

**RANCANG BANGUN PEMANCAR OFDM DENGAN  
MENGUNAKAN DSK TMS320C6713 BERBANTUAN  
MATLAB SIMULINK**

**SKRIPSI**

Oleh

**BENNY SAHAT MONANG**

**04 04 03 020 2**



DEPARTEMEN TEKNIK ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
GENAP 2007 / 2008

**RANCANG BANGUN PEMANCAR OFDM DENGAN  
MENGUNAKAN DSK TMS320C6713 BERBANTUAN  
MATLAB SIMULINK**

**SKRIPSI**

Oleh

**BENNY SAHAT MONANG**

**04 04 03 020 2**



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

**DEPARTEMEN TEKNIK ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS INDONESIA**

**GENAP 2007 / 2008**

## **PERNYATAAN KEASLIAN SKRIPSI**

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

### **RANCANG BANGUN PEMANCAR OFDM DENGAN MENGGUNAKAN DSK TMS320C6713 BERBANTUAN MATLAB SIMULINK**

yang dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada program studi Teknik Elektro Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari skripsi yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Indonesia maupun di Perguruan Tinggi atau Instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Depok, Juni 2008

**Benny Sahat Monang**

NPM 04 04 03 020 2

## **PENGESAHAN**

Skripsi dengan judul:

### **RANCANG BANGUN PEMANCAR OFDM DENGAN MENGGUNAKAN DSK TMS320C6713 BERBANTUAN MATLAB SIMULINK**

dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada program studi Teknik Elektro Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia dan disetujui untuk diajukan dalam sidang ujian skripsi

Depok, Juni 2008

Dosen Pembimbing

**Dr. Ir. Arman Djohan, M.Eng**

**NIP. 131 476 472**

## UCAPAN TERIMA KASIH

Puji syukur limpah terima kasih penulis sampaikan kepada Allah yang dalam kasih dan kuasanya terus memelihara dan memampukan penulis untuk menyelesaikan skripsi ini dengan baik.

Selain itu penulis juga mengucapkan terima kasih yang tidak terkira kepada:

1. Bapak Dr. Ir. Arman Djohan, M.Eng sebagai pembimbing skripsi yang telah sangat banyak membantu dan menolong penulis dalam menyelesaikan skripsi ini.
2. Orang tua serta kedua adik penulis. Entah dari mana semangat dan tenaga yang terus mendorong penulis untuk menyelesaikan skripsi ini selain berasal dari doa yang Orang tua dan keluarga panjatkan untuk penulis.
3. Teman-teman baik penulis: Syana, Rotu, Noel, Iwan, yang telah dengan lapang hati menerima setiap keluh kesah penulis ketika mengerjakan skripsi ini. Juga Pontas, Dicky, dan Hermin yang telah membantu proses di laboratorium.
4. Teman-teman di Pengurus Harian PO UI: Olla, Dedi, Martina, Helda, David, Cia, dan Ruth yang telah membantu tugas penulis di Pengurus Harian hingga akhirnya penulis dapat berkonsentrasi dalam penyelesaian skripsi ini.
5. Teman-teman pengurus dan pelayan BPK PT GPIB "Kasih Karunia": Dyna, Lidia, Ussy, Titin, Ka Febri, Ka Vivi, Ka Joshua, dan Ka Anekke. Semangat kalian memicu penulis untuk terus berjuang.
6. Teman-teman KTB RPA: Gaby, Suci, Dannies, dan Martina karena sudah bersedia saling mendoakan.

Penulis juga sungguh berterima kasih kepada semua pihak yang mau membantu dan mempermudah penulis. Tidak ada yang dapat penulis lakukan selain mengucapkan banyak terima kasih kepada semua pihak baik yang disebutkan maupun yang tidak disebutkan.

Benny Sahat Monang  
Departemen Teknik Elektro  
NPM 04 04 03 020 2

Dosen Pembimbing  
Dr. Ir. Arman Djohan, M.Eng

## **RANCANG BANGUN PEMANCAR OFDM DENGAN SIMULINK**

### **ABSTRAK**

SIMULINK adalah perangkat lunak sub-program dari MATLAB. SIMULINK digunakan secara luas oleh peneliti di seluruh dunia untuk membuat model dan simulasi dari suatu sistem yang dikehendaki. Pembuatan model pada SIMULINK dilakukan dengan menyusun blok-blok yang tersedia pada program ini. Selain itu, masing-masing blok harus diatur parameternya agar bekerja sesuai model yang hendak dibuat.

OFDM adalah salah satu bentuk khusus dari FDMA. Pada OFDM, penggunaan spektrum frekuensi menjadi sangat efektif karena antar satu *sub-band* dengan yang lain dapat saling berpotongan tanpa mengalami ISI dan IFI. Hal ini dikarenakan pada OFDM sinyal-sinyal yang dibentuk saling orthogonal satu dengan yang lain. Seperti halnya teknologi FDMA yang lain, OFDM juga tahan terhadap gangguan dari *frequency selective fading*.

Pada skripsi ini dibahas mengenai pembuatan rancang bangun dari pemancar OFDM pada SIMULINK. Analisa dilakukan pada bentuk gelombang keluaran dari model pada SIMULINK dan variabel-variabel yang terukur pada MATLAB.

**Kata Kunci : FDMA, OFMD, pemancar, Matlab, Simulink, orthogonal**

Benny Sahat Monang  
Departemen Teknik Elektro  
NPM 04 04 03 020 2

Counsellor  
Dr. Ir. Arman Djohan, M.Eng

## **RANCANG BANGUN PEMANCAR OFDM DENGAN SIMULINK**

### **ABSTRACT**

*SIMULINK is sub-program of MATLAB. SIMULINK is widely used by researcher around the world to build a model and simulation of a system. Model building in SIMULINK by arrange block in this program. Beside that, each block has to be set in parameter to work properly as a model we want.*

*OFDM is a special case of FDMA. In OFDM, the using of frequency spectrum is very effective because overlapping between sub-band without with no ISI and IFI. This is caused in OFDM, signals generated is orthogonal one another. Jus as other scheme of FDMA, OFDM is also resistant to frequency selective fading.*

*This research in making prototype of OFDM transmitter on SIMULINK. Analyzing made to wave form of the output of model in SIMULINK and variables which is measured by MATLAB.*

**Keywords : FDMA, OFMD, transmitter, Matlab, Simulink, orthogonal**

## DAFTAR ISI

	halaman
PERNYATAAN KEASLIAN SKRIPSI	ii
PERSETUJUAN	iii
UCAPAN TERIMA KASIH	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
DAFTAR LAMPIRAN	xii
DAFTAR ISTILAH	xiii
BAB I PENDAHULUAN	1
1.1 LATAR BELAKANG	1
1.2 PERUMUSAN MASALAH	2
1.3 TUJUAN PENELITIAN	2
1.4 BATASAN MASALAH	3
1.5 METODOLOGI PENELITIAN	3
1.6 SISTEMATIKA PENULISAN	3
BAB II TEKNOLOGI <i>ORTHOGONAL FREQUENCY</i>	5
<i>DIVISION MULTIPLEXING (OFDM)</i>	
2.1 TEKNIK MULTIPLE ACCESS	5
2.1.1 Time Division Multiple Access (TDMA)	5
2.1.2 Frequency Division Multiple Access (FDMA)	6
2.1.3 Code Division Multiple Access (CDMA)	6
2.1.4 Space Division Multiple Access (SDMA)	7
2.2 SINYAL ORTHOGONAL	7
2.3 QUADRATURE AMPLITUDE MODULATION (QAM)	11

2.4	SINYAL OFDM	12
2.5	PENGGUNAAN CYCLIC PREFIX	17
BAB III RANCANG BANGUN PEMANCAR OFDM		23
3.1	SIMULINK®	23
3.2	BLOK DIAGRAM PEMANCAR OFDM PADA SIMULINK®	24
3.2.1	Blok "Source"	24
3.2.2	Blok "OFDM Modulation"	25
3.2.2.1	Blok "General QAM Modulator Baseband"	26
3.2.2.2	Blok "Multiport Selector"	28
3.2.2.3	Blok "Matrix Concatenation"	29
3.2.3	Blok "IFFT"	30
3.2.4	Blok "Add Cyclic Prefix"	30
3.3	DIGITAL SIGNAL PROCESSOR STARTER KIT TMS320C6713	31
3.4	IMPLEMENTASI BLOK SIMULINK PADA TMS320C6713	33
BAB IV UJI COBA DAN ANALISA MODEL PEMANCAR OFDM		41
4.1	PROSEDUR UJI COBA DAN PENGAMBILAN DATA	41
4.2	PENGAMBILAN DAN ANALISA DATA PADA MATLAB	42
4.2.1	Variabel 'Integer'	44
4.2.2	Variabel 'Mapping'	44
4.2.3	Variabel 'Modulasi'	46
4.2.4	Variabel 'IFFT'	50
4.2.5	Variabel 'ACP'	51
4.3	PENGAMBILAN DAN ANALISA DATA PADA OSILOSKOP	52
BAB V KESIMPULAN		55
DAFTAR ACUAN		56
DAFTAR PUSTAKA		57
LAMPIRAN		58

## DAFTAR GAMBAR

		Halaman
Gambar 2.1	TDMA dengan <i>guard time</i> diantara masing-masing <i>time slot</i>	5
Gambar 2.2	FDMA dengan <i>Guard Band</i> pada masing-masing <i>Frequency Band</i>	6
Gambar 2.3	Frequency-hop multiple Access (CDMA)	7
Gambar 2.4	$\alpha_i$ dapat dilihat sebagai besar vektor $\mathbf{x}$ pada arah $\mathbf{v}_i$	9
Gambar 2.5	<i>M</i> -QAM <i>constellation</i> untuk $M = 4$ (a), 16 (b), 64 (c)	12
Gambar 2.6	FDMA konvensional	13
Gambar 2.7	spektrum frekuensi dari OFDM yang saling berpotongan	14
Gambar 2.8	Penggunaan IFFT pada pembentukan sinyal OFDM	16
Gambar 2.9	Perbandingan penggunaan frekuensi pada (a) FDMA dan (b) OFDM	17
Gambar 2.10	Penggunaan <i>Cyclic Prefix</i> pada simbol	18
Gambar 2.11	OFDM dengan cyclic prefix: (a) pulsa pemancar (b) pulsa detektor	20
Gambar 2.12	sinyal dan echo	21
Gambar 3.1	cara-cara menjalankan SIMULINK <sup>®</sup>	23
Gambar 3.2	Blok diagram pemancar OFDM	24
Gambar 3.3	pengaturan pada blok "source"	25
Gambar 3.4	Blok diagram pada blok "OFDM modulation"	26
Gambar 3.5	parameter blok QAM <i>Modulator</i>	27
Gambar 3.6	parameter blok <i>Multiport Selector</i>	29
Gambar 3.7	parameter blok "Matrix Concatenation"	30
Gambar 3.8	parameter blok <i>Add Cyclic Prefix</i>	31
Gambar 3.9	papan TMS320C6713	32
Gambar 3.10	blok diagram dari TMS320C6713	32

Gambar 3.11	algoritma penanaman model ke dalam TMS320C6713 sebagai <i>embedded target</i>	34
Gambar 3.12	penambahan blok C6713 DSK DAC	34
Gambar 3.13	parameter blok C6713 DSK DAC	35
Gambar 3.14	subsystem <i>sine wave generator</i> yang terdiri dari beberapa blok	35
Gambar 3.15	parameter blok sine wave 1 dan 2	36
Gambar 3.16	penambahan blok C6713 DSK	37
Gambar 3.17	pada tab menu “ <i>Real Time Workshop</i> ”	37
Gambar 3.18	pada tab menu “Solver”	38
Gambar 3.19	pada tab menu “Hardware Implementation”	38
Gambar 3.20	pada tab menu “Link for CCS”	39
Gambar 3.21	tombol incremental build	40
Gambar 4.1	parameter blok ‘ <i>to workspace</i> ’	42
Gambar 4.2	pencantuman blok-blok ‘ <i>to workspace</i> ’ pada modul utama	43
Gambar 4.3	pencantuman blok ‘ <i>to workspace</i> ’ pada <i>subsystem</i> OFDM modulation	43
Gambar 4.4	spektrum frekuensi pada model dengan $K = 192$ dan FFT length $N = 256$	47
Gambar 4.5	spektrum frekuensi pada model dengan $K = 192$ dan FFT length $N = 512$	48
Gambar 4.6	sampel pertama bentuk gelombang keluaran LINE OUT DSK	51
Gambar 4.7	sampel kedua bentuk gelombang keluaran LINE OUT DSK	52
Gambar 4.8	sinyal OFDM diukur dengan MATLAB	53

## DAFTAR TABEL

	Halaman	
Tabel 3.1	pemetaan pada modulasi 64-QAM	19
Tabel 3.2	urutan pembagian sinyal pada multiplex selector	19
Tabel 4.1	20 data pertama dari variabel 'integer'	26
Tabel 4.2	20 nilai mapping pertama dan bilangan integer yang bersesuaian	27
Tabel 4.3	data variabel 'modulasi' pada elemen ke-24 sampai ke-43	28
Tabel 4.4	variabel 'IFFT' pada elemen ke-24 sampai ke-43	32

## DAFTAR LAMPIRAN

		Halaman
Lampiran 1	Tabel Aliran Data untuk $\Delta t = 1$ detik	37
Lampiran 2	Tabel Perbandingan FFT dengan Modulasi untuk $\Delta t = 1$ detik	45

## DAFTAR ISTILAH

1. *Multipath fading*: Fenomena variasi arah dari sinyal-sinyal yang ditransmisikan akibat terjadinya pantulan atau pembiasan.
2. *Multiple Access User*: Pengguna-pengguna yang dapat melakukan akses secara bersamaan.
3. *FDMA*: Teknologi *multiple access* yang menabagi informasi ke dalam beberapa *sub-band*.
4. *OFDM*: Bentuk khusus dari FDMA dimana masing-masing *sub-band* saling orthogonal satu dengan yang lainnya.
5. *Frame*: Gabungan dari beberapa *sample*. Pengiriman sebuah *frame* ekuivalen dengan pengiriman beberapa *sample* secara bersamaan (serempak).
6. *Sample Time*: Waktu cuplik *sample*.
7. *Data rate*: Jumlah bit data yang dikirimkan per detik.

# BAB I

## PENDAHULUAN

### 1.1 LATAR BELAKANG

Kebutuhan akan saluran komunikasi yang baik telah membawa kita pada penggunaan spektrum frekuensi yang lebar. Penggunaan spektrum frekuensi yang besar memungkinkan para pengguna saluran informasi untuk mendapatkan akses informasi yang cepat. Dengan semakin berkembang pesatnya kebutuhan akan saluran komunikasi, maka kebutuhan pita frekuensi yang diperlukan juga semakin lebar. Hal ini terbentur dengan terbatasnya rentang frekuensi yang tersedia terutama pada komunikasi nirkabel.

Cara yang cukup baik untuk memanfaatkan rentang frekuensi yang terbatas agar dapat tetap dapat digunakan oleh banyak pengguna (*multi user*) adalah penggunaan teknik *multiple access*. Dengan menggunakan teknik ini, kita dapat membagi-bagi saluran komunikasi dengan lebar tertentu kepada banyak pengguna. Sehingga saluran komunikasi yang tersedia terbatas dapat digunakan secara bersamaan secara real time oleh banyak pengguna pada waktu yang sama pula. Ada beberapa teknik yang digunakan dalam *multiple access*. Salah satu diantaranya adalah Frequency Division Multiple Access (FDMA), selain juga teknik *multiple access* lain yang cukup dikenal seperti CDMA, TDMA, dan SDMA.

Pada FDMA, satu rentang frekuensi tertentu dibagi ke dalam beberapa rentang frekuensi yang lebih kecil sesuai dengan jumlah pengguna yang menggunakan saluran tersebut. Dengan demikian masing-masing pengguna menggunakan satu rentang frekuensi tetap yang adalah bagian dari keseluruhan lebar-pita pada spektrum frekuensi yang digunakan. Namun teknik ini pun terkendala karena tidak optimal menggunakan rentang frekuensi yang tersedia.

(*Orthogonal Frequency Division Multiplexing*) OFDM adalah bentuk khusus dari FDMA yang dapat menggunakan rentang frekuensi yang tersedia dengan optimal. Teknik ini memungkinkan penggunaan lebar pita secara keseluruhan.

Ketangguhan dari teknik ini terletak pada terminologi orthogonal yang memungkinkan masing-masing sub-band saling berpotongan tanpa mengalami interferensi satu dengan lain. Selain itu teknik ini juga membuat informasi yang dipancarkan lebih tahan terhadap *frequency selective fading* dan juga dapat menyalurkan informasi dengan kecepatan yang melebihi lebar pita yang digunakan.

Teknologi pemancar OFDM yang diterapkan pada Digital Signal Processing Starter Kit TMS320C6713 ini dibuat selain untuk mempelajari teknologi OFDM tetapi juga untuk membuat sebuah rancang bangun dari pemancar OFDM yang dapat digunakan sebagai modulator, modem, dan lain sebagainya dengan pengembangan yang lebih lanjut. Pembuatan dari pemancar OFDM ini dibuat dengan papan DSK TMS320C6713 dengan terlebih dahulu membuat model simulasi pemancar OFDM pada SIMULINK. Setelah model pada SIMULINK selesai dibuat, maka model telah siap untuk diterapkan pada DSK. Penerapan model pada DSK tidak membutuhkan pemrograman ulang karena SIMULINK akan membuat kode bahasa pemrograman C untuk dapat diterapkan dalam DSK. Kode bahasa pemrograman ini akan dijalankan dalam perangkat lunak Code Composer Studio (CCS) dan dari perangkat lunak ini kemudian program dijalankan pada DSK.

## **1.2 PERUMUSAN MASALAH**

Masalah yang dibahas dalam adalah pembuatan rancang bangun pengirim OFDM pada DSK TMS320C6713. Pembuatan dimulai dengan membuat model pada SIMULINK untuk kemudian diterapkan pada DSK dengan menggunakan perangkat lunak tambahan Code Composer Studio. Beberapa penyesuaian harus dilakukan agar blok yang dibuat dapat diimplementasikan pada DSK.

## **1.3 TUJUAN PENELITIAN**

Tujuan penelitian ini adalah untuk mendapatkan rancang bangun dari pemancar OFDM pada DSK TMS320C6713 dengan berbantuan SIMULINK.

## **1.4 BATASAN MASALAH**

Penelitian ini hanya bertujuan untuk membuat sebuah rancang bangun dari pemancar OFDM dan bukan mencari optimasi sinyal dari pemancar OFDM yang dibuat. Penelitian ini juga tidak membahas ketahanan sinyal OFDM terhadap rupa-rupa gangguan. Selain itu penelitian ini juga tidak membahas tentang penerima OFDM.

## **1.5 METODOLOGI PENELITIAN**

Metodologi yang digunakan adalah merancang pemancar OFDM di dalam laboratorium. Hal pertama yang dilakukan adalah mempelajari teknologi OFDM dan kemudian mempelajari cara pembentukan sinyal OFDM. Penelitian kemudian dilanjutkan dengan membangun sinyal OFDM pada blok model simulasi pada SIMULINK. Setelah sinyal OFDM yang dikehendaki didapatkan, maka selanjutnya adalah menerapkan model simulasi pada DSK TMS320C6713. Penerapan pada DSK TMS320C6713 dilakukan dengan bantuan perangkat lunak tambahan yaitu Code Composer Studio (CCS).

## **1.6 SISTEMATIKA PENELITIAN**

### **BAB I            PENDAHULUAN**

Bab ini membahas tentang latar belakang penulisan, perumusan masalah, tujuan penulisan, batasan masalah, metodologi penelitian, dan sistematika penulisan skripsi untuk memberikan gambaran umum mengenai permasalahan yang dibahas dalam skripsi ini.

### **BAB II            TEKNOLOGI    *ORTHOGONAL    FREQUENCY    DIVISION* *MULTIPLEXING* (OFDM)**

Pada bab ini dibahas tentang dasar teori dan pengetahuan-pengetahuan dasar yang diperlukan untuk dapat memahami teknologi *Orthogonal Frequency Division Multiplexing* (OFDM).

### **BAB III           RANCANG BANGUN PEMANCAR OFDM**

Bab ini membahas pembuatan rancang bangun pengirim OFDM baik pada perangkat lunak SIMULINK serta penerapannya pada perangkat keras DSK TMS320C6713.

#### **BAB IV UJI COBA DAN ANALISA MODEL PEMANCAR OFDM**

Bab ini berisi uji coba dan analisa dari model pemancar OFDM yang telah dibuat pada SIMULINK dan diterapkan pada DSK TMS320C6713.

#### **BAB V KESIMPULAN**

Pada bab ini berisi kesimpulan dari keseluruhan penelitian rancang bangun pemancar OFDM ini.

## BAB II

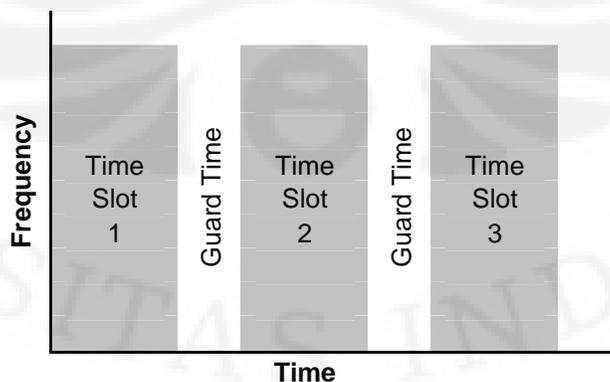
### TEKNOLOGI *ORTHOGONAL FREQUENCY DIVISION* *MULTIPLEXING (OFDM)*

#### 2.1 TEKNIK *MULTIPLE-ACCESS*

*Multiple-access* adalah sebuah teknik dimana banyak *subscriber* menggunakan saluran komunikasi secara bersama. Ada beberapa cara yang digunakan dalam *multiple-access* untuk dapat membagi saluran komunikasi agar dapat dipakai secara bersamaan oleh banyak pengguna (*subscriber*). Paling tidak ada empat cara yang dapat digunakan dalam teknik *multiple-access* antara lain: *Time Division Multiple Access (TDMA)*, *Code Division Multiple Access (CDMA)*, *Frequency Division Multiple Access (FDMA)*, dan *Space Division Multiple Access (SDMA)*.

##### 2.1.1 *Time Division Multiple Access (TDMA)*

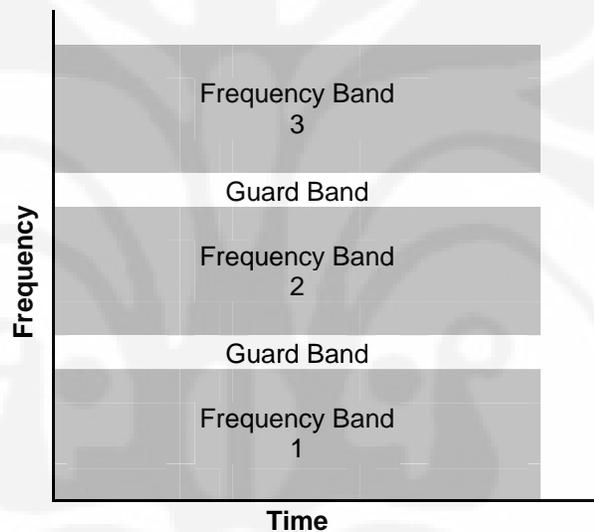
Pada teknik ini, masing-masing pengguna menggunakan seluruh spektrum frekuensi dari saluran (*channel*) tetapi hanya dalam waktu yang sangat singkat. Lamanya waktu yang digunakan oleh satu pengguna disebut sebagai *time slot*. Gambar 2.1 memperlihatkan grafik frekuensi terhadap waktu dari TDMA dengan tiga *time slot*. Pada gambar tersebut kita melihat terdapat *Guard Time* yang berfungsi sebagai pemisah antara masing-masing *time slot* agar tidak terjadi interferensi pada masing-masing *time slot*.



**Gambar 2.1** TDMA dengan *guard time* diantara masing-masing *time slot*

### 2.1.2 Frequency Division Multiple Access (FDMA)

Pada teknik ini, lebar pita frekuensi yang digunakan dibagi menjadi beberapa bagian. Masing-masing pengguna menggunakan satu lebar frekuensi tertentu yang telah ditentukan. Lebar pita yang tersedia dibagi menjadi lebar pita yang lebih kecil yang kemudian dilakukan oleh masing-masing pengguna. Pada Gambar 2.2 diperlihatkan grafik dari frekuensi vs waktu dari FDMA. Pada gambar tersebut kita melihat adanya *guard band* yang berfungsi untuk mencegah interferensi antara dua pita frekuensi yang berbeda.



