(4,:)-1);
xI=xI1.*xI2;xQ=xQ1.*xQ2;
qam=xI+1i*xQ; %QAM modulation
qamps=reshape(qam,[],1);
qamp=reshape(qam,(K-3*K/8),[]);
js=jsymbol-1;
Dcol2=qamdemod(qamps,16); %in2
```

```

% Symbol source
s=zeros(K,M);
d=[];
for k1=1:jsimbol
s(1:(K-3*K/8),2:M)=qamp(:,(M-1)*(k1-1)+1:(M-1)*(k1));
s=circshift(s,-1);
dd=reshape(s,[],1);
%s=zeros(K,M);
d(:,k1)=dd;
end
% Split into real and imag
di = real(d);
dq = imag(d);

% Pulse Shaping
% Meyer RRC (defined in time)
R=(0:(K-1))' -K/2-eps)/(a*K)+1/2;R(R<0)=0;R(R>1)=1;F=1-R;%Ramp rise/fall
R=R.^4.*(35 - 84*R+70*R.^2-20*R.^3);F=1-R;% Meyer auxiliary function
R=1/2*(cos(F*pi)+1);F=1-R;% Meyer RC rise/fall
R=sqrt(R);F=sqrt(F);%Meyer RRC
g=[F;zeros((M-2)*K,1);R];
g=g/sqrt(sum(g.^2));%normalization

gi = g; %domain waktu
%gqQAM = ifft(circshift(fft(gi), M/2)); %QAM di circshift sebesar M/2
gqQAM = ifft(circshift(fft(gi),M/2)); %%QAM tidak digeser domain waktu lagi

%IFFT
% Ai matrix
Ai = zeros(M*K, M*K);
n = 0:M*K-1; n=n';
w = exp(1j*2*pi/K);
%w=1;

```

```

for k=0:K-1
    for m=0:M-1
        Ai(:,m*K+k+1) = li^(mod(m,2))*circshift(gi,
m*K) .* w.^(k*n); %proses ifft
    end
end

% Aq OQAM matrix
% AqOQAM = zeros(M*K, M*K);
% for k=0:K-1
% for m=0:M-1
%         AqOQAM(:,m*K+k+1) =
li^(mod(m,2)+1)*circshift(gqOQAM, m*K) .*
w.^(k*n);
% end
% end

% Aq QAM matrix
AqQAM = zeros(M*K, M*K);
for k=0:K-1
for m=0:M-1
AqQAM(:,m*K+k+1)=li^(mod(m,2)+1)*circshift(gqQAM
,m*K) .* w.^(k*n);
end
end

xOQAM=(Ai)*di+(AqQAM)*dq; %sinyal modulasi hasil
ifft
%xQAM=(Ai)*di+(AqQAM)*dq;

%xOQAM_serial=reshape(xOQAM,1,[]);
%absxOQAM_serial=abs(xOQAM_serial);

% Add CP
x_cpOQAM=xOQAM([end-CP+(1:CP)],:);
xcpOQAM = [x_cpOQAM;xOQAM];
xxx=reshape(xcpOQAM,1,[]);
Pin=mean(mean(xcpOQAM.*conj(xcpOQAM)));Ex=sqrt(P
in);

```

```

% Ep=sqrt(5);*Ep
yx=xxx/Ex;
IBO=-15:1;%dB
kk=10.^(IBO/10);
% kk=0.01:0.1:6;
tabelsimbol=[];
for k=1:length(kk)
y=yx*sqrt(kk(1,k));
Pinn(1,k)=mean(mean(y.*conj(y)))
nx=y;
% xpred = abs(nx); phi = angle(nx);
% af=2.1587; bf=1.1517; ag=4.0033; bg=9.104;
% c = (af^2)/(4*bf); Lim = sqrt(c); r =
zeros(1,length(xpred)); psi0 = [];
% for i = 1:1:length(xpred)
%     if xpred(1,i) > 0 && xpred(1,i) < Lim
%         r(1,i) = (af-sqrt((af^2)-
4*bf*xpred(1,i)^2))./(2*bf*xpred(1,i));
%     elseif xpred(1,i) >= Lim
%         r(1,i) = 1/sqrt(bf);
%     end
%     psi0 = [psi0 (ag*r(1,i)^2)/(1+bg*r(1,i)^2)];
% end
% theta = phi - psi0;
% rr = r.*exp(1i*theta);
% Ppred(1,k)=mean(rr.*conj(rr))
xx=nx;
teta=angle(xx);
%HPA para
%aa=2.0587;ba=1.0517;ap=18.233;bp=9.004;
Rt=abs(xx);
%absx=mean(absx)
%max_absx=max(absx)
alfa_p=2.1587;beta_p=1.1517;alfa_q=4.033;beta_q=
9.104;
%absx=0:0.1:2;
%absxdB=10*log10(absx)
PR=(alfa_p.*Rt)./(1+beta_p.*Rt.^2);
QR=(alfa_q.*Rt.^2)./(1+beta_q.*Rt.^2);
sI=PR.*cos(QR);

```