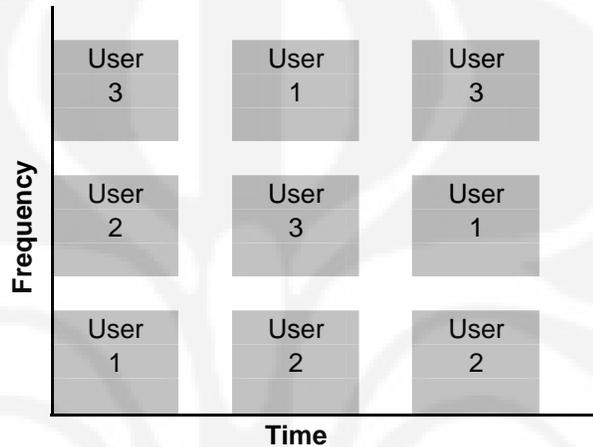
**Gambar 2.2** FDMA dengan *Guard Band* pada masing-masing *Frequency Band*

OFDM adalah salah satu bentuk khusus dari FDMA. Pada OFDM, masing-masing pita frekuensi saling berpotongan dan tidak ada *guard band*. Dengan demikian, frekuensi yang digunakan menjadi lebih efektif karena tidak ada bagian dari keseluruhan pita frekuensi yang tidak digunakan. Syarat utama dari OFDM adalah masing-masing sinyal harus saling orthogonal agar tidak terjadi interferensi pada pita frekuensi yang berbeda.

### 2.1.3 Code Divison Multiple Access (CDMA)

Teknik lain untuk membagi saluran komunikasi adalah dengan cara CDMA. Pada teknik ini, masing-masing pengguna memakai saluran komunikasi pada pita

frekuensi yang sama dalam waktu yang bersamaan pula. Yang membedakan antara pengguna yang satu dengan yang lainnya adalah pengkodean dari masing-masing pengguna. Gambar 2.3 memperlihatkan grafik dari frekuensi vs waktu dari CDMA. Pada gambar tersebut terlihat bahwa CDMA dapat dipandang sebagai kombinasi hybrid dari TDMA dan FDMA.



**Gambar 2.3** Frequency-hop multiple Access (CDMA)

#### 2.1.4 Space Division Multiple Access (SDMA)

Pada teknik *multiple-access* ini, pembagian sumber daya dilakukan dengan membedakan pengguna secara ruang. Biasanya, antena *multibeam* digunakan untuk menjangkau masing-masing pengguna berdasarkan wilayahnya. Dengan menggunakan *multibeams antenna*, pengguna dibedakan berdasarkan letaknya untuk kemudian menggunakan pita frekuensi yang sama pada saat yang bersamaan pula pada sebuah saluran komunikasi yang sama.

## 2.2 SINYAL ORTHOGONAL

Untuk dapat memahami konsep Orthogonal Frequency Division Multiplexing, kita harus mengerti pengertian dari kata orthogonal. OFDM adalah salah satu bentuk khusus dari FDMA dimana masing-masing *sub-band* dapat saling berpotongan satu sama lain tanpa mengalami gangguan saling interfensi. Ini karena pembawa dari masing-masing sub-band saling orthogonal sehingga tidak saling mempengaruhi satu sama lain.

Sebuah sinyal dapat diinterpretasikan sebagai sebuah vektor. Seperti halnya sebuah vector, sinyal dapat dioperasikan dengan operasi-operasi vektor baik berupa penambahan, perkalian baik dengan skalar ataupun dengan vektor lainnya. Dengan demikian, sinyal memenuhi sifat matematis dari ruang vektor.

Bila kita mempunyai sebuah sinyal  $x(t)$  dalam selang waktu  $0 \leq t \leq T$  dan  $x(t) = 0$  di tempat lain. Maka dengan menggunakan transformasi Fourier kita dapat menuliskannya sebagai:

$$x(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos\left(2\pi \frac{k}{T} t\right) - \sum_{k=1}^{\infty} b_k \sin\left(2\pi \frac{k}{T} t\right) \quad (2.1)$$

Dengan  $a_k$  dan  $b_k$  adalah koefisien Fourier dengan:

$$a_k = \frac{2}{T} \int_0^T \cos\left(2\pi \frac{k}{T} t\right) x(t) dt \quad (2.2)$$

dan

$$b_k = -\frac{2}{T} \int_0^T \sin\left(2\pi \frac{k}{T} t\right) x(t) dt \quad (2.3)$$

dimana  $T$  adalah periode dari sinyal,  $t$  adalah waktu dan  $k$  adalah bilangan bulat ( $k = 0, 1, 2, 3, 4, \dots$ ). Dengan melihat dari persamaan diatas, kita dapat menuliskan kembali  $k/T$  sebagai  $f_k$  dimana  $f_k = k/T$  adalah frekuensi dari gelombang sinusoidal (sin dan cos).

Koefisien Fourier adalah amplitude dari gelombang cosinus dan (negatif) sinus dari frekuensi  $f_k$ . Gelombang cosinus dan (negatif) sinus adalah basis dari transformasi Fourier dalam selang waktu  $T$ . Setiap sinyal dapat dituliskan sebagai kombinasi linear dari basis-basisnya. Bentuk transformasi Fourier diatas dapat kita tuliskan kembali dalam bentuk umum bahwa sebuah fungsi dibangun dari kombinasi linear dari basis-basisnya dengan memandang sinyal  $x(t)$  sebagai vektor berdimensi  $N$  ( $\mathbf{x} \in \mathbb{R}^N$ ) dalam vector-vector basis  $\{\mathbf{v}_i\}_{i=1}^N$  sebagai berikut:

$$\mathbf{x} = \sum_{i=1}^N \alpha_i \mathbf{v}_i \quad (2.4)$$

koefisien  $\alpha_i$  diperoleh dengan:

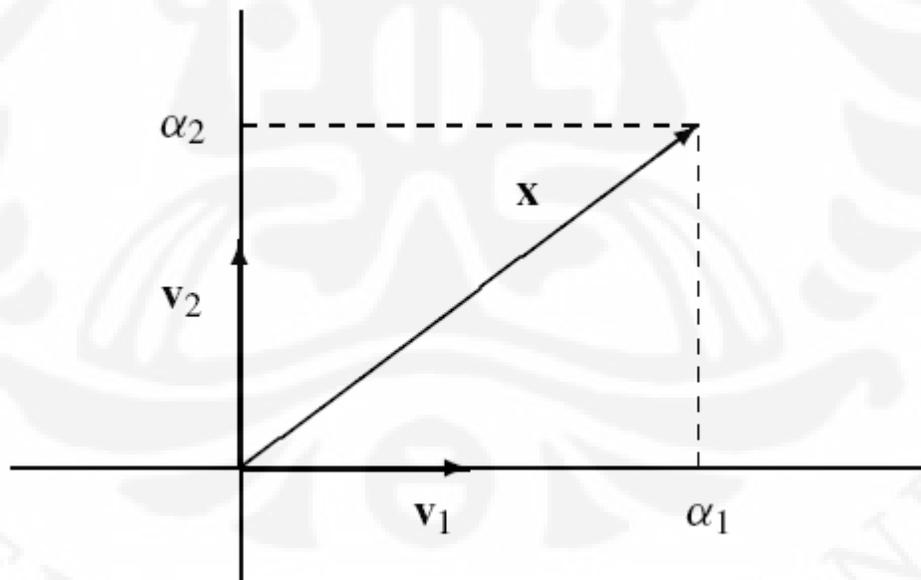
$$\alpha_i = \mathbf{v}_i \cdot \mathbf{x} \quad (2.5)$$

Basis-basis dari vektor yang digunakan disebut orthonormal jika antara dua basis vektor yang berbeda saling orthogonal satu sama lain. Dengan demikian, jika masing-masing basis vektor kita normalisasikan, maka perkalian titik dari dua basis vector yang berbeda adalah nol. Sedangkan perkalian titik dari dua basis vektor yang sama memberikan nilai 1. Atau dapat dituliskan:

$$\mathbf{v}_i \cdot \mathbf{v}_k = \delta_{ik} \quad (2.6)$$

dimana  $\delta_{ik}$  adalah Kronecker Delta ( $\delta_{ik} = 0$  untuk  $i \neq k$  dan  $\delta_{ik} = 1$  untuk  $i = k$ ).

Untuk basis vektor yang orthonormal,  $\alpha_i$  dapat dilihat sebagai proyeksi dari vektor  $\mathbf{x}$  pada masing-masing basis vektornya seperti dapat dilihat pada Gambar 2.4 untuk  $N = 2$ .



**Gambar 2.4**  $\alpha_i$  dapat dilihat sebagai besar vektor  $\mathbf{x}$  pada arah  $\mathbf{v}_i$

Transformasi Fourier menguraikan sinyal ke dalam fungsi basisnya yaitu dalam bentuk cosinus dan (negatif) sinus dengan frekuensi  $f_k = k/T$  dimana  $k$  adalah bilangan bulat sehingga untuk gelombang cosinus dan (negatif) sinus. Gelombang-gelombang tersebut akan saling orthogonal karena frekuensinya saling berkelipatan. Ini berhubungan dengan sifat bahwa dua sinyal sinusoidal saling orthogonal jika salah satu sinyal memiliki frekuensi sebesar kelipatan bulat dari frekuensi sinyal yang lainnya. Sedangkan untuk frekuensi yang sama, sinyal cosinus dan (negatif) sinus saling orthogonal. Karenanya kita dapat melihat transformasi Fourier sebagai penguraian suatu sinyal kepada basis-basis fungsi yang saling orthonormal.

Pada aplikasi nyata, sinyal yang digunakan adalah sinyal kompleks. Jika kita mempunyai sebuah sinyal kompleks  $s(t)$  pada selang waktu  $[0, T]$  dan nol pada daerah lain, maka kita dapat menuliskan transformasi Fourier untuk sinyal kompleks sebagai:

$$s(t) = \sum_{k=-\infty}^{\infty} \alpha_k v_k(t) \quad (2.7)$$

dengan fungsi basis:

$$v_k(t) = \sqrt{\frac{1}{T}} \exp\left(j2\pi \frac{k}{T} t\right) \Pi\left(\frac{t}{T} - \frac{1}{2}\right) \quad (2.8)$$

fungsi-fungsi basisnya saling orthonormal, atau dapat dituliskan:

$$\int_{-\infty}^{\infty} v_i^*(t) v_k(t) dt = \delta_{ik} \quad (2.9)$$

koefisien  $\alpha_k$  didapat dengan:

$$\alpha_k = \int_{-\infty}^{\infty} v_i^*(t) s(t) dt \quad (2.10)$$

Untuk sinyal kompleks, fungsi basis yang dihasilkan dari transformasi Fourier adalah juga orthonormal. Dengan demikian, kita dapat melihat bahwa transformasi Fourier memiliki peranan yang besar dalam pembangkitan sinyal OFDM.

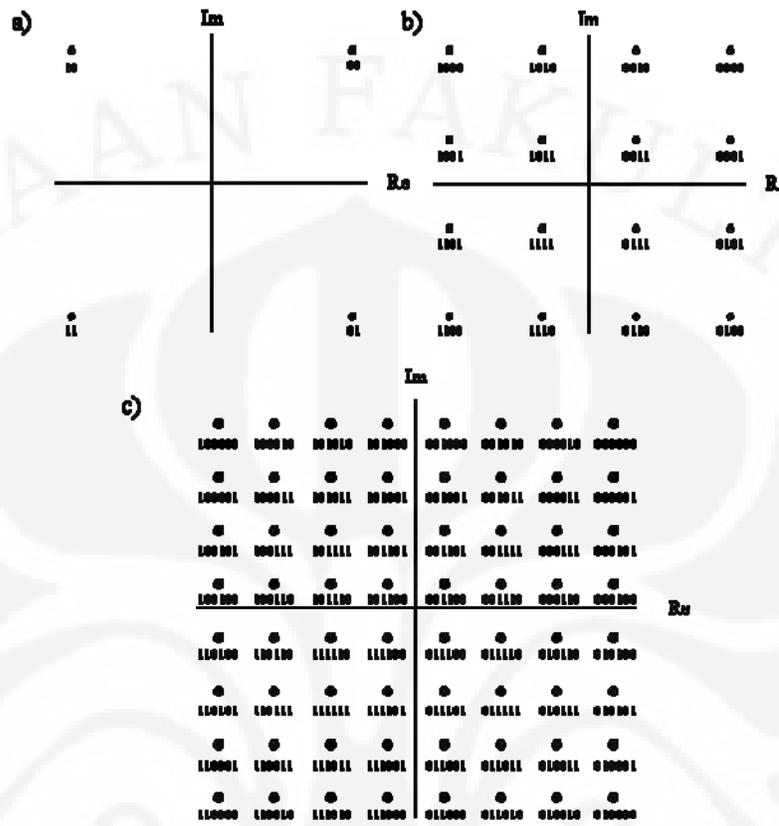
## 2.4 QUADRATURE AMPLITUDE MODULATION (QAM)

Untuk dapat melewati sinyal berupa bit-bit biner melalui saluran transmisi, kita harus memetakan bit-bit tersebut ke dalam sinyal *passband*  $\tilde{s}(t) = \Re\{s(t)e^{j2\pi f_c t}\}$ . Hal ini disebut modulasi digital. Modulasi digital linear berupa sinyal *baseband* kompleks:

$$s(t) = \sum_{k=1}^K s_k g_k(t)$$

dimana informasi dibawa dalam simbol transmit  $s_k$ . Modulasi dikatakan linear karena pemetaan dilakukan secara linear dari vektor  $s = (s_1, \dots, s_k)^T$  ke sinyal transmisi kontinu  $s(t)$ . Secara umum, dikenal beberapa jenis modulasi linear, diantaranya ASK, PSK, dan QAM. Karena bilangan yang dipetakan adalah biner, maka kita dapat memetakan bilangan-bilangan biner tersebut ke  $m$  titik dimana  $m = \log_2(M)$  dengan  $M$  adalah bilangan kelipatan 2.

Pada QAM, kita memetakan bit-bit biner ke dalam sinyal kompleks  $s_k = x_k + jy_k$ . Sinyal kontinu yang dibangkitkan dari modulasi ini mengalami perubahan baik pada amplitude dan juga fasa. Berikut adalah diagram *constellation* dari QAM ditunjukkan pada Gambar 2.5:



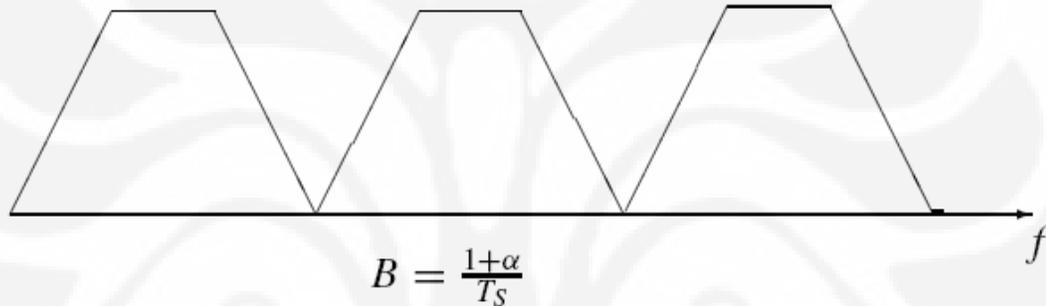
Gambar 2.5 M-QAM constellation untuk  $M = 4$  (a), 16 (b), 64 (c)

## 2.5 SINYAL OFDM

*Orthogonal Frequency Division Multiplexing* adalah bentuk khusus dari FDMA. Jika sebuah lebar pita tunggal yang besar digunakan pada saluran transmisi, maka lebih baik membagi-bagi lebar pita yang tersedia menjadi beberapa *sub-band* dibanding mentransmisikannya sebagai sebuah pita tunggal. Hal ini karena dengan membagi ke dalam beberapa *sub-band*, sinyal yang ditransmisikan menjadi lebih tahan terhadap *frequency selective fading* dibanding ditransmisikan sebagai satu lebar pita tunggal.

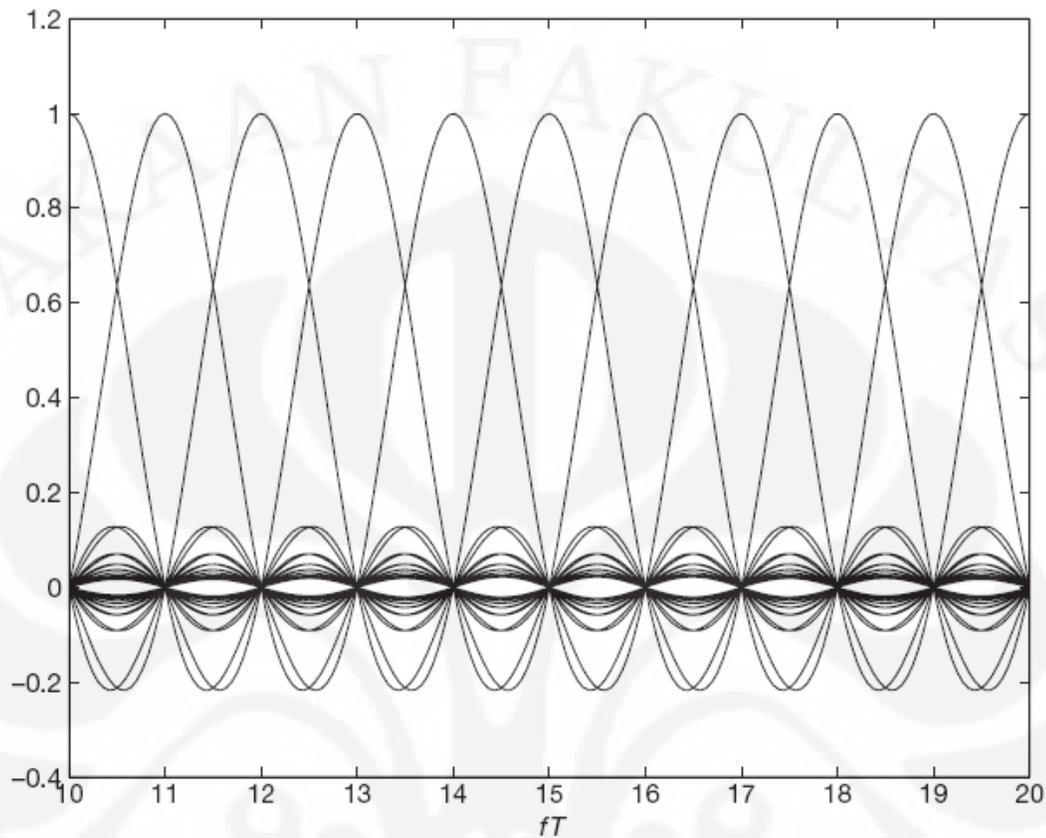
Namun dengan membagi ke dalam beberapa *sub-band*, maka penggunaan lebar frekuensi menjadi tidak maksimal karena kita harus meletakkan *guard band* agar tidak terjadi interferensi antar frekuensi. Misalnya saja jika kita ingin mentransmisikan informasi dengan durasi simbol 1ms ( $T_s = 1$  ms), maka dengan

menggunakan satu lebar pita tunggal, kita membutuhkan lebar pita sebesar 1 GHz ( $B = 1/T_s$ ). Namun jika kita membaginya ke dalam beberapa sub-band, maka diperlukan lebar pita sebesar  $B = (1 + \alpha)T_s^{-1}$  dimana  $\alpha$  adalah rasio antara *guard band* dengan lebar dari *sub-band*. Dengan demikian kita membutuhkan lebar pita yang lebih besar pada FDMA ketimbang ditransmisikan dengan satu lebar frekuensi tunggal. Ini dikarenakan pada FDMA diperlukan adanya guard band untuk menghindari terjadinya interferensi antara *sub-band* yang berbeda seperti ditunjukkan pada Gambar 2.6.



**Gambar 2.6** FDMA konvensional

Cara yang lebih baik menggunakan pita frekuensi yang tersedia adalah dengan membuat masing-masing sub-band saling berpotongan satu dengan yang lain. Gangguan akibat interferensi tidak akan terjadi jika pembawa dari masing-masing sub-band saling orthogonal. Sifat dari dua sinyal yang saling orthogonal adalah kedua sinyal tidak akan saling berinterferensi karena perkalian titik antara keduanya memberikan hasil nol. Ini berarti bahwa masing-masing sinyal yang saling orthogonal bersifat bebas linear. Gambar 2.7 memperlihatkan penggunaan spektrum frekuensi dari sinyal OFDM.



**Gambar 2.7** penggunaan spektrum frekuensi yang saling berpotongan

Persamaan envelope kompleks dari sinyal OFDM adalah sebagai berikut:

$$v(t) = \sqrt{\frac{2E_s}{T_s}} \sum_{k=0}^{\infty} \sum_{n=0}^{N-1} a_{k,n} \tilde{\varphi}_n(t - kT) \quad (2.11)$$

$$v(t) = \sum_{n=0}^{N-1} v_n(t)$$

dimana  $a_{k,n}$  membawa informasi yang akan dikirimkan melalui simbol ke- $k$  pada interval  $t \in [kT, kT + T]$  dan pada *sub-band* ke- $n$  ( $n = 0, 1, 2, \dots, N-1$ ).  $v_n(t)$  adalah envelope kompleks dari sinyal yang ditransmisikan pada sub-band ke- $n$  dengan persamaan:

$$v_n(t) = \sqrt{\frac{2E_s}{T_s}} \sum_{k=0}^{\infty} a_{k,n} \tilde{\varphi}_n(t - kT) \quad (2.12)$$

dimana  $\{\tilde{\varphi}_n(t)\}_{n=0}^{N-1}$  adalah himpunan gelombang orthonormal dengan persamaan:

$$\tilde{\varphi}_n(t) = \begin{cases} \exp\left[j2\pi\left(n - \frac{N-1}{2}\right)t/T\right], & t \in [0, T] \\ 0, & t \notin [0, T] \end{cases} \quad (2.13)$$

masing-masing gelombang pada himpunan  $\{v_n(t)\}_{n=0}^{N-1}$  masing-masing *subcarrier* pada frekuensi  $f_c + \frac{2n - (N-1)}{2T}$ .

Dengan  $k = 0$ , persamaan (2.11) dapat kita sederhanakan menjadi:

$$v(t) = \sqrt{\frac{2E_s}{T_s}} \sum_{n=0}^{N-1} a_{0,n} \exp\left(j \frac{2\pi n t}{NT_s}\right) \exp\left[-j \frac{\pi(N-1)t}{NT_s}\right], \quad 0 \leq t \leq NT_s \quad (2.14)$$

Perhatikan bahwa persamaan  $\exp\left[-j \frac{\pi(N-1)t}{NT_s}\right]$  bukan fungsi dari  $n$  dan dapat dikombinasikan dengan persamaan dari carrier  $\exp(j2\pi f_c t)$ . Dengan demikian envelope kompleks dapat dituliskan:

$$v(t) = \sqrt{\frac{2E_s}{T_s}} \sum_{n=0}^{N-1} a_{0,n} \exp\left(j \frac{2\pi n t}{NT_s}\right), \quad 0 \leq t \leq NT_s \quad (2.15)$$

dengan carrier  $\exp\left[j2\pi\left(f_c - \frac{N-1}{2NT_s}\right)t\right]$ . Dengan sampling  $v(t)$  pada  $t = \ell T_s$

memberikan:

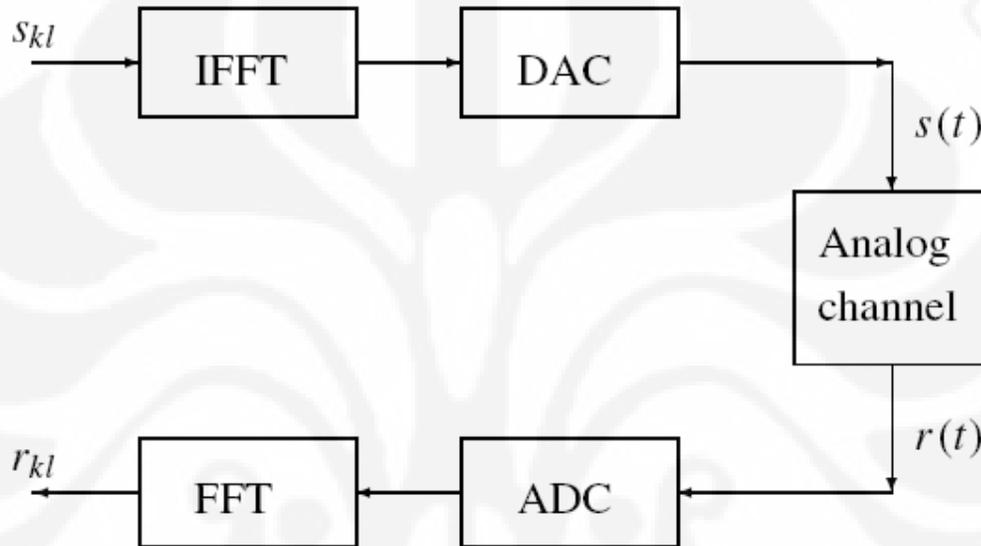
$$A_{0,\ell} \stackrel{\Delta}{=} v(\ell T_s) = \sqrt{\frac{2E_s}{T_s}} \sum_{n=0}^{N-1} a_{0,n} \exp\left(j \frac{2\pi n \ell}{N}\right), \quad \ell = 0, 1, 2, \dots, N-1 \quad (2.16)$$

yang adalah persamaan dari *inverse discrete Fourier transform* (IDFT) dari  $\{a_{0,n}\}$ .

Sampel-sampel,  $\vec{A}_0 = (A_{0,1}, A_{0,2}, \dots, A_{0,N-1})$ , kemudian diteruskan melalui sebuah

konverter *digital-to-analog* (DAC) yang digunakan untuk memodulasi *carrier* dan menghasilkan sinyal OFDM (Gambar 2.8):

$$x(t) = \Re \left\{ v(t) \exp \left[ j2\pi \left( f_c - \frac{N-1}{2NT_s} \right) t \right] \right\} \quad (2.17)$$



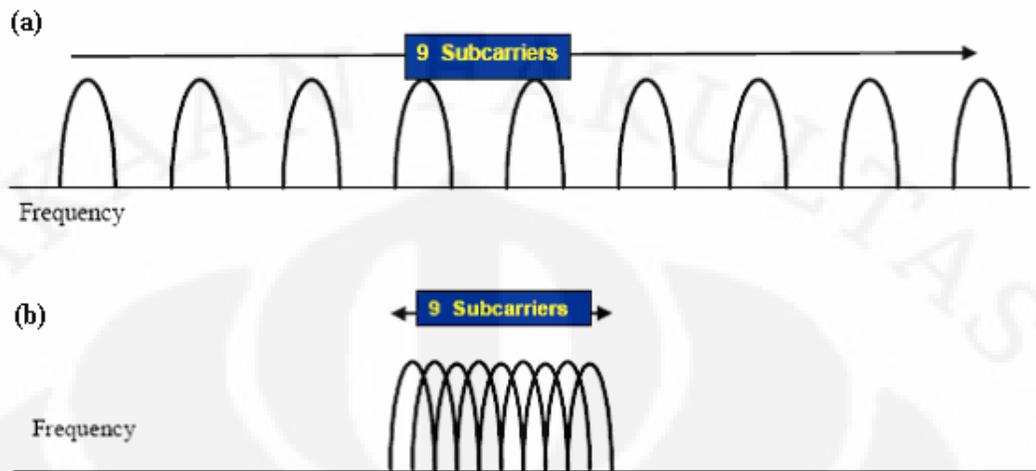
**Gambar 2.8** Penggunaan FFT pada pembentukan sinyal OFDM

Demodulasi dilakukan dengan menggunakan *inverse* dari IDFT yaitu DFT dimana sinyal yang diterima  $\bar{r}_0$  didemodulasikan menurut persamaan:

$$Z_{0,n} = \frac{1}{N} \sum_{\ell=0}^{N-1} r_{0,\ell} \exp \left( -j \frac{2\pi n \ell}{N} \right), \quad n = 0, 1, \dots, N-1 \quad (2.18)$$

Karena sifatnya yang orthogonal, OFDM memiliki beberapa keuntungan diantaranya:

- Efisiensi penggunaan spektrum frekuensi dengan adanya *overlapping* antara masing-masing *sub-band* seperti diperlihatkan pada Gambar 2.9.



**Gambar 2.9** Perbandingan penggunaan frekuensi pada (a) FDMA dan (b) OFDM

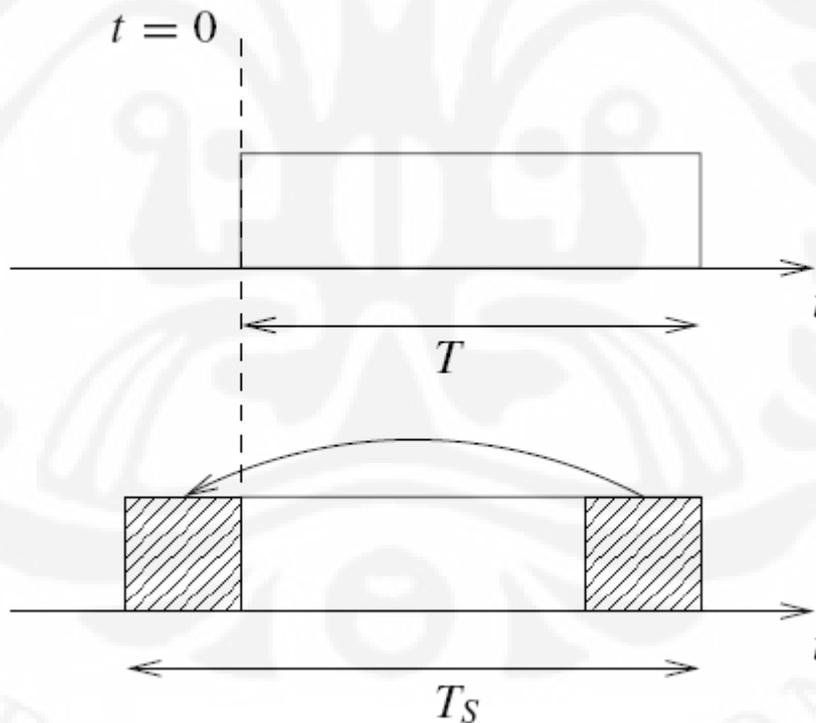
- Dengan membagi saluran ke dalam beberapa *subchannels narrowband*, OFDM dapat lebih tahan terhadap gangguan dari *frequency selective fading* ketimbang sistem *single carrier*.
- Tidak adanya *Inter Symbol Interference* (ISI) dan *Inter Frequency Interference* (IFI) karena menggunakan cyclic prefix.
- Dengan menggunakan channel coding dan interleaving yang baik, simbol yang hilang karena *frequency selective fading* dapat di-recover.
- Pengaturan saluran menjadi lebih sederhana ketimbang menggunakan *adaptive equalization* pada *single carrier*.
- Dimungkinkan untuk menggunakan maximum likelihood decoding dengan menggunakan FFT pada modulasi dan demodulasi.
- Lebih tahan terhadap *cochannel interference* dan *impulsive parasitic noise*.

## 2.6 PENGGUNAAN CYCLIC PREFIX

Pada saluran dengan frequency selective multipath fading, sinkronisasi dapat berpengaruh sangat besar terhadap penyaluran informasi. Hal ini karena *echo* dari masing-masing bagian tidak lagi merupakan sinyal yang sinkron. Dengan tidak lagi

sinkronnya sinyal, maka sinyal OFDM tidak lagi orthogonal sehingga dapat terjadi *Inter Symbol Interference* (ISI) dan *Inter Frequency Interference* (IFI) karena output detektor adalah sinyal yang tertunda  $s_r(t) = s(t - \tau)$  dengan  $0 < \tau < T$ . Penundaan ini yang kemudian menyebabkan ISI dan IFI sehingga sinyal OFDM yang diterima tidak lagi bersifat orthogonal.

Untuk dapat menghindari ISI dan IFI, kita dapat menggunakan *cyclic prefix* (atau biasa juga disebut sebagai *guard interval*). Dengan melakukan hal ini, simbol-simbol akan ditambahkan secara *cyclic* pada gelombang Fourier asli dengan periode  $T$  dan panjang guard interval  $\Delta$  dan menjadi harmonik dengan frekuensi dan fasa yang sama namun durasinya menjadi  $T_s = T + \Delta$ . Hal ini dilakukan dengan menyalin sebagian dari bagian akhir simbol dengan panjang durasi  $\Delta$  ke bagian depan dari simbol sebagaimana kita dapat lihat pada Gambar 2.10.



**Gambar 2.10** Penggunaan *Cyclic Prefix* pada simbol

Jika kita memiliki sebuah sinyal pulsa dasar yang dituliskan dengan:

$$g_k(t) = \sqrt{\frac{1}{T}} \exp\left(j2\pi \frac{k}{T} t\right) \Pi\left(\frac{t}{T} - \frac{1}{2}\right) \quad (2.19)$$

dimana simbol  $\Pi$  menotasikan sinyal *rectangular*. Maka sinyal pulsa dasar baru yang telah ditambahkan cyclic prefix dapat kita tuliskan menjadi:

$$g'_k(t) = \sqrt{\frac{1}{T_s}} \exp\left(j2\pi \frac{k}{T} t\right) \Pi\left(\frac{t + \Delta}{T_s} - \frac{1}{2}\right) \quad (2.20)$$

Perhatikan bahwa kompleks eksponensial masih dalam bentuk yang sama. Frekuensi dari pulsa juga sama  $f_k = k/T$ , dan fasanya juga tetap. Hanya interval dari pulsa rectangular yang bertambah dari  $t \in [0, T)$  menjadi  $t \in [-\Delta, T)$ . Untuk memudahkan, maka kita menormalisasikan faktor eksponensial yang adalah penggambaran dari energi pulsa menjadi satu. Sinyal yang dipancarkan dapat kita tuliskan menjadi:

$$s(t) = \sum_{kl} s_{kl} g'_{kl}(t) \quad (2.21)$$

dengan

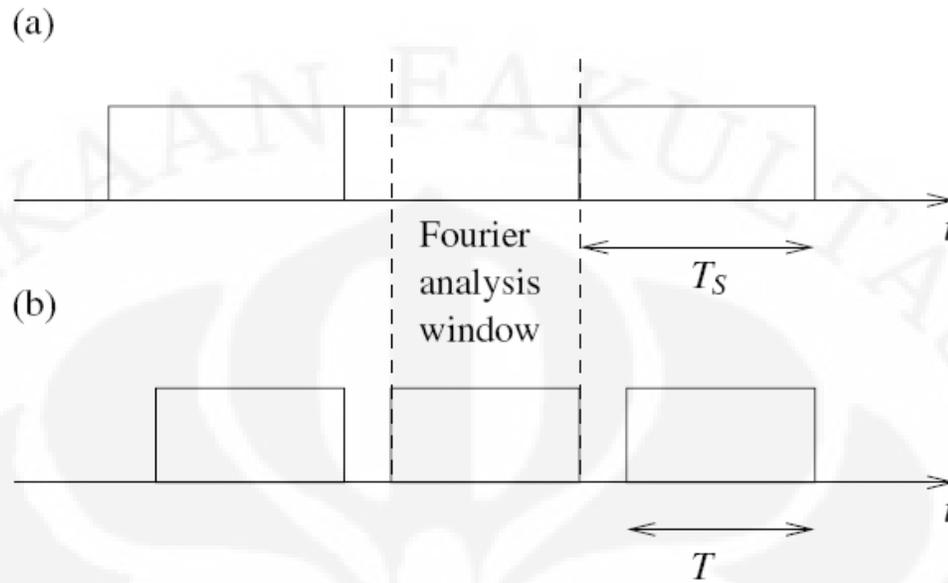
$$g'_{kl}(t) = g'_k(t - lT_s) \quad (2.22)$$

dimana  $l$  adalah bilangan dari *time slot*.

Pulsa yang dipancarkan  $g'_{kl}(t)$  tetap orthogonal satu terhadap yang lain. Pada penerima, kita menggunakan serangkaian detektor yang orthonormal dengan persamaan:

$$g_{kl}(t) = g_k(t - lT_s) \quad (2.23)$$

dimana  $g_k(t)$  adalah fungsi basis dari Fourier pada interval  $T$ . Hal ini berarti analisa Fourier pada penerima bekerja tetap pada panjang *window*  $T$ , namun hal tersebut dilakukan sepanjang durasi  $T_s = T + \Delta$ , dapat dilihat pada Gambar 2.11.



**Gambar 2.11** OFDM dengan cyclic prefix: (a) pulsa pemancar (b) pulsa detektor

Dari gambar diatas kita dapat melihat bahwa penambahan cyclic prefix memperpanjang durasi dari sinyal yang dipancarkan. Namun pada detektor, sinyal yang digunakan pada analisa Fourier memiliki panjang *window* yang sama. Dengan demikian pada penerimaan besaran sinyal yang diterima sama dengan besaran sinyal yang dipancarkan pada pemancar. Dengan cara demikian, kita dapat mengurangi ISI dan IFI.

Jika pulsa yang dikeluarkan pada detektor adalah  $g_{kl}(t)$ , dan pulsa yang dipancarkan adalah  $g'_{k'l',\tau}(t)$ , maka:

$$\langle g_{kl}(t), g'_{k'l',\tau}(t) \rangle = \sqrt{\frac{T}{T_s}} e^{-j2\pi f_k \tau} \delta k k' \delta l l' \quad (2.24)$$

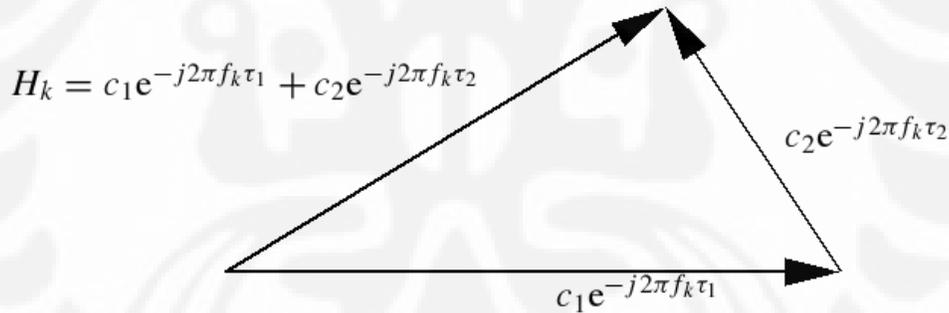
Ini berarti dengan menjaga  $\tau < \Delta$ , sifat orthogonal dari pulsa yang dipancarkan dan yang diterima dapat terus dipertahankan, dan keluaran output untuk indeks yang sama hanya dipengaruhi oleh faktor *frequency-dependent phase*. Sekerang, jika kita melihat sinyal OFDM  $s(t)$  pada Persamaan (2.21) dan menuliskan  $r(t) = s(t - \tau)$  dengan

$0 < \tau < \Delta$  sebagai sinyal yang diterima, yang merupakan sinyal pemancar yang tertunda. Dari Persamaan (2.24) dan (2.21) kita dapatkan:

$$\langle g_{kl}, r \rangle = \sqrt{\frac{T}{T_s}} e^{-j2\pi f_k \tau} s_{kl} \quad (2.25)$$

dimana simbol yang dipancarkan  $s_{kl}$  dikembalikan tanpa ISI, melainkan hanya mengalami perputaran fasa. Faktor perputaran fasa dihilangkan dengan menggunakan demodulasi diferensial.

Jika kita sinyal yang diterima adalah superposisi dari dua sinyal transmisi yang tertunda, dituliskan  $r(t) = c_1 s(t - \tau_1) + c_2 s(t - \tau_2)$  dimana  $c_1$  dan  $c_2$  adalah konstanta kompleks, maka  $s_{kl}$  dapat kembali dikembalikan tanpa ISI jika *delay* yang terjadi tidak lebih panjang dari *guard interval*. Tetapi akan dipengaruhi *complex multiplicative factor*  $H_k = c_1 e^{-j2\pi f_k \tau_1} + c_2 e^{-j2\pi f_k \tau_2}$  seperti terlihat pada Gambar 2.12.



**Gambar 2.12** sinyal dan echo

yang adalah superposisi dari fasor-fasor dua *echo path*. Dengan demikian keluaran detektor dapat kita tuliskan

$$\langle g_{kl}, r \rangle = \sqrt{\frac{T}{T_s}} H_k s_{kl} \quad (2.26)$$

Untuk superposisi dari N echo path, kita dapat menuliskan  $H_k$  menjadi:

$$H_k = \sum_{n=1}^N c_n e^{-j2\pi f_k \tau_n} \quad (2.27)$$

jika kita tuliskan fungsi alih dalam *time-variant*  $H(f,t)$  kita dapatkan:

$$H(f,t) = \int_0^{\Delta} e^{-j2\pi f \tau} h(\tau,t) d\tau \quad (2.28)$$

dan sinyal yang diterima tanpa noise kita tuliskan:

$$r(t) = \int_0^{\Delta} h(\tau,t) s(t - \tau) d\tau \quad (2.29)$$

Jika kita mengasumsikan saluran memiliki *time-variant* yang kecil pada time slot 1, maka kita dapat menuliskan Persamaan (2.28) dimana  $H(f,t) \approx H_l(f)$  dan  $h(\tau,t) \approx h_l(\tau)$  menjadi:

$$H_l(f) = \int_0^{\Delta} e^{-j2\pi f \tau} h_l(\tau) d\tau$$

pada simbol OFDM ke- $l$  selama selang  $T_s$ . Maka kita dapat menghitung keluaran detektor sebagai:

$$\begin{aligned} \langle g_{kl}, h_l * g'_{k'l'} \rangle &= \int_0^T g_{kl}^* dt \int_0^{\Delta} h_l(\tau) g'_{k'l'}(t - \tau) \\ \langle g_{kl}, h_l * g'_{k'l'} \rangle &= \int_0^{\Delta} d\tau h_l(\tau) \int_0^T dt g_{kl}^*(t) g'_{k'l'}(t - \tau) \end{aligned}$$

dari Persamaan (2.24) kita dapatkan:

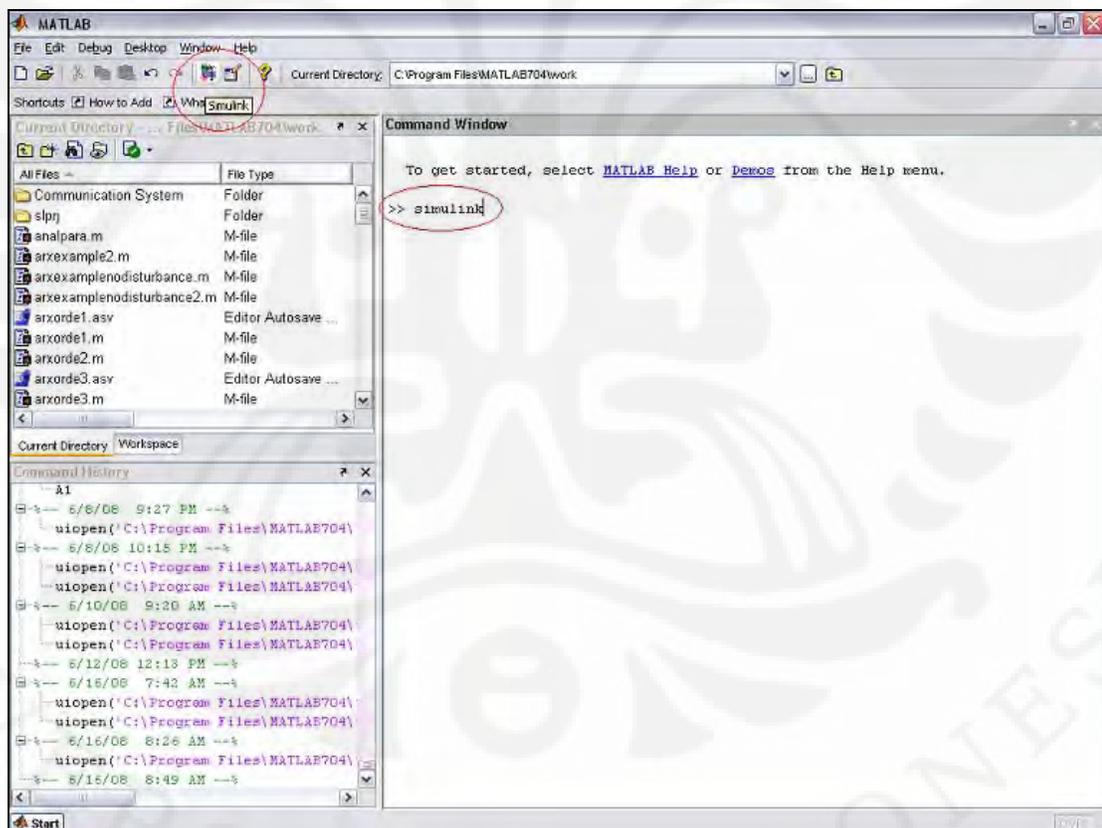
$$\langle g_{kl}, h_l * g'_{k'l'} \rangle = \sqrt{\frac{T}{T_s}} H_l(f_k) s_{kl} \quad (2.30)$$

## BAB III

### RANCANG BANGUN PENGIRIM OFDM

#### 3.1 SIMULINK<sup>®</sup>

SIMULINK<sup>®</sup> adalah perangkat lunak sub-program dari MATLAB<sup>®</sup> yang biasa digunakan untuk pemodelan, simulasi dan analisa sistem. Dengan menggunakan perangkat lunak ini kita dapat membangun blok-blok model untuk mensimulasi sistem yang kita kehendaki. Cara menggunakan SIMULINK<sup>®</sup> hanya dengan mengetikkan perintah "simulink" pada *command window* dari MATLAB<sup>®</sup>. Cara lain adalah cukup dengan menekan tombol simulink pada *toolbar* MATLAB<sup>®</sup> seperti pada Gambar 3.1.



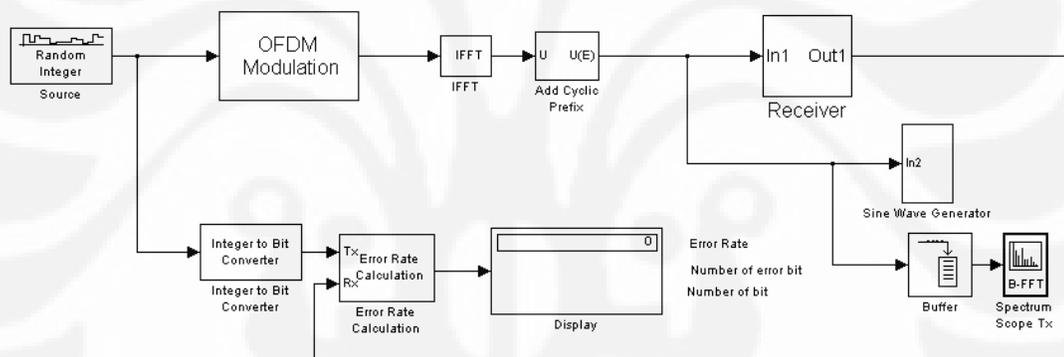
**Gambar 3.1** cara-cara menjalankan SIMULINK<sup>®</sup>

Dengan menjalankan program ini, kita akan diberikan pilihan blok-blok model yang dapat kita gabungkan satu dengan yang lain untuk membentuk simulasi sistem yang kita kehendaki.

Dari SIMULINK<sup>®</sup>, kita juga dapat menghubungkannya dengan perangkat luar seperti DSP TMS320C6713 dengan cara yang mudah. SIMULINK<sup>®</sup> akan membangkitkan secara otomatis kode bahasa pemrograman yang digunakan pada DSP tersebut.