```

sQ=PR.*sin(QR);
yy=0.5*(sI+1i*sQ).*exp(1i*teta);
YY(:,k)=yy';
yyy=yy*Ex/sqrt(kk(1,k));%/Ep
y1=reshape(yy,size(xcpQQAM));
yr=reshape(yyy,size(xcpQQAM));
Po(1,k)=mean(mean(y1.*conj(y1)))
% yrr=yr(21:100,:);
yrr = yr(min(size(x_cpQQAM))+1:min(size(yr)),:);
%Yr=fft(yrr,)

diiE=(Ai)'*yrr;diE=(real(diiE));
dqQE=(AqQQAM)'*yrr;dqE=(real(dqQE));
%dn=reshape(abs(s),[K*M 1]);
%diE=diE.*dn;
%dqE=dqE.*dn;
dE=diE+1i*dqE;
dE=reshape(dE,K,[]);
dEc=circshift(dE,1);
ddE=dEc(1:(K-3*K/8),:);
for k2=1:jsimbol
    dddE(:,(k2-1)*(M-1)+(1:(M-1)))=ddE(:,(k2-
1)*M+(2:M));
end
% figure(2)
% plot(real(dddE),imag(dddE),'b*')
dEevm=reshape(dddE,[],1);
[NX1(1,k) NX2(1,k) NX3(1,k)]=evm2(dEevm,qamps);
tabelsimbol(:,k)=dEevm;
[tabelrerata(:,k)]=phserror2(dEevm,Dcol2);
b1=(1+sign(real(dddE)))/2;
b2=(1-sign(abs(real(dddE))-2))/2;
b3=(1+sign(imag(dddE)))/2;
b4=(1-sign(abs(imag(dddE))-2))/2;
b1s=reshape(b1,1,[]);
b2s=reshape(b2,1,[]);
b3s=reshape(b3,1,[]);
b4s=reshape(b4,1,[]);
bEp=[b1s;b2s;b3s;b4s];

```



```

bE=reshape(bEp,1,[]);
[NB,BERR]=biterr(bE,b)
BER(1,k)=BERR;
end

% psdin= fft(xcpOQAM,4*M*K);
% psdin=psdin/std(psdin);      %normalize to ~0dB
% % psdin=psdin';
% Fss=4*K*M;
%
[Pxx,Fx]=pwelch(psdin,[],[],[],Fss,'centered');

YY1=reshape(YY(:,16),size(xcpOQAM));
psdout= fft(YY1,5*M*K);
psdout=psdout/std(psdout);      %normalize to ~0dB
% psdout=psdout';
Fss=10*K*M;
[PyY,Fy]=pwelch(psdout,[],[],[],Fss,'centered');

% YY2=reshape(YY(:,4),size(xcpOQAM));
% psdout= fft(YY2,4*M*K);
% psdout=psdout/std(psdout);      %normalize to ~0dB
% % psdout=psdout';
% Fss=10*K*M;
%
[PyY1,Fy1]=pwelch(psdout,[],[],[],Fss,'centered'
);
%
% YY3=reshape(YY(:,8),size(xcpOQAM));
% psdout= fft(YY3,4*M*K);
% psdout=psdout/std(psdout);      %normalize to ~0dB
% % psdout=psdout';
% Fss=10*K*M;
%
[PyY2,Fy2]=pwelch(psdout,[],[],[],Fss,'centered'
);
%
% YY4=reshape(YY(:,12),size(xcpOQAM));
% psdout= fft(YY4,4*M*K);

```

```

% psdout=psdout/std(psdout);    %normalize to ~0dB
% % psdout=psdout';
% Fss=10*K*M;
%
[Pyy3,Fy3]=pwelch(psdout,[],[],[],Fss,'centered'
);
%
% YY5=reshape(YY(:,16),size(xcpOQAM));
% psdout=fft(YY5,4*M*K);
% psdout=psdout/std(psdout);    %normalize to ~0dB
% % psdout=psdout';
% Fss=10*K*M;
%
[Pyy4,Fy4]=pwelch(psdout,[],[],[],Fss,'centered'
);