### 3.2 BLOK DIAGRAM PENGIRIM OFDM PADA SIMULINK<sup>®</sup>

Dengan menggunakan SIMULINK<sup>®</sup>, maka dibentuklah blok diagram untuk pemancar OFDM yang ditunjukkan pada Gambar 3.2 berikut:



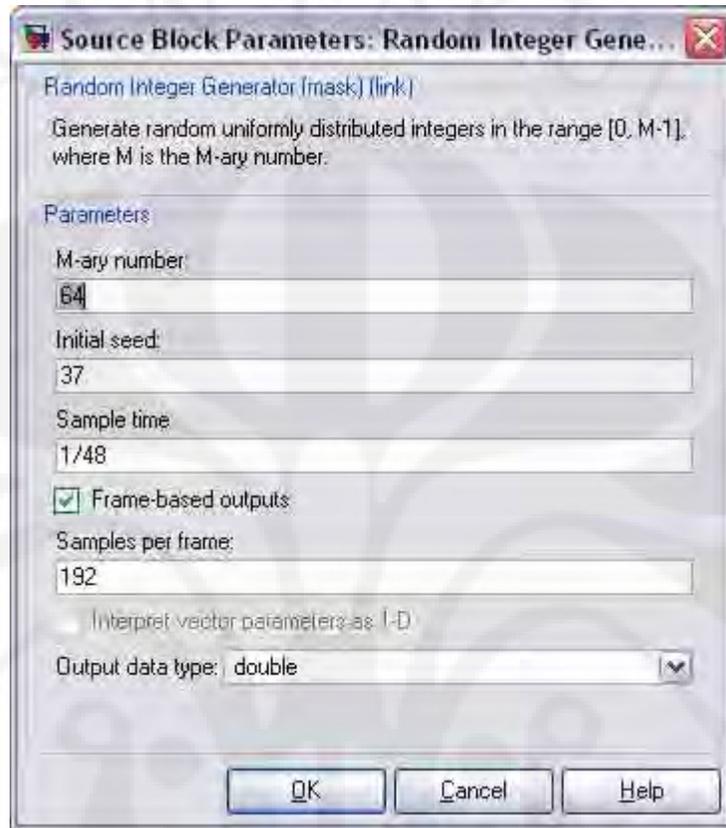
**Gambar 3.2** Blok diagram pemancar OFDM

Pada simulasi ini digunakan teknik OFDM dengan IFFT length 256 dan lebar frame dari informasi adalah 192. Pada skripsi ini, pembahasan hanya akan dilakukan pada sisi transmisi dari OFDM, dengan demikian kita hanya membatasi pembahasan blok yang digunakan pada blok *source* (sumber), *OFDM*, *IFFT*, *add cyclic prefix*.

#### 3.2.1 Blok “Source”

Blok “source” adalah blok yang digunakan sebagai sumber data yang merupakan sumber informasi yang akan dimodulasikan pada pemancar OFDM. Disini digunakan blok *Random Integer Generator* yang akan membangkitkan bilangan integer random. Blok ini membangkitkan bilangan integer terdistribusi

seragam pada rentang 0 sampai M-1. Berikut adalah *setting* yang digunakan pada blok "source" pada simulasi ini (Gambar 3.3):

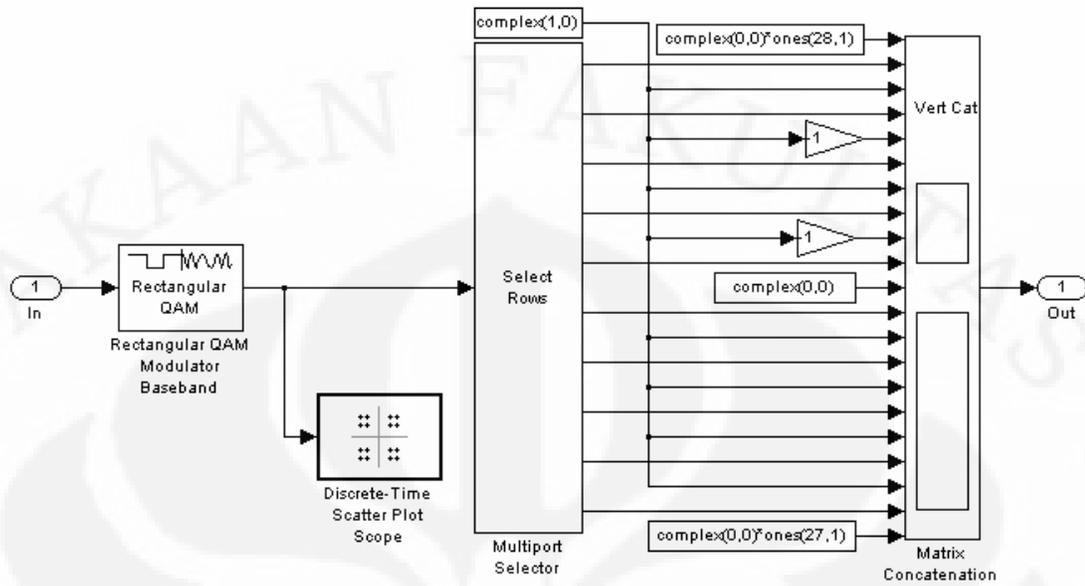


**Gambar 3.3** pengaturan pada blok "source"

Pada parameter 'M-ary number' diisikan nilai 64. Ini karena pada modulasi digital, kita akan menggunakan modulasi 64-QAM. Dengan demikian blok source akan membangkitkan bilangan integer dalam rentang 0 sampai 63 yang kemudian akan dipetakan pada grafik *constellation* dari 64-QAM. Pada parameter sample time diisi dengan 1/192 dan keluaran yang dihasilkan adalah dalam bentuk frame dimana samples per frame bernilai 192.

### 3.2.2 Blok "OFDM Modulation"

Blok "OFDM modulation" ini adalah sebuah *subsystem* yang berfungsi untuk memodulasikan sinyal informasi yang kita miliki dari blok sebelumnya. Blok ini terdiri dari beberapa blok dasar sebagai berikut (Gambar 3.4):



**Gambar 3.4** Blok diagram pada blok "OFDM modulation"

### 3.2.2.1 Blok "General QAM Modulator Baseband"

Modulasi digital yang digunakan adalah 64-QAM dengan parameter blok seperti pada Gambar 3.5 berikut:



**Gambar 3.5** parameter blok QAM Modulator

Berdasarkan parameter yang telah dibuat pada Gambar 3.5 diatas, maka kita mendapatkan koordinat mapping dari bilangan 0 sampai dengan 63 pada modulasi 64-QAM yang digunakan seperti ditunjukkan pada Tabel 3.1.

### 3.2.2.2 Blok "Multiport Selector"

Pada blok "multiport selector" sinyal sumber yang merupakan data frame dengan lebar 192 kemudian dibagi-bagi ke dalam beberapa bagian. Blok "select row" ini mengerjakan bagian serial-to-parallel dari blok diagram OFDM. Urutan pembagian tersebut ditunjukkan pada Tabel 3.2.

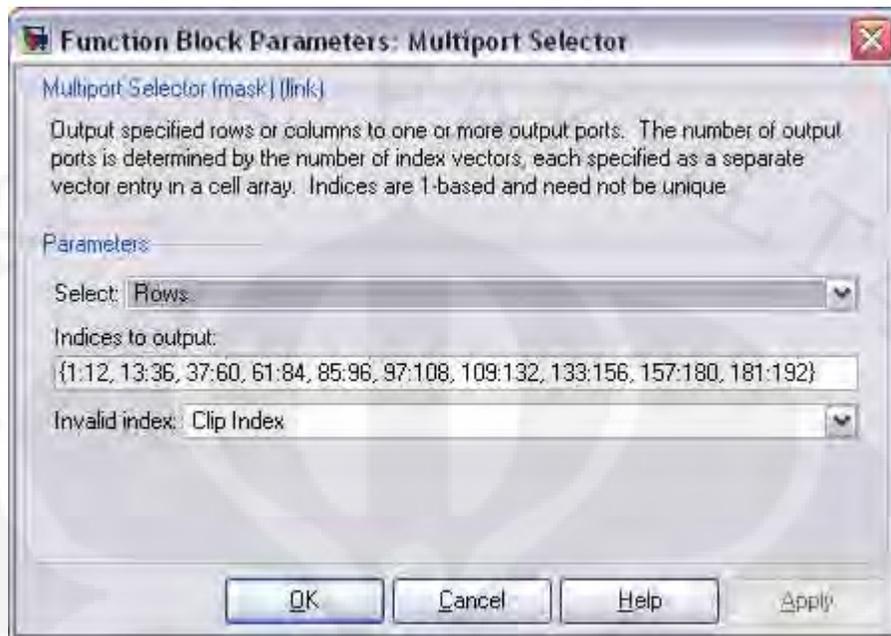
**Tabel 3.1** pemetaan pada modulasi 64-QAM

$-1.05 + 1.05i$	$-1.05 + 0.75i$	$-1.05 + 0.45i$	$-1.05 + 0.15i$
$-1.05 - 0.15i$	$-1.05 - 0.45i$	$-1.05 - 0.75i$	$-1.05 - 1.05i$
$-0.75 + 1.05i$	$-0.75 + 0.75i$	$-0.75 + 0.45i$	$-0.75 + 0.15i$
$-0.75 - 0.15i$	$-0.75 - 0.45i$	$-0.75 - 0.75i$	$-0.75 - 1.05i$
$-0.45 + 1.05i$	$-0.45 + 0.75i$	$-0.45 + 0.45i$	$-0.45 + 0.15i$
$-0.45 - 0.15i$	$-0.45 - 0.45i$	$-0.45 + 0.75i$	$-0.45 - 1.05i$
$-0.15 + 1.05i$	$-0.15 + 0.75i$	$-0.15 + 0.45i$	$-0.15 + 0.15i$
$-0.15 - 0.15i$	$-0.15 - 0.45i$	$-0.15 - 0.75i$	$-0.15 - 1.05i$
$0.15 + 1.05i$	$0.15 + 0.75i$	$0.15 + 0.45i$	$0.15 + 0.15i$
$0.15 - 0.15i$	$0.15 - 0.45i$	$0.15 - 0.75i$	$0.15 - 1.05i$
$0.45 + 1.05i$	$0.45 + 0.75i$	$0.45 + 0.45i$	$0.45 + 0.15i$
$0.45 - 0.15i$	$0.45 - 0.45i$	$0.45 - 0.75i$	$0.45 - 1.05i$
$0.75 + 1.05i$	$0.75 + 0.75i$	$0.75 + 0.45i$	$0.75 + 0.15i$
$0.75 - 0.15i$	$0.75 - 0.45i$	$0.75 - 0.75i$	$0.75 - 1.05i$
$1.05 + 1.05i$	$1.05 + 0.75i$	$1.05 - 0.45i$	$1.05 + 0.15i$
$1.05 - 0.15i$	$1.05 - 0.45i$	$1.05 - 0.75i$	$1.05 - 1.05i$

**Tabel 3.2** urutan pembagian sinyal pada multipoint selector

Bagian	Urutan data
1	1 - 12
2	13 - 36
3	37 - 60
4	61 - 84
5	85 - 96
6	97 - 108
7	109 - 132
8	133 - 156
9	157 - 180
10	181 - 192

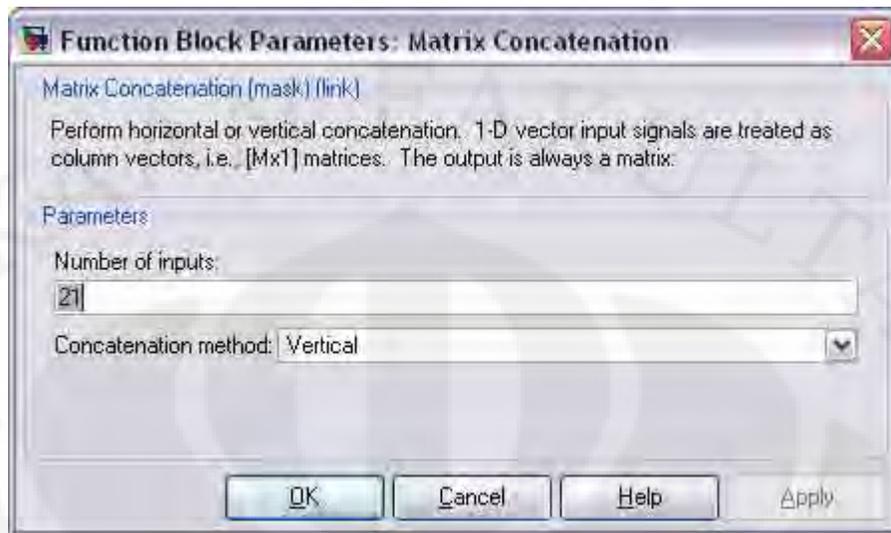
Pembagian tersebut dilakukan dengan memasukkan nilai sesuai dengan Tabel 3.2 pada parameter *Indices to output* seperti dapat kita lihat pada Gambar 3.6.



**Gambar 3.6** parameter blok *Multiport Selector*

### 3.2.2.3 Blok "Matrix Concatenation"

Blok ini mengerjakan bagian dari parallel-to-serial dari modulasi OFDM. Pada Gambar 3.4 kita melihat bahwa masukan pada blok ini tidak hanya data sumber informasi tetapi juga beberapa nilai nol. Dengan tambahan nilai tersebut, lebar frame menjadi 256. Bilangan-bilangan tersebut berfungsi dalam pembentukan spektrum frekuensi dari sinyal OFDM. Beberapa literatur menjelaskan bahwa dengan memperbanyak nilai-nilai nol yang disisipkan pada modulasi OFDM, pembentukan spektrum frekuensi dari sinyal OFDM akan semakin baik. Pada simulasi ini kita menggunakan FFT length 256 dengan frame data 192. Parameter dari blok ini dapat dilihat pada Gambar 3.7.



**Gambar 3.7** parameter blok "Matrix Concatenation"

### 3.2.3 Blok "IFFT"

Pembentukan sinyal OFDM dapat dengan mudah dilakukan dengan menggunakan algoritma IDFT seperti telah dijelaskan sebelumnya. Blok "IFFT" ini adalah blok yang mengerjakan penghitungan *Inverse Fast Fourier Transform* dari sinyal. Panjang sinyal yang melalui blok ini adalah 256 sebagaimana keluaran dari blok *OFDM Modulation*. Keluaran dari blok ini adalah sinyal yang telah orthogonal dengan panjang frame 256.

### 3.2.4 Blok "Add Cyclic Prefix"

Telah dijelaskan pada bagian 2.6 bahwa penggunaan add cyclic prefix adalah untuk mengkompensasi ketidak-sinkronan dari sinyal yang akan diterima. Blok ini mengerjakan bagian penambahan cyclic prefix pada sinyal OFDM yang telah terbentuk dari IFFT. Pada Gambar 3.8 terlihat pada bagian "*elements (-1 for all elements)*" diisikan nilai [193:256 1:256].



**Gambar 3.8** parameter blok *Add Cyclic Prefix*

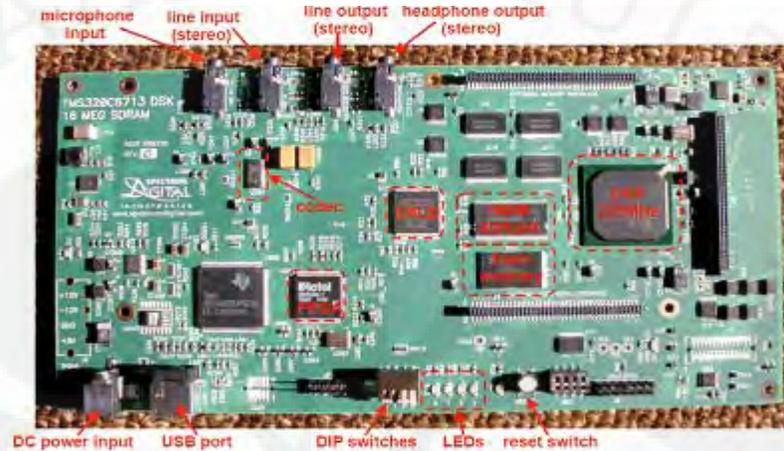
Ini berarti bahwa blok ini mengerjakan cyclic prefix dengan mengambil data dari elemen 193 sampai dengan 256 dan ditambahkan pada bagian depan dari elemen. Dengan demikian keluaran dari blok ini adalah sebesar:

$$256 + (256 - 193 + 1) = 320$$

### **3.3 DIGITAL SIGNAL PROCESSOR STARTER KIT TMS320C6713**

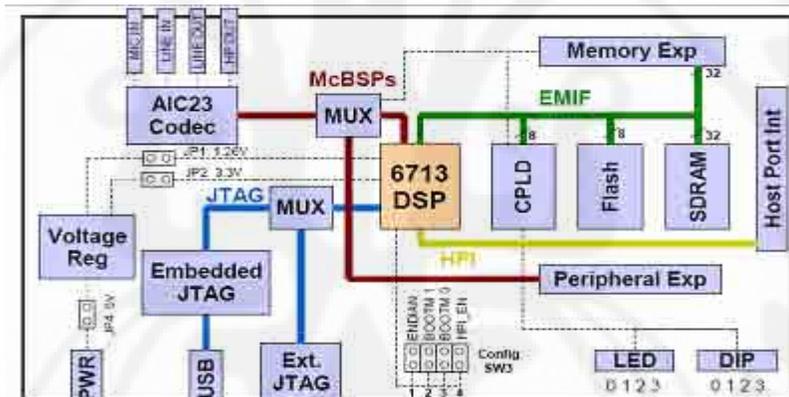
Blok diagram sebagaimana dibuat dalam SIMULINK<sup>®</sup> telah siap untuk diimplementasikan pada sebuah papan *processor*. Pada rancang bangun ini, papan *processor* yang digunakan adalah DSP Starter Kit TMS320C6713. *Digital Signal Processor* yang dalam hal ini adalah TMS320C6713 adalah *mikroprosesor* yang dibuat dalam arsitektur khusus untuk digunakan sebagai pemroses sinyal digital

(*Digital Signal Processing*). Modul ini biasa digunakan untuk membuat sebuah rancang bangun dasar dari sebuah pemroses sinyal yang hendak disimulasikan. Papan TMS320C6713 ini dapat kita lihat pada Gambar 3.9 berikut:



**Gambar 3.9** papan TMS320C6713

dengan blok diagram ditunjukkan pada Gambar 3.10 berikut:



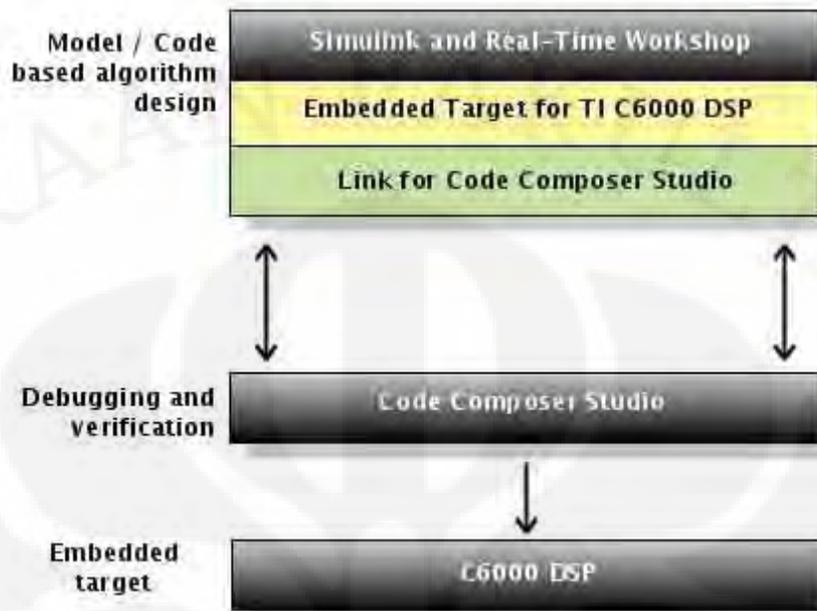
**Gambar 3.10** blok diagram dari TMS320C6713

Beberapa bagian dari papan DSK ini yang perlu diperhatikan antara lain adalah LINE IN dan LINE OUT yang merupakan modul masukan dan keluaran dari DSK ini. Modul lain yang perlu diperhatikan adalah *Joint Team Action Group* (JTEG) yang memungkinkan terjadinya komunikasi antara DSK dengan komputer. Komunikasi yang dilakukan antara komputer dengan DSK dilakukan melalui port USB (*Universal Serial Bus*). Melalui port ini dilakukan tukar menukar data antara komputer dengan DSK dan juga sinyal-sinyal pengendalian dari komputer ke DSK.

Proses komunikasi secara *real time* antara komputer dengan DSK ini disebut sebagai RTDX (Real Time Data Exchange). Modul lain yang juga dapat digunakan sebagai keluaran sinyal adalah modul LED dan DIP *switch*. Pada modul LED, sinyal keluaran direpresentasikan dengan nyala dari lampu LED. Sedangkan pada DIP *switch*, sinyal keluaran direpresentasikan dengan perubahan nilai tegangan dari tombol DIP *switch*.

### 3.4 IMPLEMENTASI BLOK SIMULINK PADA TMS320C6713

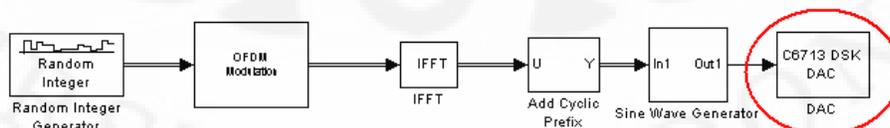
Blok yang telah dibuat pada simulink dapat kemudian kita masukkan dalam DSK TMS320C6713 dengan hanya melakukan beberapa setting pada *Configuration Parameters*. Proses debugging dari SIMULINK<sup>®</sup> ke DSK memerlukan perangkat lunak lain yaitu Code Composer Studio (CCS) yang merupakan sebuah komposer bahasa C/C++ yang digunakan untuk memprogram papan DSK dengan menggunakan bahasa C/C++. Maka sebenarnya yang dilakukan oleh SIMULINK<sup>®</sup> adalah membangun kode pemrograman dalam bahasa C/C++ dari model yang disimulasikan. Kode pemrograman tersebut kemudian dijalankan oleh CCS. Dan kemudian SIMULINK<sup>®</sup> dan papan DSK TMS320C6713 dapat berkomunikasi melalui kode-kode pemrograman yang dibuat oleh SIMULINK<sup>®</sup>. Inilah kemudahan yang diberikan oleh SIMULINK<sup>®</sup>, yaitu kita tidak perlu membuat dan membangun kode pemrograman dari model yang kita simulasikan. Kita hanya cukup membangun blok model pada SIMULINK<sup>®</sup> dan dengan mudah kita akan mendapatkan kode pemrograman dalam bahasa C/C++ yang dapat digunakan untuk membuat rancang bangun dari model yang telah kita buat dalam SIMULINK<sup>®</sup> pada DSK TMS320C6713. Gambar 3.11 menunjukkan algoritma dari penerapan SIMULINK pada DSK.



**Gambar 3.11** algoritma penanaman model ke dalam TMS320C6713 sebagai *embedded target*

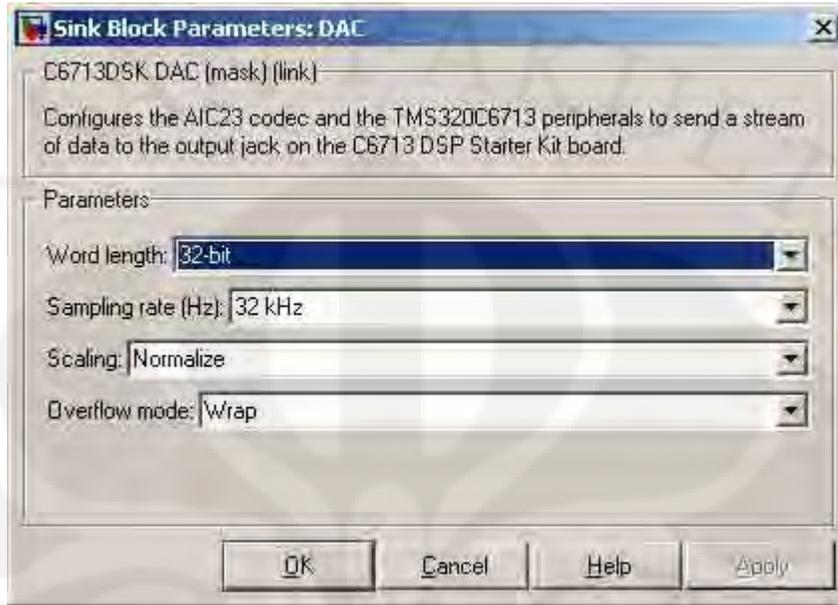
Berikut adalah langkah-langkah yang dilakukan untuk dapat mengimplementasikan model yang telah kita buat dalam SIMULINK<sup>®</sup> ke dalam papan TMS320C6713 sebagai *embedded target*:

- 1) Memasangkan Digital-to-Analog Converter (DAC) sebagai keluaran dari DSK (Gambar 3.12). Pada blok ini, sinyal keluaran dari OFDM kemudian akan dikonversikan dari bentuk digital ke dalam bentuk analog yang akan menjadi keluaran dari papan DSK. Dengan menggunakan blok ini, kita akan dapat melihat sinyal keluaran OFDM dari keluaran papan (LINE OUT) berupa sinyal kontinu.