PidB=10*log10(Pinn);
% PpreddB=10*log10(Ppred);
PodB=10*log10(Po);
Gain=Po./Pinn;GdB=10*log10(Gain);
figure(3)
semilogy(IBO,BER,'b:')
axis([-10 1 10^-6 10^0])
hold on
figure(4)
% plot(PpreddB,PodB)
% hold on
% plot(PidB,PodB+(-15-(min(PodB))), 'b:')
plot(PidB,PodB,'b:')
hold on
figure(5)
plot(PidB,GdB,'b--')
hold on
figure(6)
plot(IBO,NX1,'b-.')
hold on
figure(7)
plot(PidB,GdB/max(GdB), 'b--')
hold on

```

```

figure(8)
plot(Fy,10*log10(Pyy),'b')
hold on
figure(9)
plot(Pinn,Po,'b')
hold on
% plot(Fy1,10*log10(Pyy1),'b')
% hold on
% plot(Fy2,10*log10(Pyy2),'g')
% hold on
% plot(Fy3,10*log10(Pyy3),'k')
% hold on
% plot(Fy4,10*log10(Pyy4),'c')
% hold on
% Fx,10*log10(Pxx),'b',
ZZZ=[-3+3i;-3+1i;-3-1i;-3-3i;-1+3i;-1+1i;-1-1i;-
1-3i;1+3i;1+1i;1-1i;1-3i;3+3i;3+1i;3-1i;3-3i];

```

SINTAKS GFDM-16 OQAM DENGAN PREDISTORTER

```
clear all
% close all

%-----GFDM nonlinier dengan predistorter-----%

% 16-QAM
%Parameter
K=8;%number of subcarriers
M=10;%number of subsymbols
%Kindex = 1:K;
%s=M*K*8;
%Fs=16;
r=2.5;%length of the cyclic prefix (CP) in
multiples of 'subsymbols'
CP=r*K;
a=0.5;%roll-off
jsymbol=1000;
jumlahbit=4*(K-3*K/8)*(M-1)*jsymbol;
rng(2,'v4');
b=randi([0 1],1,jumlahbit);
sp=reshape(b,4,[]);

%mapping
xI1=2*sp(1,:)-1;
xI2=2-(2*sp(2,:)-1);
xQ1=2*sp(3,:)-1;
xQ2=2-(2*sp(4,:)-1);
xI=xI1.*xI2;xQ=xQ1.*xQ2;
qam=xI+1i*xQ; %QAM modulation
qamps=reshape(qam,[],1);
qamp=reshape(qam,(K-3*K/8),[]);
js=jsymbol-1;
Dcol2=qamdemod(qamps,16); %in2

% Symbol source
s=zeros(K,M);
d=[];
for k1=1:jsymbol
```

```

s(1:(K-3*K/8),2:M)=qamp(:,(M-1)*(k1-1)+1:(M-
1)*(k1));
s=circshift(s,-1);
dd=reshape(s,[],1);
s=zeros(K,M);
d(:,k1)=dd;
end
% Split into real and imag
di = real(d);
dq = imag(d);

% Pulse Shaping
% Meyer RRC (defined in time)
R=(0:(K-1))' -K/2-
eps)/(a*K)+1/2;R(R<0)=0;R(R>1)=1;F=1-R;% Ramp
rise/fall
R=R.^4.*(35 - 84*R+70*R.^2-20*R.^3);F=1-R;% Meyer
auxiliary function
R=1/2*(cos(F*pi)+1);F=1-R;% Meyer RC rise/fall
R=sqrt(R);F=sqrt(F);%Meyer RRC
g=[F;zeros((M-2)*K,1);R];
g=g/sqrt(sum(g.^2));%normalization

gi = g; %domain waktu
%gqOQAM = ifft(circshift(fft(gi), M/2)); %OQAM di
circshift sebesar M/2
gqQAM = ifft(circshift(fft(gi),M/2)); %%QAM tidak
digeser domain waktu lagi

%IFFT
% Ai matrix
Ai = zeros(M*K, M*K);
n = 0:M*K-1; n=n';
w = exp(1j*2*pi/K);
%w=1;
for k=0:K-1
    for m=0:M-1
        Ai(:,m*K+k+1) = 1i^(mod(m,2))*circshift(gi, m*K)
        .* w.^(k*n); %proses ifft

```

```

end
end

% Aq OQAM matrix
% AqOQAM = zeros(M*K, M*K);
% for k=0:K-1
% for m=0:M-1
%
%           AqOQAM(:,m*K+k+1)           =
li^(mod(m,2)+1)*circshift(gqOQAM,      m*K)   .*
w.^(k*n);
% end
% end

% Aq QAM matrix
AqQAM = zeros(M*K, M*K);
for k=0:K-1
for m=0:M-1
AqQAM(:,m*K+k+1)           =
li^(mod(m,2)+1)*circshift(gqQAM,      m*K)   .*
w.^(k*n);
end
end

xOQAM=(Ai)*di+(AqQAM)*dq; %sinyal modulasi hasil
ifft
%xQAM=(Ai)*di+(AqQAM)*dq;