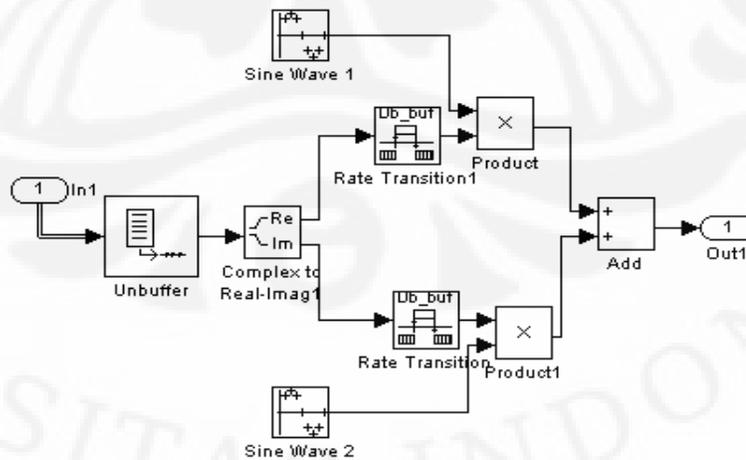
**Gambar 3.12** penambahan blok C6713 DSK DAC

Dengan parameter blok DAC seperti ditunjukkan pada Gambar 3.13 berikut:



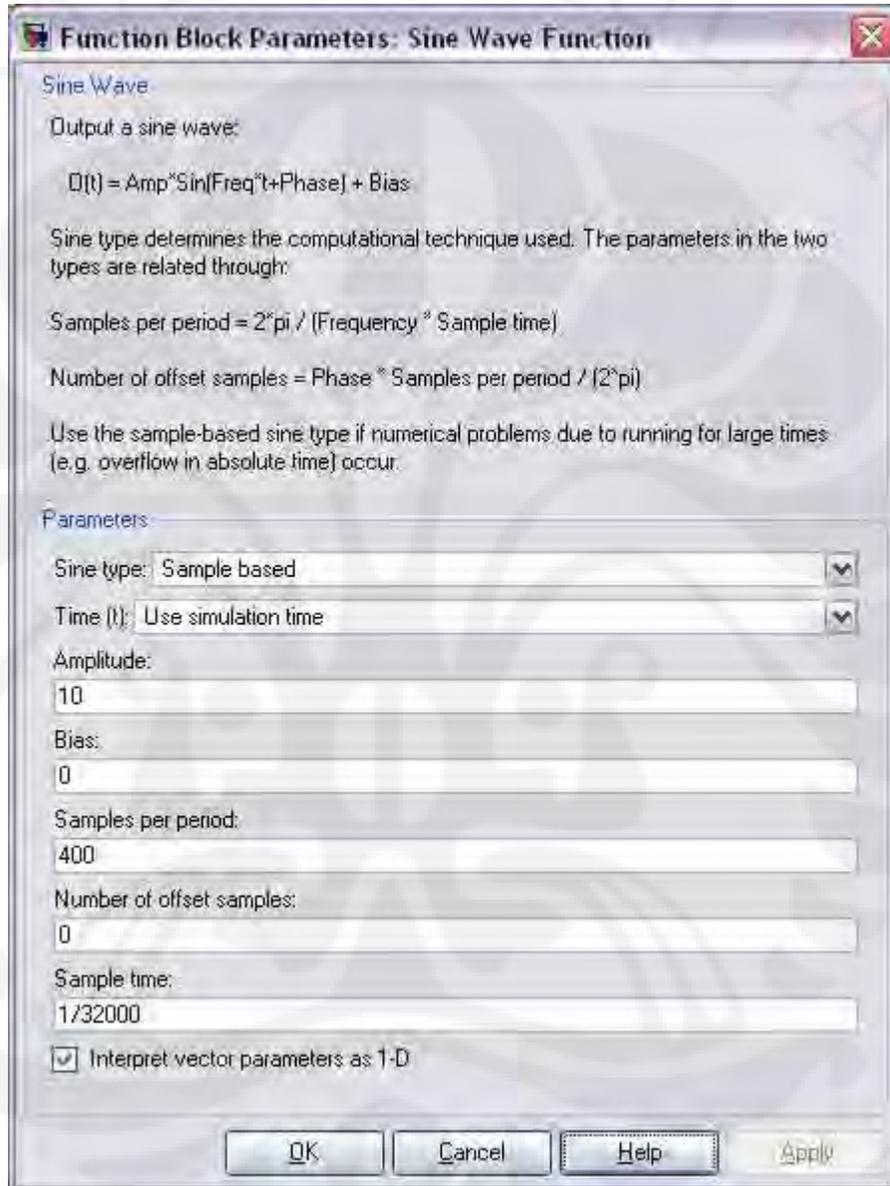
**Gambar 3.13** parameter blok C6713 DSK DAC

Pada Gambar 3.12 ada tambahan blok *sine wave generator*. Blok ini berfungsi untuk membangun sinyal sinusoidal agar keluaran dari modul simulasi dapat kita lihat dan ukur sebagai sinyal sinusoidal. Blok ini adalah sebuah subsystem yang terdiri dari beberapa blok dasar seperti yang dapat kita lihat pada Gambar 3.14 sebagai berikut:



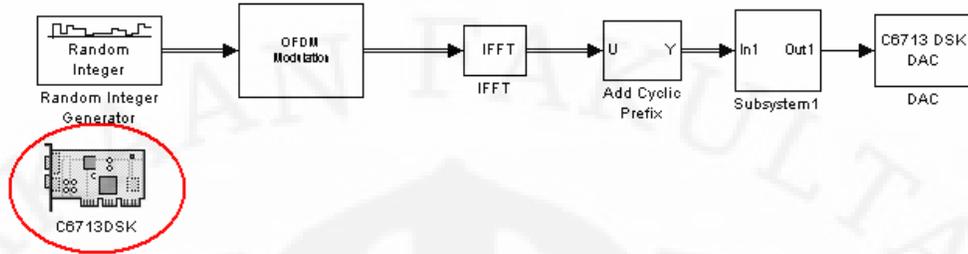
**Gambar 3.14** subsystem *sine wave generator* yang terdiri dari beberapa blok

Hal yang perlu diperhatikan dari subsystem ini adalah parameter blok dari sine wave 1 dan sine wave 2. Gambar 3.15 menunjukkan parameter dari blok sine wave 1 dan sine wave 2 sebagai berikut:



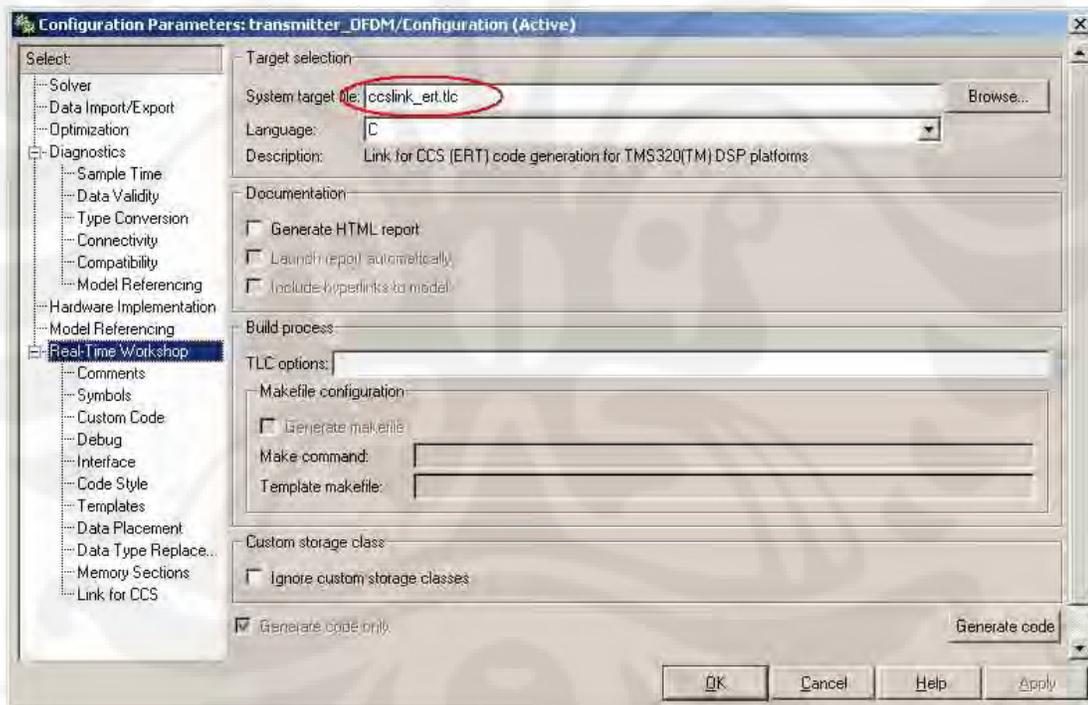
**Gambar 3.15** parameter blok sine wave 1 dan 2

- 2) Mencatumkan blok C6713DSK dari library browser ke model yang telah dibuat (Gambar 3.16). Blok ini menandakan bahwa simulasi yang kita buat siap untuk diimplementasikan pada papan DSK TMS320C6713 sebagai embedded target.

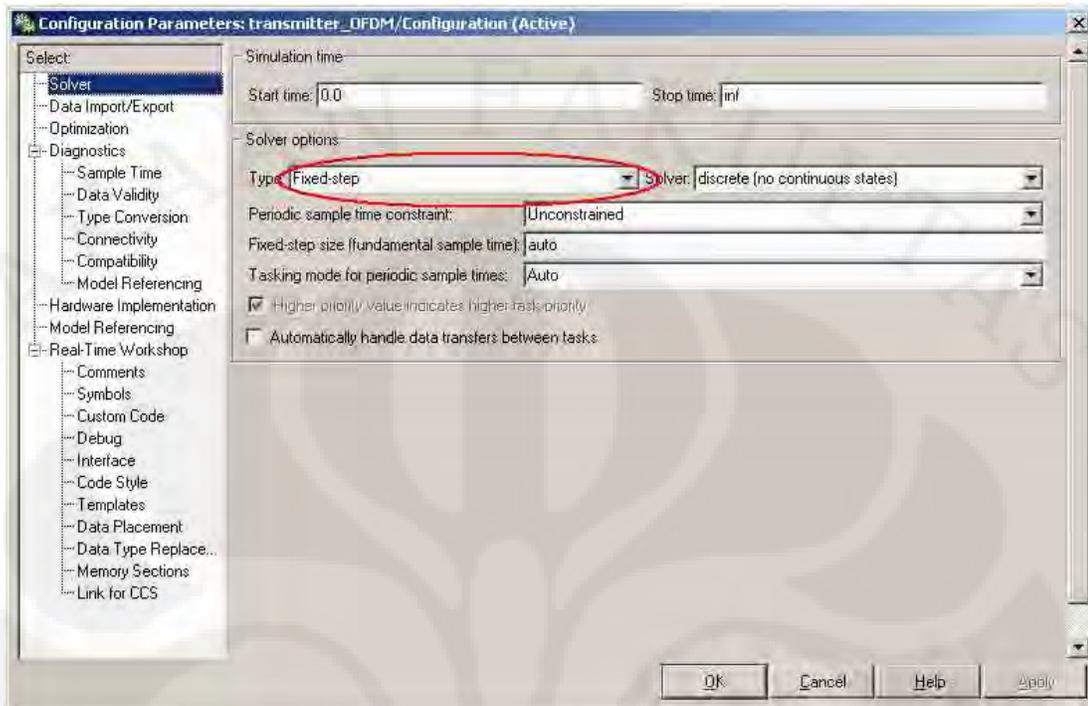


**Gambar 3.16** penambahan blok C6713 DSK

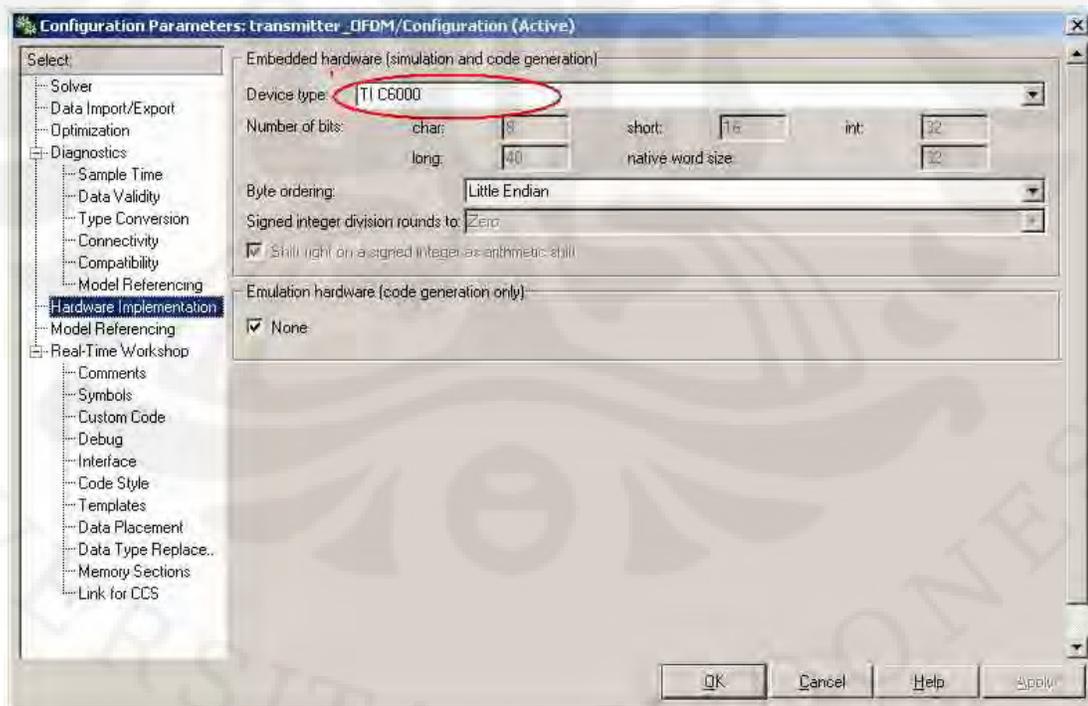
- 3) Pada toolbar pilih simulations → configuration parameters atau dengan menekan Ctrl+E. Dan lakukan beberapa setting seperti terlihat pada Gambar 3.17 sampai dengan Gambar 3.20 berikut:



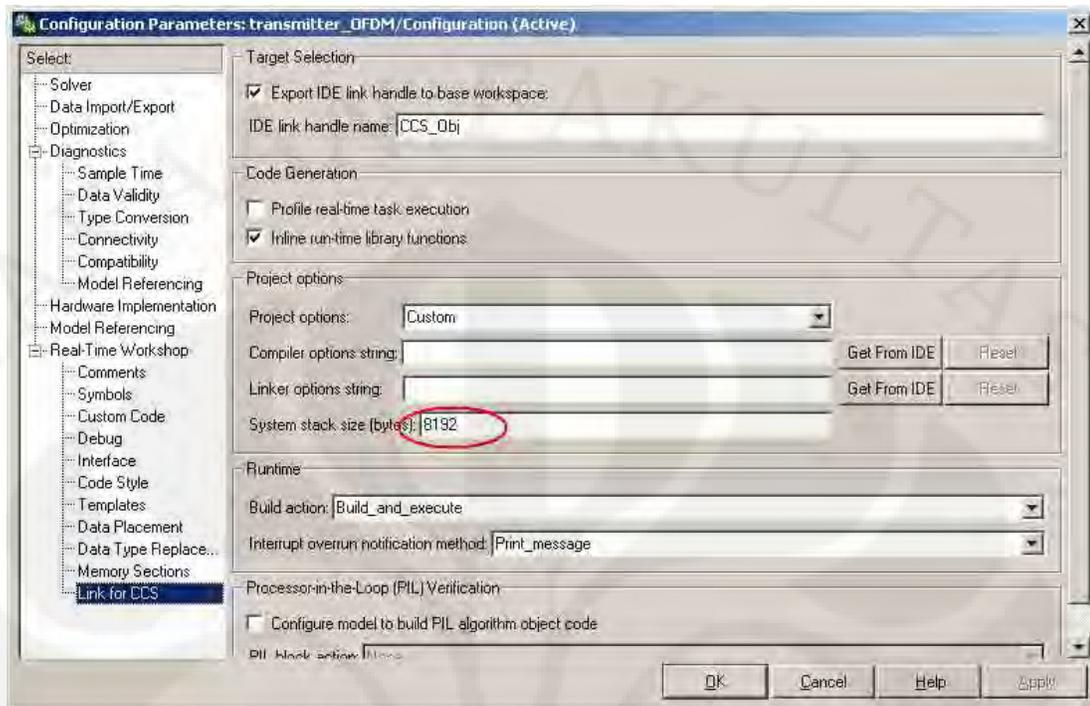
**Gambar 3.17** pada tab menu “Real Time Workshop”



**Gambar 3.18** pada tab menu “Solver”



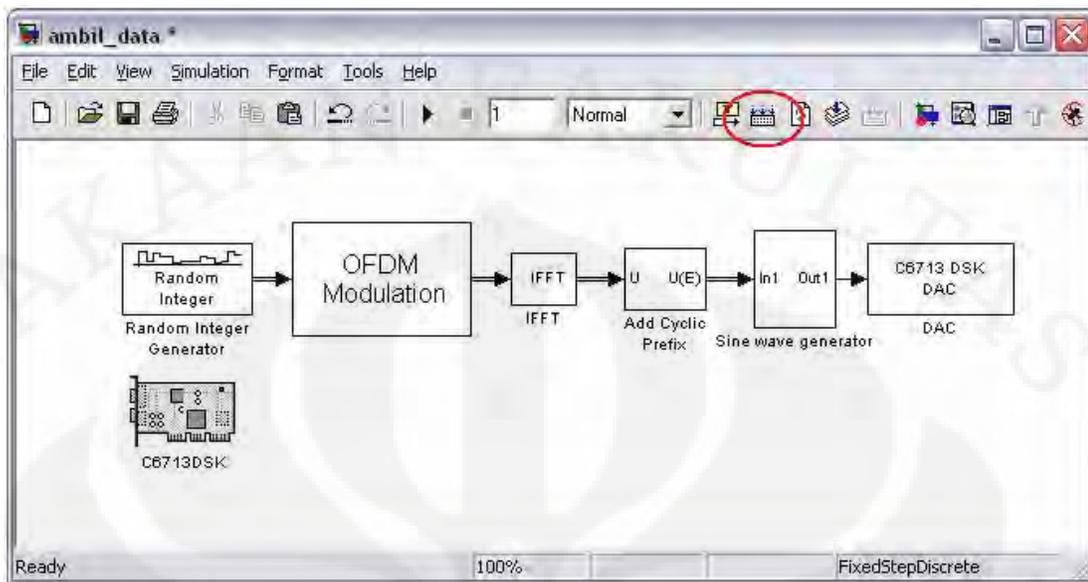
**Gambar 3.19** pada tab menu “Hardware Implementation”



**Gambar 3.20** pada tab menu “Link for CCS”

Setelah semua setting pada Configuration Parameters dilakukan kemudian pilih OK dan kembali ke model simulasi.

- 4) Pada toolbar SIMULINK, tekan tombol incremental build seperti pada Gambar 3.21. Setelah itu SIMULINK akan membuat kode bahasa C dari blok model yang telah dibuat dan dengan demikian blok model telah siap diimplementasikan pada papan DSK.



**Gambar 3.21** tombol incremental build

Setelah semua langkah tersebut dilakukan, maka SIMULINK akan membangun kode-kode pemrograman dalam bahasa C pada Code Composer Studio. Setelah proses pembuatan kode selesai, maka CCS akan membuka secara otomatis dan DSK telah terhubung dengan komputer melalui perangkat lunak CCS. Setelah proses ini selesai, maka pada DSK telah dimasukkan modul simulasi pemancar OFDM. Model ini kemudian akan berjalan pada perangkat DSK secara real time.

## BAB IV

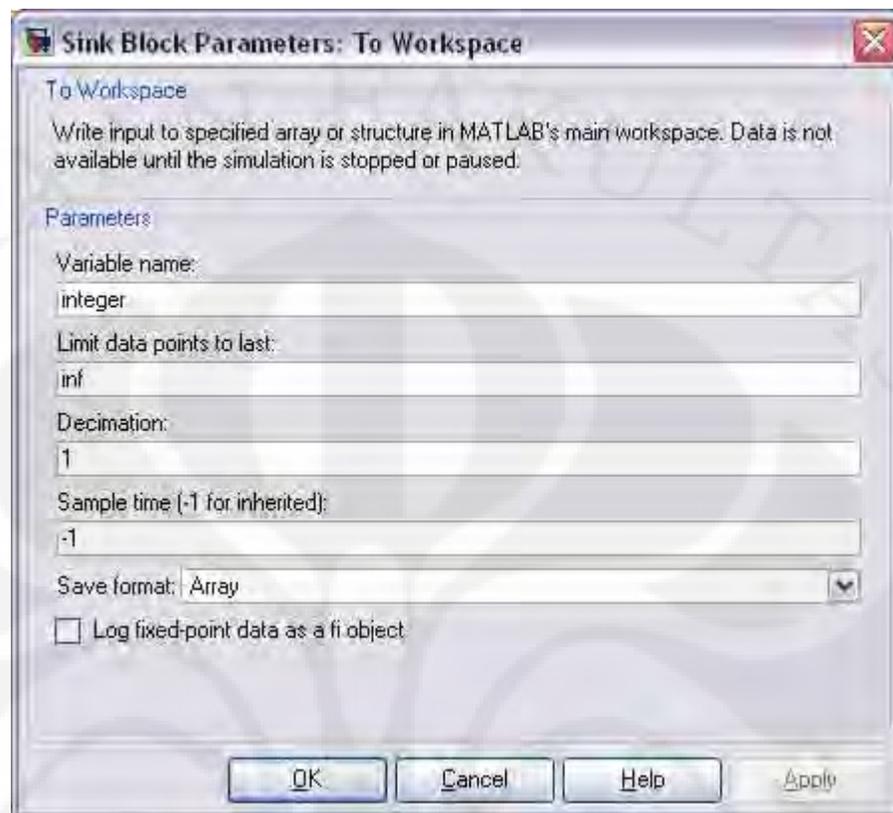
### UJI COBA DAN ANALISA MODEL PEMANCAR OFDM

#### 4.1 PROSEDUR UJI COBA DAN PENGAMBILAN DATA

Uji coba dilakukan setelah model simulasi telah berhasil diterapkan pada DSK. Keluaran yang diukur didapat dengan menggunakan dua metode yaitu dengan menggunakan program MATLAB dan dengan menggunakan osiloskop.

Pengukuran dengan menggunakan osiloskop digunakan untuk melihat sinyal keluaran dari papan DSK. Sinyal yang dikeluarkan pada DSK tersebut adalah sinyal pada keadaan sebenarnya yang merupakan sinyal OFDM yang telah siap memasuki saluran komunikasi. Kendala pada pengukuran dengan menggunakan DSK ini adalah pengukuran hanya dapat dilakukan terbatas pada pengamatan bentuk gelombang yang terjadi. Analisa terhadap nilai-nilai masukan dari sumber data tidak dapat dilakukan karena terkendala pada besarnya range data dan terbatasnya lebar pengamatan yang dapat terlihat pada osiloskop. Karena keterbatasan tersebut, maka pengambilan data sinyal pada *LINE OUT* dari DSK hanya dilakukan untuk melihat bentuk sinyal keluaran dari *LINE OUT* dan dibandingkan dengan bentuk sinyal keluaran yang terukur pada MATLAB.

Pengukuran pada SIMULINK dapat ditelusuri sampai kepada bilangan-bilangan untuk masing-masing blok. Dengan mengambil nilai masukan dan keluaran dari masing-masing blok, analisa dilakukan dengan melihat hubungan antara masukan dengan keluaran pada masing-masing blok. Pengambilan data untuk masing-masing blok dilakukan dengan menambahkan blok '*to workspace*' untuk mengambil data. Dengan menggunakan blok '*to workspace*' data yang diambil dari masing-masing blok diolah menjadi variabel pada *workspace* dari MATLAB dan dapat langsung dilihat nilai masukan dari masing-masing variabel pada MATLAB. Parameter blok '*to workspace*' ditunjukkan pada Gambar 4.1 berikut:

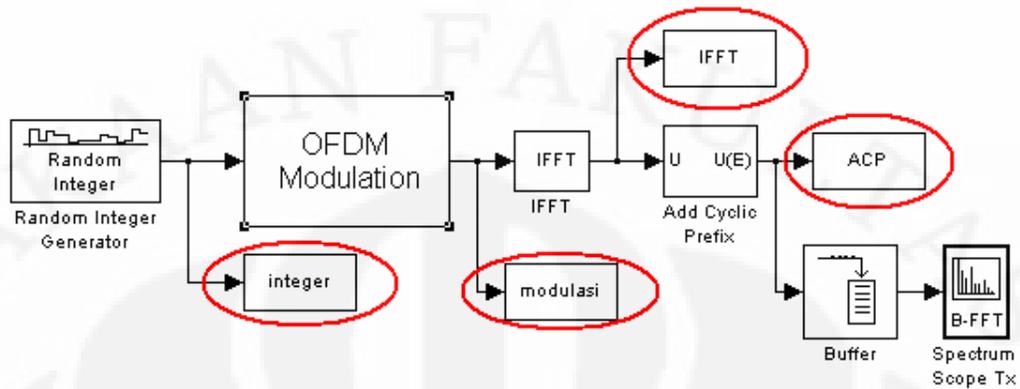


**Gambar 4.1** parameter blok *'to workspace'*

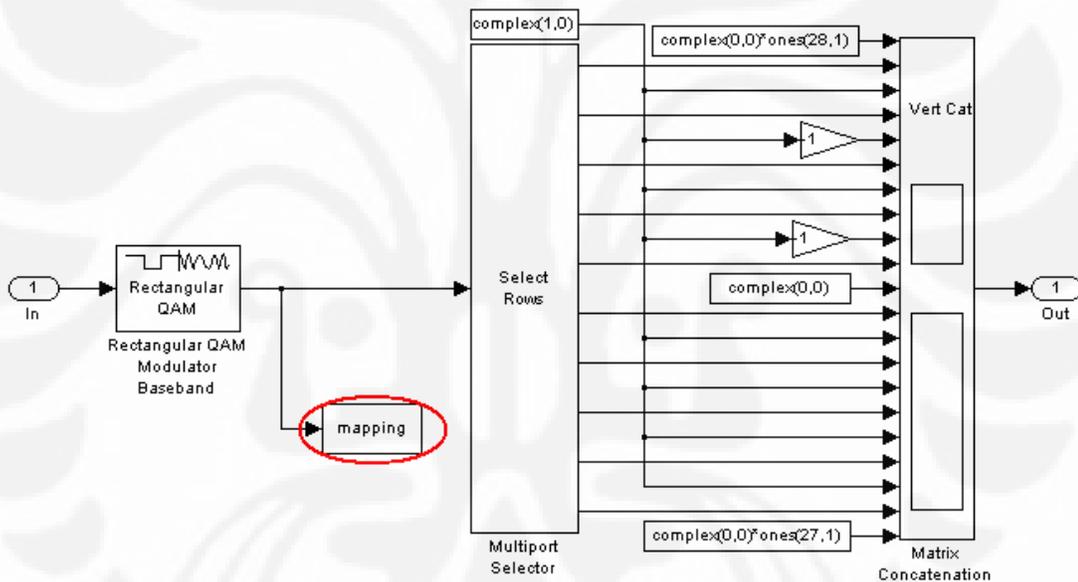
Hal yang perlu diperhatikan dari parameter blok ini adalah pada baris *'Save format'*, maka format yang dipilih adalah array.

## **4.2 PENGAMBILAN DAN ANALISA DATA PADA MATLAB**

Pencantuman blok *'to workspace'* ditunjukkan pada Gambar 4.2 dan Gambar 4.3 berikut:



**Gambar 4.2** pencantuman blok-blok 'to workspace' pada modul utama



**Gambar 4.3** pencantuman blok 'to workspace' pada subsystem OFDM modulation

Data- data yang didapatkan dari blok to workspace kemudian disimpan dalam bentuk array pada MATLAB. Nama-nama variabel yang digunakan seperti yang dapat dilihat pada Gambar 4.2 dan Gambar 4.3 antara lain: integer, mapping, modulasi, IFFT, dan ACP. Nilai-nilai dari masing-masing variabel yang terekam terdapat pada lempiran.

#### 4.2.1 Variabel 'Integer'

Integer adalah nama variabel yang digunakan untuk menyimpan data keluaran dari Random Integer Generator. Nilai-nilai dari variabel ini bertipe data integer sebagaimana sumber data yang digunakan adalah sumber data integer. Rentang nilai dari variabel ini adalah antara 0 sampai dengan 63. Hal ini dikarenakan modulasi yang kita gunakan adalah 64-QAM. Variabel ini kemudian menjadi masukan bagi blok 64-QAM yang akan memetakan bilangan sumber data integer menjadi bilangan kompleks. Dimensi dari variabel ini adalah array dengan dimensi [192x1]. Telah dibahas pada bab sebelumnya bahwa untuk dapat memahami konsep orthogonal maka kita harus memandang sinyal sebagai vektor. Pada penelitian ini lebar data yang digunakan untuk satu frame adalah sebesar 192.

Jika kita menggunakan tipe data integer dengan rentang nilai antara 0 - 63, maka kita memerlukan 6 bit bilangan biner untuk masing-masing bilangan integer yang dibangkitkan. Pada parameter blok 'source' pada bagian *Samples per frame* nilainya adalah 192 seperti diperlihatkan pada Gambar 3.3. Ini berarti pada selama satu detik dibangkitkan 192 data integer. Untuk satu bilangan integer dibutuhkan 6 bit, maka untuk satu detik kecepatan datanya adalah:

$$\text{bit rate} = 192 \times 6 = 1152 \text{ bit per second}$$

Tabel 4.1 menampilkan 20 bilangan integer pertama (elemen ke-1 sampai dengan 20) serta bit-bit yang bersesuaian. Data selengkapnya dari variabel 'integer' pada selang waktu 1 detik terlampir.

#### 4.2.2 Variabel 'Mapping'

Variabel ini berisi nilai-nilai hasil *mapping* dari bilangan integer masukan 64-QAM. Tipe data untuk variabel ini adalah tipe data kompleks. Masing-masing bilangan integer masukan dari 64-QAM dipetakan menjadi bilangan kompleks yang bersesuaian. Nilai kompleks untuk masing-masing bilangan integer dapat dilihat pada Tabel 3.1. Untuk variabel ini, bentuk datanya tetap berupa vektor berukuran [192x1]. Hasil *mapping* dari 20 variabel 'integer' pertama ditunjukkan pada Tabel 4.2. Pada tabel tersebut terlihat bahwa hasil pemetaan dari masing-masing bilangan integer

pada variabel 'integer' telah sesuai dengan Tabel 3.1. Hasil lengkap nilai variabel 'mapping' terlampir.

**Tabel 4.1** 20 data pertama dari variabel 'integer'

elemen	integer	bit
1	34	100010
2	12	001100
3	4	000100
4	25	011001
5	42	101010
6	58	111010
7	34	100010
8	63	111111
9	35	100011
10	13	001101
11	62	111110
12	20	010100
13	25	011001
14	30	011110
15	14	001110
16	55	110111
17	41	101001
18	22	010110
19	18	010010
20	7	000111

**Tabel 4.2** 20 nilai mapping pertama dan bilangan integer yang bersesuaian

elemen	integer	mapping
1	34	0.15 + 0.45i
2	12	-0.75 - 0.15i
3	4	-1.05 - 0.15i
4	25	-0.15 + 0.75i
5	42	0.45 + 0.45i
6	58	1.05 + 0.45i
7	34	0.15 + 0.45i
8	63	1.05 - 1.05i
9	35	0.15 + 0.15i
10	13	-0.75 - 0.45i
11	62	1.05 - 0.75i
12	20	-0.45 - 0.15i
13	25	-0.15 + 0.75i
14	30	-0.15 - 0.75i
15	14	-0.75 - 0.75i
16	55	0.75 - 1.05i
17	41	0.45 + 0.75i
18	22	-0.45 - 0.75i
19	18	-0.45 + 0.45i
20	7	-1.05 - 1.05i

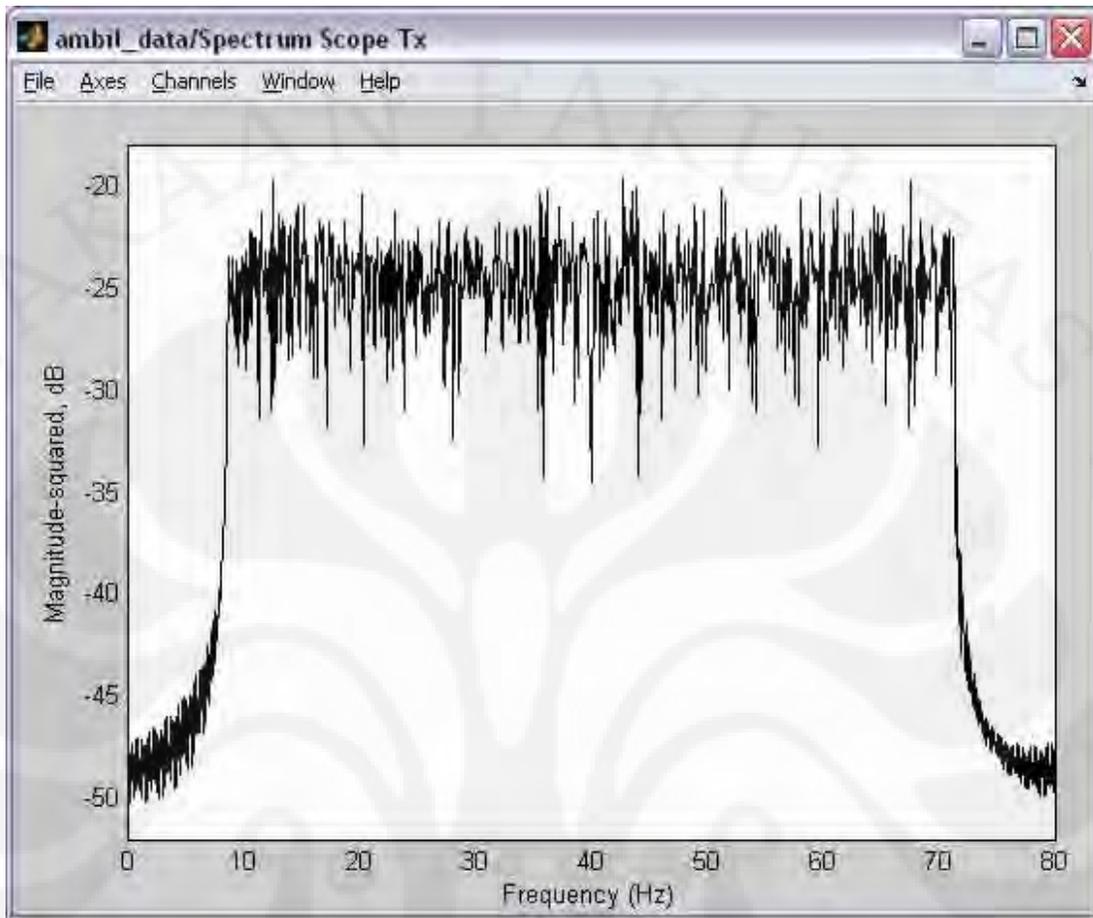
#### 4.2.3 Variabel 'Modulasi'

Pada variabel ini, ukuran vektor menjadi [256x1] ini karena pada subsystem OFDM modulation terdapat penambahan nilai-nilai kompleks 0, 1, dan -1. Penambahan bilangan ini adalah untuk pembentukan spektrum frekuensi yang baik. Beberapa literatur menyatakan bahwa dengan semakin banyaknya zero padding yang digunakan, maka bentuk dari spektrum frekuensi akan semakin baik. Jika sama sekali tidak menggunakan zero padding, maka bentuk spektrum frekuensi menjadi sangat buruk. Pada percobaan ini besar data pada satu frame adalah [192x1] dengan penambahan zero padding, besar satu frame menjadi [256x1]. Tabel 4.3 menunjukkan contoh data pada variabel 'modulasi' untuk elemen ke-24 sampai ke-43. Variabel 'modulasi' untuk selang waktu 1 detik terlampir.

**Tabel 4.3** data variabel 'modulasi' pada elemen ke-24 sampai ke-43

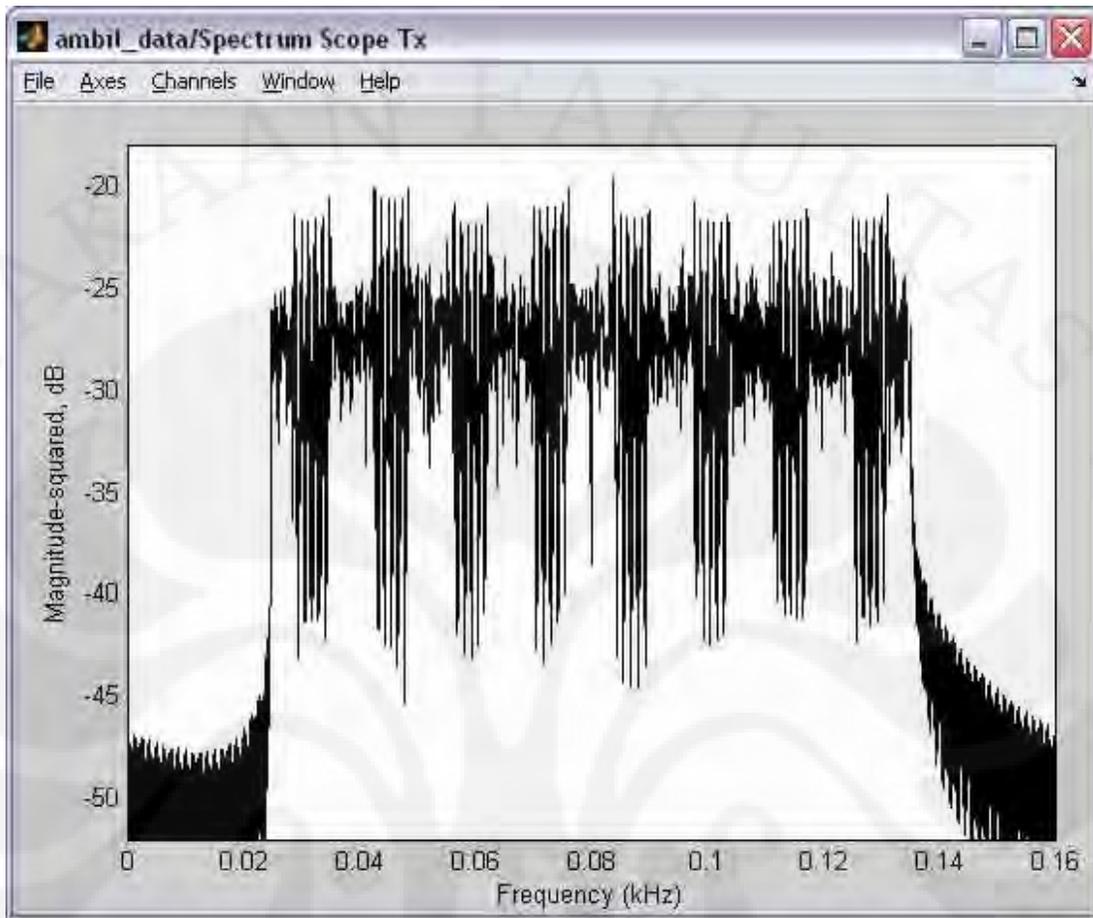
elemen	integer	mapping	modulasi
24			0
25			0
26			0
27			0
28			0
29	34	0.15 + 0.45i	0.15 + 0.45i
30	12	-0.75 - 0.15i	-0.75 - 0.15i
31	4	-1.05 - 0.15i	-1.05 - 0.15i
32	25	-0.15 + 0.75i	-0.15 + 0.75i
33	42	0.45 + 0.45i	0.45 + 0.45i
34	58	1.05 + 0.45i	1.05 + 0.45i
35	34	0.15 + 0.45i	0.15 + 0.45i
36	63	1.05 - 1.05i	1.05 - 1.05i
37	35	0.15 + 0.15i	0.15 + 0.15i
38	13	-0.75 - 0.45i	-0.75 - 0.45i
39	62	1.05 - 0.75i	1.05 - 0.75i
40	20	-0.45 - 0.15i	-0.45 - 0.15i
41			1
42	25	-0.15 + 0.75i	-0.15 + 0.75i
43	30	-0.15 - 0.75i	-0.15 - 0.75i

Gambar 4.4 menunjukkan spektrum frekuensi dari sinyal OFDM yang dikeluarkan oleh pemancar sebagai berikut:



**Gambar 4.4** spektrum frekuensi pada model dengan  $K = 192$  dan FFT length  $N = 256$

Pada Gambar 4.4 diatas terlihat bahwa spektrum frekuensi untuk  $K = 192$  dan FFT length  $N = 256$  belum terlalu halus. Untuk nilai  $K$  yang tetap, dengan semakin panjangnya FFT length  $N$ , maka bentuk spektrum frekuensi dari sinyal OFDM akan semakin baik. Sebagai perbandingan, Gambar 4.5 adalah spektrum frekuensi untuk  $K=192$  dan FFT length  $N=512$  sebagai berikut:



**Gambar 4.5** spektrum frekuensi pada model dengan  $K = 192$  dan FFT length  $N = 512$ . Pada Gambar 4.5 terlihat bahwa untuk FFT length yang lebih besar, spektrum frekuensi pada rentang frekuensi yang digunakan lebih baik, namun side band yang terjadi juga menjadi lebih besar.

Nilai data untuk variabel modulasi dapat dilihat pada lampiran. Pada lampiran tersebut terlihat bahwa untuk elemen vektor ke-1 sampai ke-28 dan elemen vektor ke-230 sampai ke-256 nilainya adalah nol. Hal ini karena seperti terlihat pada Gambar 3.4 bahwa pada awal vektor diberikan 28 nilai kompleks nol, dari elemen vektor ke-1 sampai ke-28. Dan pada akhir vektor diberikan 27 nilai kompleks nol dari elemen vektor ke-230 sampai ke-256.

Setelah 28 nilai nol pada awal vektor 'modulasi', elemen vektor kemudian berisi nilai hasil mapping dari 64-QAM pada elemen vektor ke-29 sampai ke-40. Ini adalah bagian satu dari data serial hasil mapping seperti terlihat pada Tabel 3.2

dengan panjang 12 elemen. Vektor ‘modulasi’ kembali berisi data hasil mapping pada elemen ke-42 sampai ke-65 (24 elemen) yang adalah bagian kedua dari hasil mapping seperti dilihat pada Tabel 3.2. Seterusnya secara berurutan data hasil simulasi menempati elemen vektor ke-67 sampai ke-90, elemen vektor ke-92 sampai ke-115, elemen vektor ke-117 sampai ke-128, elemen vektor ke-130 sampai ke-141, elemen vektor ke-143 sampai ke-166, elemen vektor ke-168 sampai ke-191, elemen vektor ke-193 sampai ke-216, elemen vektor ke-218 sampai ke-229 dengan urutan dan panjang masing-masing bagian seperti terdapat pada Tabel 3.2

Untuk elemen vektor ke-41, 66, 91, 116, 129, 142, 167, 192, 217 berisi nilai-nilai secara berurutan adalah: 1, -1, 1, -1, 0, 1, 1, 1, dan 1. Ini sesuai dengan blok yang kita gunakan seperti ditunjukkan pada Gambar 3.4. Dengan demikian, keseluruhan nilai vektor yang didapat adalah seperti ditunjukkan pada lampiran.

#### **4.2.4 Variabel ‘IFFT’**

Variabel modulasi seperti telah dijelaskan sebelumnya kemudian menjadi masukan bagi blok IFFT. Pada blok ini dikerjakan perhitungan Inverse Fourier Transform, sehingga variabel ‘IFFT’ yang adalah keluaran dari blok ‘IFFT’ adalah nilai Inverse Fast Fourier Transform dari variabel ‘modulasi’. Perhitungan IFFT melalui MATLAB untuk variabel ‘modulasi’ memberikan hasil yang sama dengan variabel ‘IFFT’ pada *workspace*.

Perhitungan FFT dari variabel ‘IFFT’ seharusnya juga memberikan hasil yang sama dengan variabel ‘modulasi’. Penghitungan FFT pada MATLAB untuk variabel IFFT memberikan hasil yang sangat mendekati dengan nilai dari variabel ‘modulasi’. Memang terjadi perbedaan, namun sangat kecil, pembulatan yang dilakukan akan memberikan hasil yang sama dengan variabel ‘modulasi’.

Bentuk data dari variabel ini adalah vektor dengan ukuran [256x1] sesuai dengan ukuran dari masukan blok ‘IFFT’, sesuai dengan teori bahwa untuk  $N$  poin FFT atau IFFT akan menghasilkan  $N$  poin hasil transformasi. Tabel 4.4 menunjukkan hasil IFFT dari variabel ‘modulasi’ dan juga pengujian kembali dengan menggunakan FFT untuk mengembalikan hasil IFFT kembali ke variabel ‘modulasi’ untuk elemen ke-24 sampai ke-43.

**Tabel 4.4** variabel ‘IFFT’ pada elemen ke-24 sampai ke-43

Elemen	modulasi	IFFT	FFT
24	0	0.019336 + 0.013497i	2.7756e-016 -4.0246e-016i
25	0	0.049978 - 0.0010305i	1.1102e-016 -1.3878e-016i
26	0	-0.016035 - 0.021056i	-1.6653e-016 -3.8858e-016i
27	0	0.00054881 + 0.0090598i	3.3307e-016 -2.2204e-016i
28	0	-0.058598 + 0.021224i	-4.4409e-016 +2.7756e-017i
29	0.15 + 0.45i	-0.013666 - 0.040993i	0.15 + 0.45i
30	-0.75 - 0.15i	0.043725 - 0.057655i	-0.75 - 0.15i
31	-1.05 - 0.15i	-0.031448 + 0.021191i	-1.05 - 0.15i
32	-0.15 + 0.75i	0.029394 + 0.058563i	-0.15 + 0.75i
33	0.45 + 0.45i	0.026479 + 0.00852i	0.45 + 0.45i
34	1.05 + 0.45i	-0.036814 + 0.031008i	1.05 + 0.45i
35	0.15 + 0.45i	0.066665 - 0.022905i	0.15 + 0.45i
36	1.05 - 1.05i	-0.056302 - 0.032781i	1.05 - 1.05i
37	0.15 + 0.15i	-0.058304 - 0.05699i	0.15 + 0.15i
38	-0.75 - 0.45i	0.033729 + 0.053906i	-0.75 - 0.45i
39	1.05 - 0.75i	0.048161 - 0.028104i	1.05 - 0.75i
40	-0.45 - 0.15i	-0.026189 + 0.03306i	-0.45 - 0.15i
41	1	-0.0028357 + 0.02261i	1
42	-0.15 + 0.75i	0.02544 + 0.014638i	-0.15 + 0.75i
43	-0.15 - 0.75i	-0.011459 - 0.043418i	-0.15 - 0.75i

Nilai-nilai variabel ‘IFFT’ dapat dilihat pada lampiran. Pada lampiran juga disertakan perhitungan FFT dari variabel ‘IFFT’ yang memberikan hasil yang sama dengan variabel ‘modulasi’.