% Add CP
x_cpOQAM=xOQAM([end-CP+(1:CP)],:);
xcpOQAM = [x_cpOQAM;xOQAM];
xxx=reshape(xcpOQAM,1,[]);
Pin=mean(mean(xcpOQAM.*conj(xcpOQAM)));Ex=sqrt(P
in);

yx=xxx/Ex;

IBO=-15:1;%dB
kk=10.^(IBO/10);

```

```

% kk=0.01:0.1:6;
tabelsimbol=[];

for k=1:length(kk)
y=yx*sqrt(kk(1,k));
ZZ2(k,:)=reshape(y,1,[]);
Pinn(1,k)=mean(mean(y.*conj(y)))
nx=y;
xpred = abs(nx); phi = angle(nx);
af=2.1587; bf=1.1517; ag=4.0033; bg=9.104;
c = (af^2)/(4*bf); Lim = sqrt(c); r =
zeros(1,length(xpred)); psi0 = [];
for i = 1:1:length(xpred)
    if xpred(1,i) > 0 && xpred(1,i) < Lim
        r(1,i) = (af-sqrt((af^2)-
4*bf*xpred(1,i)^2))./(2*bf*xpred(1,i));
        elseif xpred(1,i) >= Lim
            r(1,i) = 1/sqrt(bf);
        end
        psi0 = [psi0 (ag*r(1,i)^2)/(1+bg*r(1,i)^2)];
end
theta = phi - psi0;
rr = r.*exp(1i*theta);
Ppred(1,k)=mean(rr.*conj(rr))
xx=rr;
teta=angle(xx);
%HPA para
%aa=2.0587;ba=1.0517;ap=18.233;bp=9.004;
Rt=abs(xx);
%max_absx=mean(absx)
%max_absx=max(absx)
alfa_p=2.1587;beta_p=1.1517;alfa_q=4.033;beta_q=
9.104;
%absx=0:0.1:2;
%absxdB=10*log10(absx)
PR=(alfa_p.*Rt)./(1+beta_p.*Rt.^2);
QR=(alfa_q.*Rt.^2)./(1+beta_q.*Rt.^2);
sI=PR.*cos(QR);
sQ=PR.*sin(QR);

```

```

yy=(sI+li*sQ) .*exp (li*teta);
YY(:,k)=yy';
yyy=yy*Ex/sqrt(kk(1,k));
y1=reshape(yy,size(xcpOQAM));
yr=reshape(yyy,size(xcpOQAM));
ZZ1(k,:)=reshape(y1,1,[]);
Po(1,k)=mean(mean(y1.*conj(y1)))
% yrr=yr(21:100,:);
yrr = yr(min(size(x_cpOQAM))+1:min(size(yr)),:);
%Yr=fft(yrr,);
diiE=((Ai))'*yrr;diE=(real(diiE));
dqQE=((AqQAM))'*yrr;dqE=(real(dqQE));
%dn=reshape(abs(s),[K*M 1]);
%diE=diE.*dn;
%dqE=dqE.*dn;
dE=diE+j*dqE;
dE=reshape(dE,K,[]);
dEc=circshift(dE,1);
ddE=dEc(1:(K-3*K/8),:);
for k2=1:jsimbol
    dddE(:,(k2-1)*(M-1)+(1:(M-1)))=ddE(:,(k2-
1)*M+(2:M));
end
% figure(2)
% plot(real(dddE),imag(dddE),'b*')
dEevm=reshape(dddE,[],1);
[NX1(1,k) NX2(1,k) NX3(1,k)]=evm2(dEevm,qamps);
tabelsimbol(:,k)=dEevm;
[tabelrerata(:,k)]=phserror2(dEevm,Dcol2);
b1=(1+sign(real(dddE)))/2;
b2=(1-sign(abs(real(dddE))-2))/2;
b3=(1+sign(imag(dddE)))/2;
b4=(1-sign(abs(imag(dddE))-2))/2;
b1s=reshape(b1,1,[]);
b2s=reshape(b2,1,[]);
b3s=reshape(b3,1,[]);
b4s=reshape(b4,1,[]);
bEp=[b1s;b2s;b3s;b4s];
bE=reshape(bEp,1,[]);
[NB,BERR]=biterr(bE,b)

```



```

BER(1, k)=BERR;
z(1, k)=0.852*IBO(1, k)+3.831;
end

% psdin= fft(xcpOQAM,5*M*K);
% psdin=psdin/std(psdin);      %normalize to ~0dB
% % psdin=psdin';
% Fss=10*K*M;
%
[Pxx, Fx]=pwelch(psdin, [], [], [], Fss, 'centered');