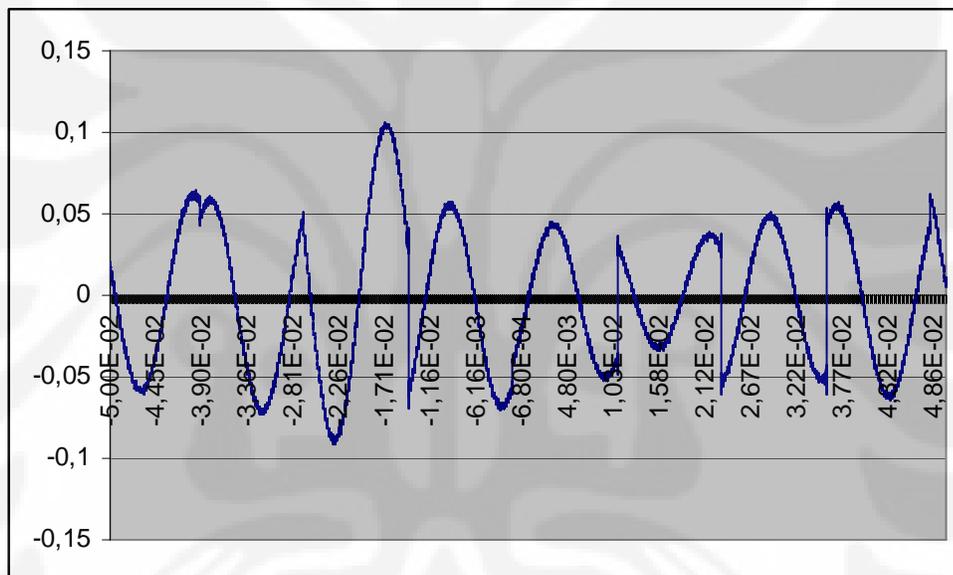
#### 4.2.5 Variabel ‘ACP’

Masukan bagi blok ‘Add Cycle Prefix’ adalah variabel IFFT. Blok ‘Add Cyclic Prefix’ dengan parameter seperti terlihat pada Gambar 3.8 menambahkan elemen ke-193 sampai ke-256 dari variabel ‘IFFT’ ke bagian depan dari variabel ‘IFFT’. Dengan demikian isi dari variabel ‘ACP’ sama dengan variabel ‘IFFT’ dengan menambahkan elemen ke-193 sampai ke-256 ke bagian awal dari variabel ‘IFFT’. Bentuk dari variabel ‘modulasi’ ini adalah vektor dengan ukuran [320x1]. Panjang vektor 320 adalah hasil penambahan panjang vektor 256 ditambah dengan 64 (elemen ke-193 sampai ke-256). Nilai-nilai dari variabel ini dapat dilihat pada lampiran.

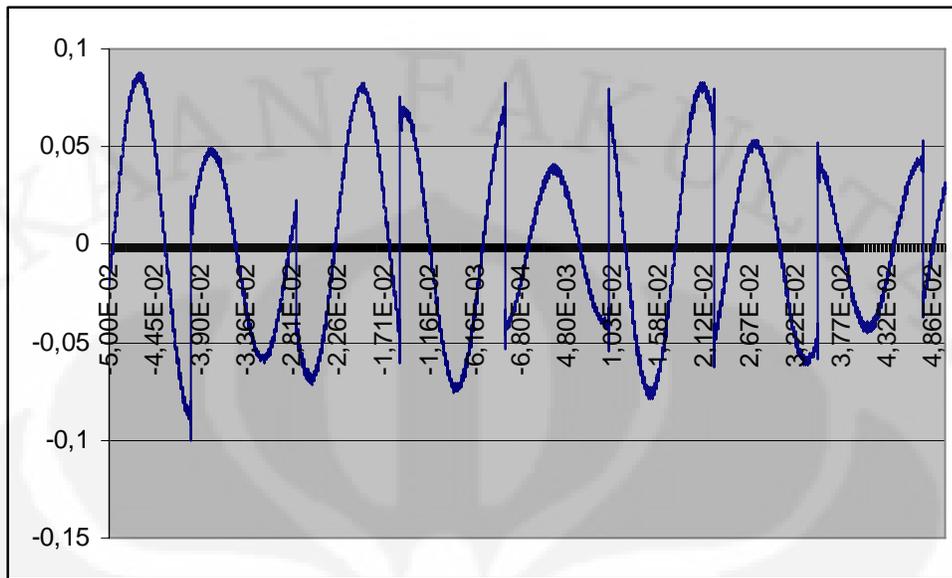
Variabel 'ACP' yang adalah keluaran dari blok 'Add Cyclic Prefix' adalah sinyal OFDM yang telah siap untuk ditransmisikan pada saluran komunikasi.

#### 4.3 PENGAMBILAN DAN ANALISA DATA PADA OSILOSKOP

Sinyal OFDM dari LINE OUT DSK diukur dengan menggunakan osiloskop untuk dapat melihat bentuk gelombang yang dihasilkan. Analisa terhadap sinyal keluaran tidak dapat dilakukan karena modul papan DSK berjalan secara *real time*. Berikut adalah beberapa bentuk sinyal yang diukur dengan menggunakan osiloskop seperti ditunjukkan oleh Gambar 4.6, dan Gambar 4.7 sebagai berikut:



**Gambar 4.6** sampel pertama bentuk gelombang keluaran LINE OUT DSK



**Gambar 4.7** sampel kedua bentuk gelombang keluaran LINE OUT DSK

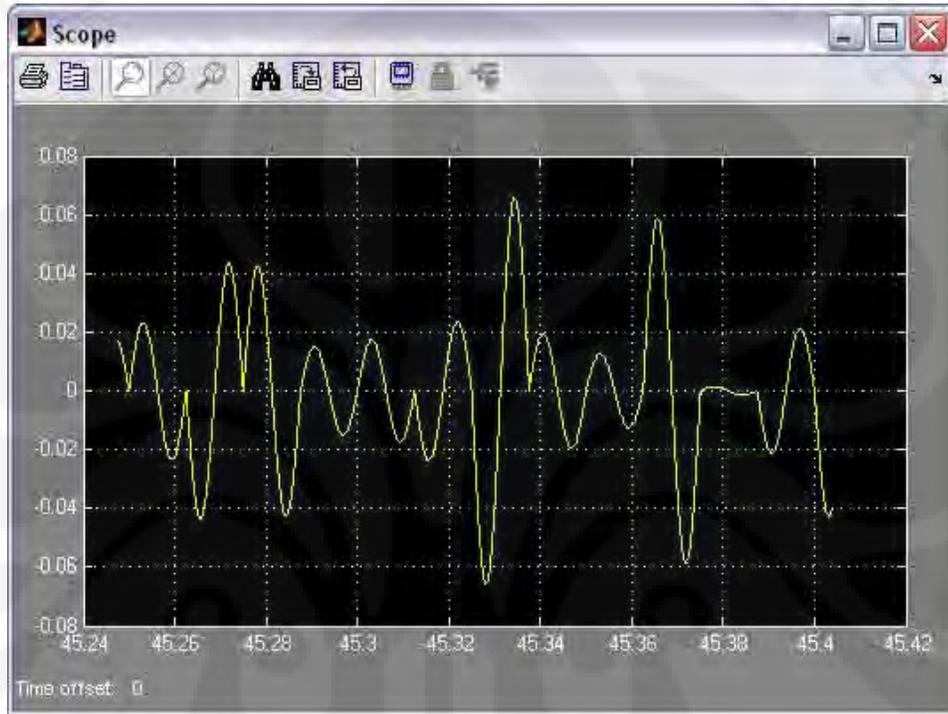
Cara pengambilan bentuk gelombang dengan menggunakan osiloskop adalah dengan menempelkan jarum pengukuran osiloskop pada LINE OUT DSK. Sinyal yang didapat pada dari keluaran DSK berupa perubahan tegangan yang membentuk gelombang. Perubahan tegangan tersebut kemudian terukur pada osiloskop. Osiloskop yang digunakan memungkinkan untuk merekam bentuk gelombang yang terbaca pada osiloskop dengan menggunakan pilihan *Save Active Waveform*. Gelombang yang ditangkap kemudian disimpan dalam media *floppy disk* agar dapat dipindahkan pada media lain dan dapat dibaca pada komputer. File disimpan dalam bentuk file *dot csv (\*.csv)*.

Pada komputer, data yang disimpan dalam bentuk *.csv* dapat dibuka dengan menggunakan aplikasi *spreadsheet* seperti *microsoft excel*. Pada aplikasi ini data dikenali sebagai sekumpulan nilai. Gelombang pada osiloskop dapat kembali dibangun dengan membuat *line chart* dari serial data yang terbaca tersebut. Seperti yang ditunjukkan pada Gambar 4.5 dan Gambar 4.6. Untuk satu frame data yang tersimpan dalam file, banyaknya pencuplikan adalah sebesar 10000 pencuplikan.

Gelombang yang terlihat pada LINE OUT DSK adalah bentuk gelombang sinusoidal yang memiliki amplitudo dan fasa yang berbeda-beda. Hal ini sesuai dengan jenis modulasi yang digunakan yaitu modulasi 64-QAM. Seperti telah

dijelaskan pada bagian 2.4.3 bahwa bentuk gelombang hasil modulasi QAM memiliki perubahan baik dalam ampiltudo maupun fasa.

Sebagai perbandingan, berikut adalah bentuk gelombang keluaran yang terukur dengan menggunakan MATLAB seperti ditunjukkan oleh Gambar 4.8:



**Gambar 4.8** sinyal OFDM diukur dengan MATLAB

Dalam Gambar 4.8 bentuk gelombang keluaran dari rancang bangun pemancar OFDM pada SIMULINK berupa sinyal QAM yang sesuai dengan keluaran dari DSK.

## **BAB V**

### **KESIMPULAN**

Berdasarkan analisa yang dilakukan pada penelitian ini diperoleh beberapa kesimpulan:

1. Pemancar OFDM yang dibangun pada DSK telah berhasil dan berjalan sesuai dengan teori dan tujuan.
2. Sinyal keluaran dari LINE OUT DSK pada penelitian ini telah sesuai dengan yang diharapkan berupa sinyal hasil modulasi Quadrature Amplitude Modulation (QAM).
3. Aliran data dari pemancar OFDM yang dibangun yang diukur pada workspace dari MATLAB telah memberikan hasil yang sesuai dengan teori dan tujuan.

## DAFTAR ACUAN

- [1] Arman, D.P., "Rancang Bangun Rangkaian OFDM dengan Menggunakan DSK TMS C6713 Berbasis SIMULINK". Proposal Grant Riset ICT. Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia., 2007
- [2] Charan Langton (2004). *Orthogonal Frequency Division Multiplex (OFDM) Tutorial*. Diakses 11 Februari 2008, dari Complex to Real.  
<http://www.complextoreal.com>
- [3] Luis Intini, Anibal. "Orthogonal Frequency Division Multiplexing for Wireless Networks". Electrical and Computer Engineering Department University of California - Santa Barbara, 2000.

## DAFTAR PUSTAKA

Schulze, Henrik, Christian Luders, *Theory and Application of OFDM and CDMA*  
(Chichester: John Wiley & Sons, 2005)

S. Haykin, *Communication Systems*, 4<sup>th</sup> ed (New York: John Wiley & Sons, 2001)

Mark, Jon W., Weihua Zhuang, *Wireless Communication and Networking* (New  
Jersey: Pearson Education, Inc, 2003)

Ifeachor. Emmanuel C., Barrie W. Jervis, *Digital Signal Processing*, 2nd ed (New  
Jersey: Prentice Hall, 2001)

# LAMPIRAN

## Lampiran 1:

Tabel Aliran Data untuk  $\Delta t = 1$  detik

elemen	Integer	Mapping	Modulasi	IFFT	ACP
1					0.073828 + 0.0058594i
2					-0.0011708 - 0.040306i
3					0.050448 - 0.00446i
4					-0.081157 + 0.035412i
5					-0.073817 - 0.029611i
6					0.029806 - 0.0045455i
7					0.038341 - 0.029274i
8					-0.014028 + 0.024689i
9					0.081309 + 0.097113i
10					-0.047314 - 0.034416i
11					-0.022023 - 0.065277i
12					0.01696 - 0.022464i
13					0.0077832 + 0.011522i
14					-0.044186 + 0.037059i
15					0.0074525 - 0.0021534i
16					-0.01349 + 0.031182i
17					0.021404 - 0.023799i
18					0.017297 - 0.024683i
19					0.04781 + 0.0066912i
20					-0.037274 + 0.0069036i
21					0.0082512 - 0.008361i
22					-0.013154 + 0.025354i
23					-0.10243 - 0.023133i
24					0.063177 - 0.019136i
25					0.011027 + 0.039185i
26					0.045573 - 0.0056358i
27					0.031246 + 0.010713i
28					-0.037994 - 0.020685i
29					-0.038988 - 0.020978i
30					7.7715e-005 + 0.021419i
31					-0.011999 + 0.018637i
32					0.027896 - 0.034439i
33					0.005552 + 0.025028i
34					-0.024357 - 0.0092074i
35					0.041313 - 0.0096827i
36					-0.026322 + 0.030397i
37					0.014434 - 0.021674i
38					0.023133 + 0.0065565i
39					-0.058521 + 0.010084i
40					0.021953 - 0.048703i
41					0.0024851 + 0.025907i

42					-0.012513 + 0.04166i
43					-0.019586 - 0.046004i
44					0.047779 - 0.019299i
45					0.0035383 + 0.080781i
46					0.014837 - 0.048423i
47					-0.058997 + 0.00016419i
48					0.054185 - 0.012125i
49					-0.032597 - 0.01347i
50					-0.04744 + 0.047422i
51					0.047755 - 0.034421i
52					0.0096966 + 0.015735i
53					0.011615 + 0.022326i
54					-0.0048364 - 0.052272i
55					-0.020717 + 0.034528i
56					0.011592 + 0.0019318i
57					0.011689 - 0.0087413i
58					-0.018603 - 0.017403i
59					-0.022372 + 0.02779i
60					0.020628 - 0.037399i
61					-0.0022873 + 0.030597i
62					0.02547 - 0.0090245i
63					-0.030742 + 0.06294i
64					0.027739 - 0.073057i
65			0	0.0097656 - 0.0082031i	0.0097656 - 0.0082031i
66			0	-0.038048 - 0.026865i	-0.038048 - 0.026865i
67			0	-0.011425 + 0.066521i	-0.011425 + 0.066521i
68			0	0.046744 + 0.043903i	0.046744 + 0.043903i
69			0	-0.027007 - 0.046223i	-0.027007 - 0.046223i
70			0	0.0091414 - 0.053075i	0.0091414 - 0.053075i
71			0	-0.053149 - 0.010411i	-0.053149 - 0.010411i
72			0	0.056548 + 0.051758i	0.056548 + 0.051758i
73			0	0.0014114 + 0.019404i	0.0014114 + 0.019404i
74			0	0.034336 + 0.018292i	0.034336 + 0.018292i
75			0	-0.021213 - 0.05259i	-0.021213 - 0.05259i
76			0	-0.035846 + 0.045736i	-0.035846 + 0.045736i
77			0	0.0055276 - 0.052372i	0.0055276 - 0.052372i
78			0	-0.038748 - 0.040905i	-0.038748 - 0.040905i
79			0	0.018281 + 0.0085237i	0.018281 + 0.0085237i
80			0	0.021655 + 0.10503i	0.021655 + 0.10503i
81			0	0.05468 - 0.014626i	0.05468 - 0.014626i
82			0	0.014696 - 0.021846i	0.014696 - 0.021846i
83			0	-0.084838 - 0.011663i	-0.084838 - 0.011663i
84			0	0.021169 - 0.0072077i	0.021169 - 0.0072077i
85			0	0.012163 - 0.015629i	0.012163 - 0.015629i
86			0	-0.081672 - 0.058678i	-0.081672 - 0.058678i
87			0	0.048918 + 0.08699i	0.048918 + 0.08699i

88				0	0.019336 + 0.013497i	0.019336 + 0.013497i
89				0	0.049978 - 0.0010305i	0.049978 - 0.0010305i
90				0	-0.016035 - 0.021056i	-0.016035 - 0.021056i
91				0	0.00054881 + 0.0090598i	0.00054881 + 0.0090598i
92				0	-0.058598 + 0.021224i	-0.058598 + 0.021224i
93	34	0.15 + 0.45i	0.15 + 0.45i		-0.013666 - 0.040993i	-0.013666 - 0.040993i
94	12	-0.75 - 0.15i	-0.75 - 0.15i		0.043725 - 0.057655i	0.043725 - 0.057655i
95	4	-1.05 - 0.15i	-1.05 - 0.15i		-0.031448 + 0.021191i	-0.031448 + 0.021191i
96	25	-0.15 + 0.75i	-0.15 + 0.75i		0.029394 + 0.058563i	0.029394 + 0.058563i
97	42	0.45 + 0.45i	0.45 + 0.45i		0.026479 + 0.00852i	0.026479 + 0.00852i
98	58	1.05 + 0.45i	1.05 + 0.45i		-0.036814 + 0.031008i	-0.036814 + 0.031008i
99	34	0.15 + 0.45i	0.15 + 0.45i		0.066665 - 0.022905i	0.066665 - 0.022905i
100	63	1.05 - 1.05i	1.05 - 1.05i		-0.056302 - 0.032781i	-0.056302 - 0.032781i
101	35	0.15 + 0.15i	0.15 + 0.15i		-0.058304 - 0.05699i	-0.058304 - 0.05699i
102	13	-0.75 - 0.45i	-0.75 - 0.45i		0.033729 + 0.053906i	0.033729 + 0.053906i
103	62	1.05 - 0.75i	1.05 - 0.75i		0.048161 - 0.028104i	0.048161 - 0.028104i
104	20	-0.45 - 0.15i	-0.45 - 0.15i		-0.026189 + 0.03306i	-0.026189 + 0.03306i
105				1	-0.0028357 + 0.02261i	-0.0028357 + 0.02261i
106	25	-0.15 + 0.75i	-0.15 + 0.75i		0.02544 + 0.014638i	0.02544 + 0.014638i
107	30	-0.15 - 0.75i	-0.15 - 0.75i		-0.011459 - 0.043418i	-0.011459 - 0.043418i
108	14	-0.75 - 0.75i	-0.75 - 0.75i		-0.018507 + 0.011543i	-0.018507 + 0.011543i
109	55	0.75 - 1.05i	0.75 - 1.05i		-0.019801 + 0.01529i	-0.019801 + 0.01529i
110	41	0.45 + 0.75i	0.45 + 0.75i		0.03262 - 0.022968i	0.03262 - 0.022968i
111	22	-0.45 - 0.75i	-0.45 - 0.75i		0.013509 - 0.060059i	0.013509 - 0.060059i
112	18	-0.45 + 0.45i	-0.45 + 0.45i		-0.003917 + 0.040413i	-0.003917 + 0.040413i
113	7	-1.05 - 1.05i	-1.05 - 1.05i		-0.034055 + 0.011947i	-0.034055 + 0.011947i
114	57	1.05 + 0.75i	1.05 + 0.75i		0.017041 + 0.051066i	0.017041 + 0.051066i
115	52	0.75 - 0.15i	0.75 - 0.15i		-0.02857 + 0.016644i	-0.02857 + 0.016644i
116	22	-0.45 - 0.75i	-0.45 - 0.75i		0.051014 - 0.05345i	0.051014 - 0.05345i
117	19	-0.45 + 0.15i	-0.45 + 0.15i		0.013627 - 0.027335i	0.013627 - 0.027335i
118	45	0.45 - 0.45i	0.45 - 0.45i		-0.0090801 + 0.0098926i	-0.0090801 + 0.0098926i
119	57	1.05 + 0.75i	1.05 + 0.75i		-0.046388 + 0.014823i	-0.046388 + 0.014823i
120	21	-0.45 - 0.45i	-0.45 - 0.45i		-0.022804 - 0.051136i	-0.022804 - 0.051136i
121	7	-1.05 - 1.05i	-1.05 - 1.05i		0.041813 + 0.031125i	0.041813 + 0.031125i
122	54	0.75 - 0.75i	0.75 - 0.75i		0.024943 + 0.036297i	0.024943 + 0.036297i
123	10	-0.75 + 0.45i	-0.75 + 0.45i		-0.020598 + 0.043433i	-0.020598 + 0.043433i
124	51	0.75 + 0.15i	0.75 + 0.15i		0.011754 - 0.027784i	0.011754 - 0.027784i
125	37	0.15 - 0.45i	0.15 - 0.45i		-0.0042432 - 0.031987i	-0.0042432 - 0.031987i
126	38	0.15 - 0.75i	0.15 - 0.75i		0.0035018 - 0.062901i	0.0035018 - 0.062901i
127	42	0.45 + 0.45i	0.45 + 0.45i		-0.041555 + 0.019912i	-0.041555 + 0.019912i
128	31	-0.15 - 1.05i	-0.15 - 1.05i		0.019757 + 0.045964i	0.019757 + 0.045964i
129	48	0.75 + 1.05i	0.75 + 1.05i		0.0035156 + 0.043359i	0.0035156 + 0.043359i
130				-1	0.016081 - 0.052i	0.016081 - 0.052i
131	25	-0.15 + 0.75i	-0.15 + 0.75i		-0.008611 + 0.012673i	-0.008611 + 0.012673i
132	3	-1.05 + 0.15i	-1.05 + 0.15i		0.0075523 - 0.027744i	0.0075523 - 0.027744i
133	40	0.45 + 1.05i	0.45 + 1.05i		0.012789 + 0.019288i	0.012789 + 0.019288i