YY1=reshape(YY(:, 16), size(xcpOQAM));
psdout= fft(YY1, 5*M*K);
psdout=psdout/std(psdout);      %normalize to ~0dB
% psdout=psdout';
Fss=10*K*M;
[Py, Fy]=pwelch(psdout, [], [], [], Fss, 'centered');

PidB=10*log10(Pinn);
PpreddB=10*log10(Ppred);
PodB=10*log10(Po);
Gain=Po./Pinn; GdB=10*log10(Gain);
figure(3)
semilogy(IBO, BER, 'b:')
axis([-10 1 10^-6 10^0])
hold on
figure(4)
%
% plot(PidB, PpreddB+(min(PodB)-
min(PpreddB)), 'r')
plot(PidB, PpreddB, 'r:')
hold on
plot(PidB, PodB, 'k:')
hold on
% plot(IBO, z+(min(PodB)-min(z)), 'g')
figure(5)
plot(PidB, GdB, 'r--')
hold on
figure(6)

```

```

plot(IBO, NX1, 'r-.')
hold on
figure(7)
plot(PidB, GdB, 'r--')
hold on
figure(8)
plot(Fy, 10*log10(Pyy), 'm')
hold on
figure(9)
plot(Pinn, Po, 'g')
hold on

% Fx, 10*log10(Pxx), 'k',
ZZZ1=[-3+3i;-3+1i;-3-1i;-3-3i;-1+3i;-1+1i;-1-
1i;-1-3i;1+3i;1+1i;1-1i;1-3i;3+3i;3+1i;3-1i;3-
3i];

```

[Halaman Ini Sengaja Dikosongkan]

SINTAKS GFDM-8 PSK TANPA PREDISTORTER

```
clear all
% close all
clc
%-----GFDM nonlinier tanpa predistorter-----%

% 8-PSK
%Parameter
K=8;%number of subcarriers
M=10;%number of subsymbols
%Kindex = 1:K;
%s=M*K*8;
%Fs=16;
r=2.5;%length of the cyclic prefix (CP) in
multiples of 'subsymbols'
CP=r*K;
a=0.5;%roll-off
jsymbol=1000;
rng(2,'v4');
b=randi([0 1],1,(K-3*K/8)*(M-1)*3*jsymbol);
sp=reshape(b,[],3);
D = bi2de(sp); %in2
%mapping
txpsk=pskmod(D,8,pi/8,'gray');
txpskp=reshape(txpsk,(K-3*K/8),[]);
% Symbol source
s=zeros(K,M);
d=[];
for k1=1:jsymbol
    s(1:(K-3*K/8),2:M)=txpskp(:,(M-1)*(k1-1)+1:(M-1)*(k1));
    s=circshift(s,-1);
    dd=reshape(s,[],1);
    %s=zeros(K,M);
    d(:,k1)=dd;
end
% Split into real and imag
di = real(d);
dq = imag(d);
```

```

% Pulse Shaping
% Meyer RRC (defined in time)
R = ((0:(K-1))' - K/2 -
eps) / (a*K + 1/2); R(R < 0) = 0; R(R > 1) = 1; F = 1 - R; % Ramp
rise/fall
R = R.^4 .* (35 - 84*R + 70*R.^2 - 20*R.^3); F = 1 - R; % Meyer
auxiliary function
R = 1/2 * (cos(F*pi) + 1); F = 1 - R; % Meyer RC rise/fall
R = sqrt(R); F = sqrt(F); % Meyer RRC
g = [F; zeros((M-2)*K, 1); R];
g = g / sqrt(sum(g.^2)); % normalization

gi = g;
gqOffs = ifft(circshift(fft(gi), M/2)); %% 8-PSK
digeser sebesar M/2

%IFFT
% Ai matrix
Ai = zeros(M*K, M*K);
n = 0:M*K-1; n = n';
w = exp(1j*2*pi/K);
%w=1;
for k=0:K-1
for m=0:M-1
Ai(:, m*K+k+1) = 1i^(mod(m,2)) * circshift(gi, m*K)
.* w.^(k*n); %proses ifft
end
end

% Aq Offset 8-PSK matrix
AqOPSK = zeros(M*K, M*K);
for k=0:K-1
for m=0:M-1
AqOPSK(:, m*K+k+1) =
1i^(mod(m,2)+1) * circshift(gqOffs, m*K) .*
w.^(k*n);
end
end

```

```

xOPSK=(Ai) *di+(AqOPSK) *dq; %sinyal modulasi hasil
ifft

% Add CP
x_cpOPSK=xOPSK ([end-CP+(1:CP)], :);
xcpOPSK=[x_cpOPSK;xOPSK];
xxx=reshape(xcpOPSK,1, []);

Pin=mean(mean(xcpOPSK.*conj(xcpOPSK)));
Ex=sqrt(Pin);
yx=xxx/Ex;
IBO=-15:1;
kk=10.^(IBO/10);
tabelrerata=[];
for k=1:length(kk)
y=yx*sqrt(kk(1,k));
Pinn=mean(mean(y.*conj(y)))
xx=y;
teta=angle(xx);
Rt=abs(xx);
alfa_p=2.1587;beta_p=1.1517;alfa_q=4.033;beta_q=
9.104;
PR=(alfa_p.*Rt)./(1+beta_p.*Rt.^2);
QR=(alfa_q.*Rt.^2)./(1+beta_q.*Rt.^2);
sI=PR.*cos(QR);
sQ=PR.*sin(QR);
yy=(sI+j*sQ).*exp(j*teta);
yyy=1*yy*Ex/sqrt(kk(1,k));
yr=reshape(yyy,size(xcpOPSK));
% receiver
% yrr=yrr(21:100, :);
yOPSK
yr(min(size(x_cpOPSK))+1:min(size(yr)), :);

diiEOPSK=(Ai)'*yOPSK;diE=real(diiEOPSK);
dqEOPSK=(AqOPSK)'*yOPSK;dqE=real(dqEOPSK);

dE=diE+li*dqE;

```

```

dE=reshape(dE,K,[]);
dEc=circshift(dE,1);
ddE=dEc(1:(K-3*K/8),:);
for k2=1:jsimbol
dddE(:,(k2-1)*(M-1)+(1:(M-1)))=ddE(:,(k2-
1)*M+(2:M));
end
rxpsk=reshape(dddE,[],1); %in1
[NX1(1,k) NX2(1,k) NX3(1,k)]=evm3(rxpsk,txpsk);
tabelsimbol(:,k)=rxpsk;
[tabelrerata(:,k)]=phserror1(rxpsk,D);
Drx=pskdemod(rxpsk,8,pi/8,'gray');
DDrx=de2bi(Drx);
rxb=reshape(DDrx,1,[]);
[NB,BERR]=biterr(rxb,b)
BER(1,k)=BERR;
end
figure(1)
semilogy(IBO,BER,'b:');
figure(2)
plot(IBO,NX1,'b-.');

ZZZ=[0.9239+0.3827i;          0.3827+0.9239i;          -
0.9239+0.3827i;  -0.3827+0.9239i;  0.9239-0.3827i;
0.3827-0.9239i;  -0.9239-0.3827i;  -0.3827-0.9239i]

```

SINTAKS GFDM-8 PSK DENGAN PREDISTORTER

```
clear all
% close all
clc
%-----GFDM nonlinier dengan predistorter-----%

%8-PSK
%Parameter
K=8;%number of subcarriers
M=10;%number of subsymbols
%Kindex = 1:K;
%s=M*K*8;
%Fs=16;
r=2.5;%length of the cyclic prefix (CP) in
multiples of 'subsymbols'
CP=r*K;
a=0.5;%roll-off
jsymbol=1000;
rng(2,'v4');
b=randi([0 1],1,(K-3*K/8)*(M-1)*3*jsymbol);
sp=reshape(b,[],3);
D = bi2de(sp); %in2
%mapping
txpsk=pskmod(D,8,pi/8,'gray');
txpskp=reshape(txpsk,(K-3*K/8),[]);
% Symbol source
s=zeros(K,M);
d=[];
for k1=1:jsymbol
    s(1:(K-3*K/8),2:M)=txpskp(:,(M-1)*(k1-1)+1:(M-1)*(k1));
    s=circshift(s,-1);
    dd=reshape(s,[],1);
    %s=zeros(K,M);
    d(:,k1)=dd;
end
% Split into real and imag
di = real(d);
dq = imag(d);
```



```

% Pulse Shaping
% Meyer RRC (defined in time)
R = ((0:(K-1))' - K/2 -
eps) / (a*K + 1/2); R(R < 0) = 0; R(R > 1) = 1; F = 1 - R; % Ramp
rise/fall
R = R.^4 .* (35 - 84*R + 70*R.^2 - 20*R.^3); F = 1 - R; % Meyer
auxiliary function
R = 1/2 * (cos(F*pi) + 1); F = 1 - R; % Meyer RC rise/fall
R = sqrt(R); F = sqrt(F); % Meyer RRC
g = [F; zeros((M-2)*K, 1); R];
g = g / sqrt(sum(g.^2)); % normalization

gi = g;
gqOffs = ifft(circshift(fft(gi), M/2)); %% 8-PSK
digeser sebesar M/2

%IFFT
% Ai matrix
Ai = zeros(M*K, M*K);
n = 0:M*K-1; n = n';
w = exp(1j*2*pi/K);
%w=1;
for k=0:K-1
for m=0:M-1
Ai(:, m*K+k+1) = 1i^(mod(m,2)) * circshift(gi, m*K)
.* w.^(k*n); %proses ifft
end
end

% Aq Offset 8-PSK matrix
AqOPSK = zeros(M*K, M*K);
for k=0:K-1
for m=0:M-1
AqOPSK(:, m*K+k+1) =
1i^(mod(m,2)+1) * circshift(gqOffs, m*K) .*
w.^(k*n);
end
end

```

```

xOPSK=(Ai)*di+(AqOPSK)*dq; %sinyal modulasi hasil
ifft

% Add CP
x_cpOPSK=xOPSK([end-CP+(1:CP)],:);
xcpOPSK=[x_cpOPSK;xOPSK];
xxx=reshape(xcpOPSK,1,[]);

Pin=mean(mean(xcpOPSK.*conj(xcpOPSK)));
Ex=sqrt(Pin);
yx=xxx/Ex;
IBO=-15:1;
kk=10.^(IBO/10);
tabelrerata=[];
for k=1:length(kk)
y=yx*sqrt(kk(1,k));
Pinn=mean(mean(y.*conj(y)))
nx=y;
xpred = abs(nx); phi = angle(nx);
af=2.1587; bf=1.1517; ag=4.0033; bg=9.104;
c = (af^2)/(4*bf); Lim = sqrt(c); r =
zeros(1,length(xpred)); psi0 = [];
for i = 1:1:length(xpred)
    if xpred(1,i) > 0 && xpred(1,i) < Lim
        r(1,i) = (af-sqrt((af^2)-
4*bf*xpred(1,i)^2))./(2*bf*xpred(1,i));
    elseif xpred(1,i) >= Lim
        r(1,i) = 1/sqrt(bf);
    end
    psi0 = [psi0 (ag*r(1,i)^2)/(1+bg*r(1,i)^2)];
end
theta = phi - psi0;
rr = r.*exp(1i*theta);
xx=rr;
teta=angle(xx);
Rt=abs(xx);
alfa_p=2.1587;beta_p=1.1517;alfa_q=4.033;beta_q=
9.104;

```

```

PR=(alfa_p.*Rt)./(1+beta_p.*Rt.^2);
QR=(alfa_q.*Rt.^2)./(1+beta_q.*Rt.^2);
sI=PR.*cos(QR);
sQ=PR.*sin(QR);
yy=(sI+j*sQ).*exp(j*teta);
yyy=1*yy*Ex/sqrt(kk(1,k));
yr=reshape(yyy,size(xcpOPSK));
% receiver
% yrr=yrr(21:100,:);
yOPSK
yr(min(size(x_cpOPSK)+1:min(size(yr)),:));

diiEOPSK=(Ai)'*yOPSK;diE=real(diiEOPSK);
dqQEOPSK=(AqOPSK)'*yOPSK;dqE=real(dqQEOPSK);

dE=diE+1i*dqE;
dE=reshape(dE,K,[]);
dEc=circshift(dE,1);
ddE=dEc(1:(K-3*K/8),:);
for k2=1:jsimbol
dddE(:,(k2-1)*(M-1)+(1:(M-1)))=ddE(:,(k2-
1)*M+(2:M));
end
rxpsk=reshape(dddE,[],1);
[NX1(1,k) NX2(1,k) NX3(1,k)]=evm3(rxpsk,txpsk);
tabelsimbol(:,k)=rxpsk;
[tabelrerata(:,k)]=phserror1(rxpsk,D);
Drx=pskdemod(rxpsk,8,pi/8,'gray');
DDrx=de2bi(Drx);
rxb=reshape(DDrx,1,[]);
[NB,BERR]=biterr(rxb,b)
BER(1,k)=BERR;
end
figure(1)
semilogy(1BO,BER,'b:')
figure(2)
plot(1BO,NX1,'b-.')

```

```

ZZZ=[0.9239+0.3827i;          0.3827+0.9239i;          -
0.9239+0.3827i; -0.3827+0.9239i; 0.9239-0.3827i;
0.3827-0.9239i; -0.9239-0.3827i; -0.3827-0.9239i]

```

SINTAKS PENDUKUNG

```

% evm2.m
function [out1,out2,out3] = evm2(in1,in2)
% Digunakan untuk mencari nilai evm dari modulasi
16-QAM / QAM
Z=in1;
X=in2;
N=length(in1);
s1=1+1i;s2=1+3i;s3=1-1i;s4=1-
3i;s5=3+1i;s6=3+3i;s7=3-1i;s8=3-3i;
s9=-1+1i;s10=-1+3i;s11=-1-1i;s12=-1-3i;s13=-
3+1i;s14=-3+3i;s15=-3-1i;s16=-3-3i;
%-----%
%-----%
e1=Z-s1;e2=Z-s2;e3=Z-s3;e4=Z-s4;e5=Z-s5;e6=Z-
s6;e7=Z-s7;e8=Z-s8;
e9=Z-s9;e10=Z-s10;e11=Z-s11;e12=Z-s12;e13=Z-
s13;e14=Z-s14;e15=Z-s15;e16=Z-s16;
%-----%
%-----%
eT=[e1 e2 e3 e4 e5 e6 e7 e8 e9 e10 e11 e12 e13 e14
e15 e16];
%-----%
%-----%
E=[];
for i=1:N
    E=[E; min(eT(i,:))];
end
P1=mean(abs(E).^2);
%-----%
%-----%
% P0=mean(Z.*conj(Z));
P0=mean(X.*conj(X));

```

```

out1=sqrt (P1/P0);
out2=out1*100;
out3=20*log10 (out1);
% out1=(sqrt (S/N)/sqrt (Sref/N));
% out2=20*log10 (out1);

%evm3.m
function [out1,out2,out3] = evm3(in1,in2)
% Digunakan untuk mencari nilai evm dari modualasi
8-PSK
Z=in1;
X=in2;
N=length(in1);
s1=0.9239+0.3827i;s2=0.3827+0.9239i;s3=-
0.9239+0.3827i;s4=-0.3827+0.9239i;s5=0.9239-
0.3827i;s6=0.3827-0.9239i;s7=-0.9239-
0.3827i;s8=-0.3827-0.9239i;
%-----
%-----%
e1=Z-s1;e2=Z-s2;e3=Z-s3;e4=Z-s4;e5=Z-s5;e6=Z-
s6;e7=Z-s7;e8=Z-s8;
%-----
%-----%
eT=[e1 e2 e3 e4 e5 e6 e7 e8];
%-----
%-----%
E=[];
for i=1:N
    E=[E; min(eT(i,:))];
end
P1=mean (abs (E) .^2);
%-----
%-----%
% P0=mean (Z.*conj (Z));
P0=mean (X.*conj (X));

out1=sqrt (P1/P0);

```

```

out2=out1*100;
out3=20*log10(out1);

```

```

%phserror1.m

```

```

function [out1] = phserror1(in1,in2)
% program ini untuk 8-PSK, digunakan untuk mencari
rata-rata simbol di
% masing-masing titik.
% in1 adalah simbol yg datang di penerima belum
dalam bentuk desimal (rxpsk)
% in2 adalah urutan simbol yg dikirimkan dalam
bentuk desimal(D)
M=8;
Mn=zeros(M,1);
Z1=in1;
Z2=in2;
for i=0:1:M-1
    cc=1;
    Sm=0;
    for j=1:length(Z1);
        if Z2(j) == i
            Sm = Sm + Z1(j);
            cc=cc+1;
        end
    end
    Mn(i+1,1)=Sm/cc;
end
out1=Mn; %keluaran dala bentuk vektor kolom

```

```

%phserror2

```

```

function [out1] = phserror2(in1,in2)
% program ini untuk 16-PSK, digunakan untuk
mencari nilai rata-rata simbol
% di masing-masing titik
% in1 adalah simbol yg datang di penerima belum
dalam bentuk desimal (dEevm)
% in2 adalah urutan simbol yg dikirimkan dalam
bentuk desimal(Dcol2)

```

```

M=16;
Mn=zeros(M,1);
Z1=in1;
Z2=in2;
for i=0:1:M-1
    cc=1;
    Sm=0;
    for j=1:length(Z1);
        if Z2(j) == i
            Sm = Sm + Z1(j);
            cc=cc+1;
        end
    end
    Mn(i+1,1)=Sm/cc;
end
out1=Mn; %keluaran dala bentuk vektor kolom

```

BIOGRAFI PENULIS



Penulis bernama Wildan Bramaji Wahyudianto. Lahir di Surabaya, 10 Januari 1996 merupakan putra pertama dari Bapak Kelik Prasetyo Wahyudianto dan Ibu Heny Triwidyastuti. Penulis telah menempuh pendidikan formal di SD Negeri Jember Lor 3, SMP Negeri 2 Kupang, dan SMA Negeri 7 Malang. Pada tahun 2014 penulis melanjutkan pendidikan S1 di Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya dan mengambil bidang studi Telekomunikasi Multimedia. Selama menjadi mahasiswa penulis aktif dalam kegiatan organisasi mahasiswa dan menjadi koordinator dan asisten praktikum. Penulis dapat dihubungi melalui alamat email wildanbramajiw@gmail.com