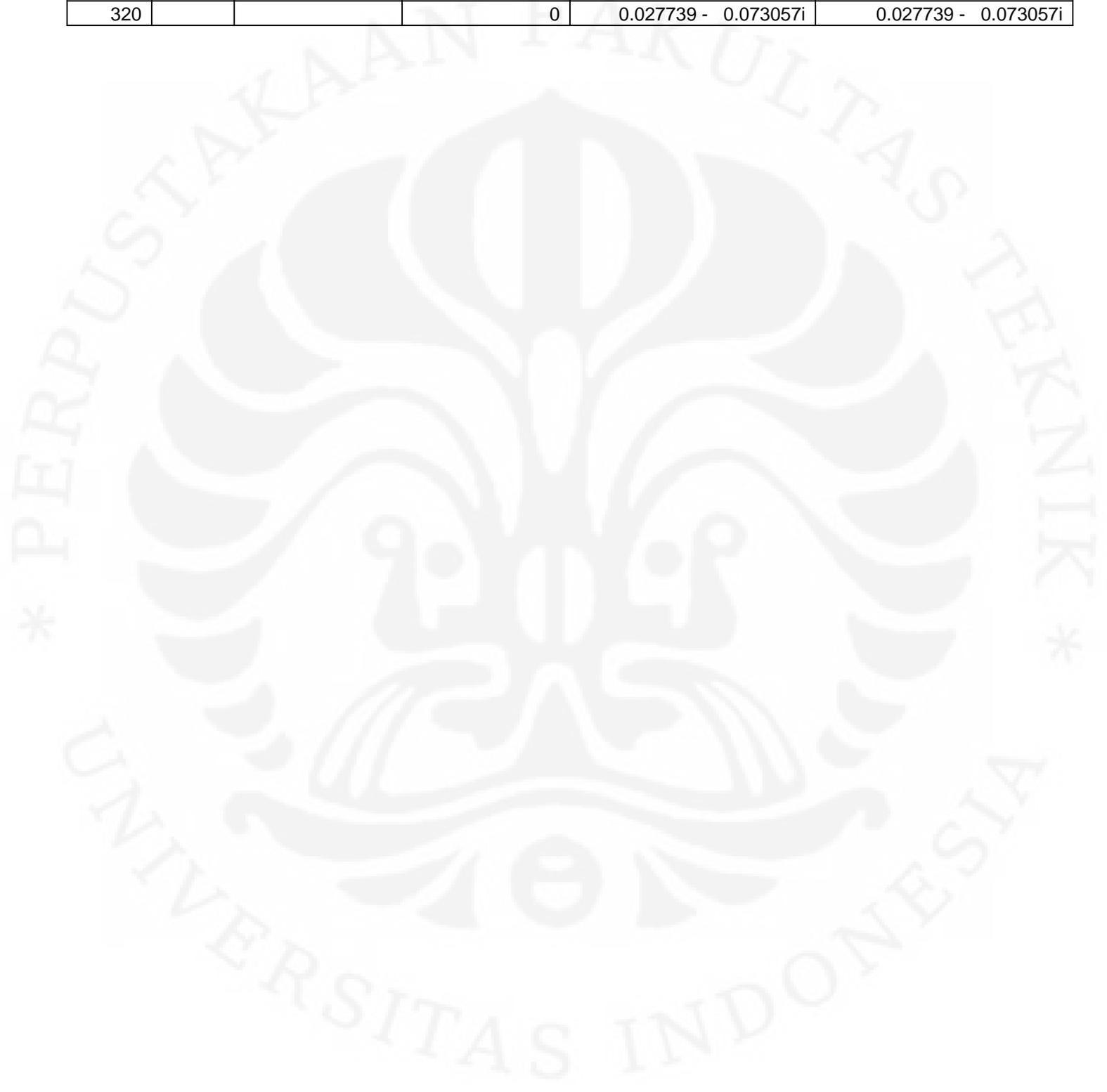
134	36	0.15 - 0.15i	0.15 - 0.15i	-0.020822 + 0.018916i	-0.020822 + 0.018916i
135	7	-1.05 - 1.05i	-1.05 - 1.05i	-0.035773 - 0.014217i	-0.035773 - 0.014217i
136	60	1.05 - 0.15i	1.05 - 0.15i	0.036291 - 0.014487i	0.036291 - 0.014487i
137	27	-0.15 + 0.15i	-0.15 + 0.15i	0.00097944 + 0.0048581i	0.00097944 + 0.0048581i
138	45	0.45 - 0.45i	0.45 - 0.45i	0.0013273 - 0.02135i	0.0013273 - 0.02135i
139	45	0.45 - 0.45i	0.45 - 0.45i	-0.0042786 + 0.0019564i	-0.0042786 + 0.0019564i
140	49	0.75 + 0.75i	0.75 + 0.75i	-0.0098631 + 0.0695i	-0.0098631 + 0.0695i
141	20	-0.45 - 0.15i	-0.45 - 0.15i	0.034801 - 0.017556i	0.034801 - 0.017556i
142	57	1.05 + 0.75i	1.05 + 0.75i	-0.0016013 - 0.035183i	-0.0016013 - 0.035183i
143	32	0.15 + 1.05i	0.15 + 1.05i	-0.04801 + 0.0023718i	-0.04801 + 0.0023718i
144	37	0.15 - 0.45i	0.15 - 0.45i	0.028197 + 0.0076228i	0.028197 + 0.0076228i
145	16	-0.45 + 1.05i	-0.45 + 1.05i	-0.030835 - 0.023811i	-0.030835 - 0.023811i
146	7	-1.05 - 1.05i	-1.05 - 1.05i	0.013596 + 0.00681i	0.013596 + 0.00681i
147	61	1.05 - 0.45i	1.05 - 0.45i	0.040252 - 0.0067411i	0.040252 - 0.0067411i
148	40	0.45 + 1.05i	0.45 + 1.05i	0.012653 + 0.026659i	0.012653 + 0.026659i
149	14	-0.75 - 0.75i	-0.75 - 0.75i	-0.076025 + 0.061726i	-0.076025 + 0.061726i
150	13	-0.75 - 0.45i	-0.75 - 0.45i	0.043223 - 0.0614i	0.043223 - 0.0614i
151	35	0.15 + 0.15i	0.15 + 0.15i	0.01729 - 0.016237i	0.01729 - 0.016237i
152	42	0.45 + 0.45i	0.45 + 0.45i	-0.044361 - 0.013911i	-0.044361 - 0.013911i
153	32	0.15 + 1.05i	0.15 + 1.05i	0.030717 + 0.0010164i	0.030717 + 0.0010164i
154	0	-1.05 + 1.05i	-1.05 + 1.05i	-0.033388 - 0.0077333i	-0.033388 - 0.0077333i
155			1	-0.0067837 + 0.071362i	-0.0067837 + 0.071362i
156	5	-1.05 - 0.45i	-1.05 - 0.45i	0.095795 - 0.0014092i	0.095795 - 0.0014092i
157	30	-0.15 - 0.75i	-0.15 - 0.75i	-0.080198 - 0.0693i	-0.080198 - 0.0693i
158	53	0.75 - 0.45i	0.75 - 0.45i	0.028251 + 0.048708i	0.028251 + 0.048708i
159	59	1.05 + 0.15i	1.05 + 0.15i	-0.028693 - 0.037947i	-0.028693 - 0.037947i
160	44	0.45 - 0.15i	0.45 - 0.15i	0.014053 + 0.017553i	0.014053 + 0.017553i
161	57	1.05 + 0.75i	1.05 + 0.75i	-0.011021 + 0.019503i	-0.011021 + 0.019503i
162	24	-0.15 + 1.05i	-0.15 + 1.05i	0.026179 - 0.027011i	0.026179 - 0.027011i
163	7	-1.05 - 1.05i	-1.05 - 1.05i	-0.0025413 - 0.01035i	-0.0025413 - 0.01035i
164	59	1.05 + 0.15i	1.05 + 0.15i	-0.014782 + 0.016169i	-0.014782 + 0.016169i
165	21	-0.45 - 0.45i	-0.45 - 0.45i	-0.0060301 + 0.0071914i	-0.0060301 + 0.0071914i
166	32	0.15 + 1.05i	0.15 + 1.05i	-0.034312 + 0.032894i	-0.034312 + 0.032894i
167	20	-0.45 - 0.15i	-0.45 - 0.15i	0.08485 - 0.022396i	0.08485 - 0.022396i
168	16	-0.45 + 1.05i	-0.45 + 1.05i	-0.03791 - 0.052136i	-0.03791 - 0.052136i
169	41	0.45 + 0.75i	0.45 + 0.75i	0.045476 + 0.053179i	0.045476 + 0.053179i
170	48	0.75 + 1.05i	0.75 + 1.05i	-0.033235 - 0.045058i	-0.033235 - 0.045058i
171	55	0.75 - 1.05i	0.75 - 1.05i	-0.085314 + 0.038608i	-0.085314 + 0.038608i
172	33	0.15 + 0.75i	0.15 + 0.75i	0.014352 - 0.00698i	0.014352 - 0.00698i
173	55	0.75 - 1.05i	0.75 - 1.05i	0.069288 - 0.0095515i	0.069288 - 0.0095515i
174	19	-0.45 + 0.15i	-0.45 + 0.15i	0.058566 + 0.056352i	0.058566 + 0.056352i
175	45	0.45 - 0.45i	0.45 - 0.45i	-0.064908 - 0.05984i	-0.064908 - 0.05984i
176	45	0.45 - 0.45i	0.45 - 0.45i	-0.024936 - 0.011992i	-0.024936 - 0.011992i
177	20	-0.45 - 0.15i	-0.45 - 0.15i	0.032273 + 0.023247i	0.032273 + 0.023247i
178	49	0.75 + 0.75i	0.75 + 0.75i	-0.03148 - 0.017605i	-0.03148 - 0.017605i
179	33	0.15 + 0.75i	0.15 + 0.75i	0.0061982 + 0.045142i	0.0061982 + 0.045142i

180					-1	-0.0057016 - 0.024936i	-0.0057016 - 0.024936i
181	41	0.45 + 0.75i	0.45 + 0.75i	0.45 + 0.75i	0.45 + 0.75i	0.016917 - 0.014919i	0.016917 - 0.014919i
182	5	-1.05 - 0.45i	-1.05 - 0.45i	-1.05 - 0.45i	-1.05 - 0.45i	0.034264 + 0.024777i	0.034264 + 0.024777i
183	59	1.05 + 0.15i	1.05 + 0.15i	1.05 + 0.15i	1.05 + 0.15i	-0.067733 - 0.032972i	-0.067733 - 0.032972i
184	34	0.15 + 0.45i	0.15 + 0.45i	0.15 + 0.45i	0.15 + 0.45i	0.064864 + 0.026468i	0.064864 + 0.026468i
185	49	0.75 + 0.75i	0.75 + 0.75i	0.75 + 0.75i	0.75 + 0.75i	-0.018637 + 0.024711i	-0.018637 + 0.024711i
186	47	0.45 - 1.05i	0.45 - 1.05i	0.45 - 1.05i	0.45 - 1.05i	-0.035676 - 0.048758i	-0.035676 - 0.048758i
187	36	0.15 - 0.15i	0.15 - 0.15i	0.15 - 0.15i	0.15 - 0.15i	0.012898 + 0.02863i	0.012898 + 0.02863i
188	4	-1.05 - 0.15i	-1.05 - 0.15i	-1.05 - 0.15i	-1.05 - 0.15i	0.018192 - 0.065645i	0.018192 - 0.065645i
189	42	0.45 + 0.45i	0.45 + 0.45i	0.45 + 0.45i	0.45 + 0.45i	-0.0036823 + 0.083564i	-0.0036823 + 0.083564i
190	32	0.15 + 1.05i	0.15 + 1.05i	0.15 + 1.05i	0.15 + 1.05i	-0.039641 - 0.0014885i	-0.039641 - 0.0014885i
191	40	0.45 + 1.05i	0.45 + 1.05i	0.45 + 1.05i	0.45 + 1.05i	0.077472 - 0.014893i	0.077472 - 0.014893i
192	14	-0.75 - 0.75i	-0.75 - 0.75i	-0.75 - 0.75i	-0.75 - 0.75i	-0.054238 - 0.046249i	-0.054238 - 0.046249i
193					0	0.0050781 + 0.029297i	0.0050781 + 0.029297i
194	24	-0.15 + 1.05i	-0.15 + 1.05i	-0.15 + 1.05i	-0.15 + 1.05i	-0.00059939 + 0.009261i	-0.00059939 + 0.009261i
195	5	-1.05 - 0.45i	-1.05 - 0.45i	-1.05 - 0.45i	-1.05 - 0.45i	0.0021674 + 0.0087578i	0.0021674 + 0.0087578i
196	21	-0.45 - 0.45i	-0.45 - 0.45i	-0.45 - 0.45i	-0.45 - 0.45i	0.065736 - 0.0018524i	0.065736 - 0.0018524i
197	2	-1.05 + 0.45i	-1.05 + 0.45i	-1.05 + 0.45i	-1.05 + 0.45i	-0.054359 - 0.0048i	-0.054359 - 0.0048i
198	53	0.75 - 0.45i	0.75 - 0.45i	0.75 - 0.45i	0.75 - 0.45i	-0.073168 - 0.047457i	-0.073168 - 0.047457i
199	8	-0.75 + 1.05i	-0.75 + 1.05i	-0.75 + 1.05i	-0.75 + 1.05i	0.018322 + 0.031246i	0.018322 + 0.031246i
200	4	-1.05 - 0.15i	-1.05 - 0.15i	-1.05 - 0.15i	-1.05 - 0.15i	0.061676 + 0.035128i	0.061676 + 0.035128i
201	19	-0.45 + 0.15i	-0.45 + 0.15i	-0.45 + 0.15i	-0.45 + 0.15i	0.063083 - 0.032312i	0.063083 - 0.032312i
202	16	-0.45 + 1.05i	-0.45 + 1.05i	-0.45 + 1.05i	-0.45 + 1.05i	-0.022593 + 0.068932i	-0.022593 + 0.068932i
203	32	0.15 + 1.05i	0.15 + 1.05i	0.15 + 1.05i	0.15 + 1.05i	-0.10052 - 0.092136i	-0.10052 - 0.092136i
204	2	-1.05 + 0.45i	-1.05 + 0.45i	-1.05 + 0.45i	-1.05 + 0.45i	-0.035783 + 0.0041935i	-0.035783 + 0.0041935i
205	16	-0.45 + 1.05i	-0.45 + 1.05i	-0.45 + 1.05i	-0.45 + 1.05i	0.092733 + 0.0045326i	0.092733 + 0.0045326i
206					1	0.02978 + 0.03167i	0.02978 + 0.03167i
207	0	-1.05 + 1.05i	-1.05 + 1.05i	-1.05 + 1.05i	-1.05 + 1.05i	-0.026901 - 0.0086987i	-0.026901 - 0.0086987i
208	31	-0.15 - 1.05i	-0.15 - 1.05i	-0.15 - 1.05i	-0.15 - 1.05i	0.013774 + 0.0082726i	0.013774 + 0.0082726i
209	54	0.75 - 0.75i	0.75 - 0.75i	0.75 - 0.75i	0.75 - 0.75i	-0.053062 + 0.020049i	-0.053062 + 0.020049i
210	17	-0.45 + 0.75i	-0.45 + 0.75i	-0.45 + 0.75i	-0.45 + 0.75i	0.0045613 + 0.011065i	0.0045613 + 0.011065i
211	35	0.15 + 0.15i	0.15 + 0.15i	0.15 + 0.15i	0.15 + 0.15i	-0.0083451 - 0.093604i	-0.0083451 - 0.093604i
212	30	-0.15 - 0.75i	-0.15 - 0.75i	-0.15 - 0.75i	-0.15 - 0.75i	0.051831 - 0.021918i	0.051831 - 0.021918i
213	48	0.75 + 1.05i	0.75 + 1.05i	0.75 + 1.05i	0.75 + 1.05i	-0.0067394 + 0.094074i	-0.0067394 + 0.094074i
214	47	0.45 - 1.05i	0.45 - 1.05i	0.45 - 1.05i	0.45 - 1.05i	-0.0011411 + 0.049946i	-0.0011411 + 0.049946i
215	41	0.45 + 0.75i	0.45 + 0.75i	0.45 + 0.75i	0.45 + 0.75i	0.029491 - 0.077985i	0.029491 - 0.077985i
216	44	0.45 - 0.15i	0.45 - 0.15i	0.45 - 0.15i	0.45 - 0.15i	-0.095256 - 0.004114i	-0.095256 - 0.004114i
217	52	0.75 - 0.15i	0.75 - 0.15i	0.75 - 0.15i	0.75 - 0.15i	0.040666 + 0.008508i	0.040666 + 0.008508i
218	25	-0.15 + 0.75i	-0.15 + 0.75i	-0.15 + 0.75i	-0.15 + 0.75i	0.0081824 + 0.025529i	0.0081824 + 0.025529i
219	7	-1.05 - 1.05i	-1.05 - 1.05i	-1.05 - 1.05i	-1.05 - 1.05i	-0.01533 - 0.036378i	-0.01533 - 0.036378i
220	58	1.05 + 0.45i	1.05 + 0.45i	1.05 + 0.45i	1.05 + 0.45i	0.012904 + 0.01575i	0.012904 + 0.01575i
221	20	-0.45 - 0.15i	-0.45 - 0.15i	-0.45 - 0.15i	-0.45 - 0.15i	0.04596 - 0.06017i	0.04596 - 0.06017i
222	0	-1.05 + 1.05i	-1.05 + 1.05i	-1.05 + 1.05i	-1.05 + 1.05i	-0.0082669 + 0.091874i	-0.0082669 + 0.091874i
223	22	-0.45 - 0.75i	-0.45 - 0.75i	-0.45 - 0.75i	-0.45 - 0.75i	-0.045057 + 0.0068135i	-0.045057 + 0.0068135i
224	5	-1.05 - 0.45i	-1.05 - 0.45i	-1.05 - 0.45i	-1.05 - 0.45i	-0.004679 - 0.016823i	-0.004679 - 0.016823i
225	52	0.75 - 0.15i	0.75 - 0.15i	0.75 - 0.15i	0.75 - 0.15i	0.043052 - 0.038989i	0.043052 - 0.038989i

226	15	-0.75 - 1.05i	-0.75 - 1.05i	-0.10888 - 0.010983i	-0.10888 - 0.010983i
227	19	-0.45 + 0.15i	-0.45 + 0.15i	0.08057 + 0.00074537i	0.08057 + 0.00074537i
228	55	0.75 - 1.05i	0.75 - 1.05i	0.061156 + 0.027557i	0.061156 + 0.027557i
229	34	0.15 + 0.45i	0.15 + 0.45i	-0.049442 + 0.058762i	-0.049442 + 0.058762i
230	56	1.05 + 1.05i	1.05 + 1.05i	-0.0043653 - 0.097547i	-0.0043653 - 0.097547i
231			1	0.0015754 + 0.057719i	0.0015754 + 0.057719i
232	30	-0.15 - 0.75i	-0.15 - 0.75i	-0.024552 - 0.017438i	-0.024552 - 0.017438i
233	56	1.05 + 1.05i	1.05 + 1.05i	-0.0044086 - 0.012634i	-0.0044086 - 0.012634i
234	39	0.15 - 1.05i	0.15 - 1.05i	0.013919 + 0.0030631i	0.013919 + 0.0030631i
235	0	-1.05 + 1.05i	-1.05 + 1.05i	0.037954 + 0.0046073i	0.037954 + 0.0046073i
236	11	-0.75 + 0.15i	-0.75 + 0.15i	-0.030403 + 0.011566i	-0.030403 + 0.011566i
237	43	0.45 + 0.15i	0.45 + 0.15i	0.0071488 - 0.0093468i	0.0071488 - 0.0093468i
238	11	-0.75 + 0.15i	-0.75 + 0.15i	-0.025675 + 0.0062086i	-0.025675 + 0.0062086i
239	56	1.05 + 1.05i	1.05 + 1.05i	0.040054 + 0.0081059i	0.040054 + 0.0081059i
240	51	0.75 + 0.15i	0.75 + 0.15i	-0.03736 - 0.046286i	-0.03736 - 0.046286i
241	31	-0.15 - 1.05i	-0.15 - 1.05i	0.035941 + 0.011089i	0.035941 + 0.011089i
242	45	0.45 - 0.45i	0.45 - 0.45i	-0.042291 + 0.067485i	-0.042291 + 0.067485i
243	20	-0.45 - 0.15i	-0.45 - 0.15i	0.034084 - 0.022814i	0.034084 - 0.022814i
244	29	-0.15 - 0.45i	-0.15 - 0.45i	-0.00076852 - 0.040811i	-0.00076852 - 0.040811i
245	33	0.15 + 0.75i	0.15 + 0.75i	-0.00057319 - 0.00032371i	-0.00057319 - 0.00032371i
246	39	0.15 - 1.05i	0.15 - 1.05i	-0.0091969 - 0.002218i	-0.0091969 - 0.002218i
247	35	0.15 + 0.15i	0.15 + 0.15i	-0.017959 + 0.034872i	-0.017959 + 0.034872i
248	22	-0.45 - 0.75i	-0.45 - 0.75i	-0.034382 - 0.032743i	-0.034382 - 0.032743i
249	12	-0.75 - 0.15i	-0.75 - 0.15i	0.057748 + 0.027101i	0.057748 + 0.027101i
250	9	-0.75 + 0.75i	-0.75 + 0.75i	0.068695 + 0.052422i	0.068695 + 0.052422i
251	6	-1.05 - 0.75i	-1.05 - 0.75i	-0.063603 - 0.06589i	-0.063603 - 0.06589i
252	56	1.05 + 1.05i	1.05 + 1.05i	-0.0041156 - 0.022918i	-0.0041156 - 0.022918i
253	49	0.75 + 0.75i	0.75 + 0.75i	-0.028914 - 0.026532i	-0.028914 - 0.026532i
254	37	0.15 - 0.45i	0.15 - 0.45i	-0.0066917 + 0.087177i	-0.0066917 + 0.087177i
255	46	0.45 - 0.75i	0.45 - 0.75i	0.009758 - 0.055601i	0.009758 - 0.055601i
256			1	-0.021485 + 0.046676i	-0.021485 + 0.046676i
257	51	0.75 + 0.15i	0.75 + 0.15i	0.073828 + 0.0058594i	0.073828 + 0.0058594i
258	39	0.15 - 1.05i	0.15 - 1.05i	-0.0011708 - 0.040306i	-0.0011708 - 0.040306i
259	46	0.45 - 0.75i	0.45 - 0.75i	0.050448 - 0.00446i	0.050448 - 0.00446i
260	47	0.45 - 1.05i	0.45 - 1.05i	-0.081157 + 0.035412i	-0.081157 + 0.035412i
261	47	0.45 - 1.05i	0.45 - 1.05i	-0.073817 - 0.029611i	-0.073817 - 0.029611i
262	13	-0.75 - 0.45i	-0.75 - 0.45i	0.029806 - 0.0045455i	0.029806 - 0.0045455i
263	13	-0.75 - 0.45i	-0.75 - 0.45i	0.038341 - 0.029274i	0.038341 - 0.029274i
264	40	0.45 + 1.05i	0.45 + 1.05i	-0.014028 + 0.024689i	-0.014028 + 0.024689i
265	31	-0.15 - 1.05i	-0.15 - 1.05i	0.081309 + 0.097113i	0.081309 + 0.097113i
266	50	0.75 + 0.45i	0.75 + 0.45i	-0.047314 - 0.034416i	-0.047314 - 0.034416i
267	14	-0.75 - 0.75i	-0.75 - 0.75i	-0.022023 - 0.065277i	-0.022023 - 0.065277i
268	61	1.05 - 0.45i	1.05 - 0.45i	0.01696 - 0.022464i	0.01696 - 0.022464i
269	10	-0.75 + 0.45i	-0.75 + 0.45i	0.0077832 + 0.011522i	0.0077832 + 0.011522i
270	45	0.45 - 0.45i	0.45 - 0.45i	-0.044186 + 0.037059i	-0.044186 + 0.037059i
271	13	-0.75 - 0.45i	-0.75 - 0.45i	0.0074525 - 0.0021534i	0.0074525 - 0.0021534i

272	4	-1.05 - 0.15i	-1.05 - 0.15i	-0.01349 + 0.031182i	-0.01349 + 0.031182i
273	11	-0.75 + 0.15i	-0.75 + 0.15i	0.021404 - 0.023799i	0.021404 - 0.023799i
274	63	1.05 - 1.05i	1.05 - 1.05i	0.017297 - 0.024683i	0.017297 - 0.024683i
275	10	-0.75 + 0.45i	-0.75 + 0.45i	0.04781 + 0.0066912i	0.04781 + 0.0066912i
276	53	0.75 - 0.45i	0.75 - 0.45i	-0.037274 + 0.0069036i	-0.037274 + 0.0069036i
277	14	-0.75 - 0.75i	-0.75 - 0.75i	0.0082512 - 0.008361i	0.0082512 - 0.008361i
278	29	-0.15 - 0.45i	-0.15 - 0.45i	-0.013154 + 0.025354i	-0.013154 + 0.025354i
279	5	-1.05 - 0.45i	-1.05 - 0.45i	-0.10243 - 0.023133i	-0.10243 - 0.023133i
280	4	-1.05 - 0.15i	-1.05 - 0.15i	0.063177 - 0.019136i	0.063177 - 0.019136i
281			1	0.011027 + 0.039185i	0.011027 + 0.039185i
282	30	-0.15 - 0.75i	-0.15 - 0.75i	0.045573 - 0.0056358i	0.045573 - 0.0056358i
283	10	-0.75 + 0.45i	-0.75 + 0.45i	0.031246 + 0.010713i	0.031246 + 0.010713i
284	12	-0.75 - 0.15i	-0.75 - 0.15i	-0.037994 - 0.020685i	-0.037994 - 0.020685i
285	40	0.45 + 1.05i	0.45 + 1.05i	-0.038988 - 0.020978i	-0.038988 - 0.020978i
286	17	-0.45 + 0.75i	-0.45 + 0.75i	7.7715e-005 + 0.021419i	7.7715e-005 + 0.021419i
287	12	-0.75 - 0.15i	-0.75 - 0.15i	-0.011999 + 0.018637i	-0.011999 + 0.018637i
288	44	0.45 - 0.15i	0.45 - 0.15i	0.027896 - 0.034439i	0.027896 - 0.034439i
289	49	0.75 + 0.75i	0.75 + 0.75i	0.005552 + 0.025028i	0.005552 + 0.025028i
290	48	0.75 + 1.05i	0.75 + 1.05i	-0.024357 - 0.0092074i	-0.024357 - 0.0092074i
291	31	-0.15 - 1.05i	-0.15 - 1.05i	0.041313 - 0.0096827i	0.041313 - 0.0096827i
292	15	-0.75 - 1.05i	-0.75 - 1.05i	-0.026322 + 0.030397i	-0.026322 + 0.030397i
293	38	0.15 - 0.75i	0.15 - 0.75i	0.014434 - 0.021674i	0.014434 - 0.021674i
294			0	0.023133 + 0.0065565i	0.023133 + 0.0065565i
295			0	-0.058521 + 0.010084i	-0.058521 + 0.010084i
296			0	0.021953 - 0.048703i	0.021953 - 0.048703i
297			0	0.0024851 + 0.025907i	0.0024851 + 0.025907i
298			0	-0.012513 + 0.04166i	-0.012513 + 0.04166i
299			0	-0.019586 - 0.046004i	-0.019586 - 0.046004i
300			0	0.047779 - 0.019299i	0.047779 - 0.019299i
301			0	0.0035383 + 0.080781i	0.0035383 + 0.080781i
302			0	0.014837 - 0.048423i	0.014837 - 0.048423i
303			0	-0.058997 + 0.00016419i	-0.058997 + 0.00016419i
304			0	0.054185 - 0.012125i	0.054185 - 0.012125i
305			0	-0.032597 - 0.01347i	-0.032597 - 0.01347i
306			0	-0.04744 + 0.047422i	-0.04744 + 0.047422i
307			0	0.047755 - 0.034421i	0.047755 - 0.034421i
308			0	0.0096966 + 0.015735i	0.0096966 + 0.015735i
309			0	0.011615 + 0.022326i	0.011615 + 0.022326i
310			0	-0.0048364 - 0.052272i	-0.0048364 - 0.052272i
311			0	-0.020717 + 0.034528i	-0.020717 + 0.034528i
312			0	0.011592 + 0.0019318i	0.011592 + 0.0019318i
313			0	0.011689 - 0.0087413i	0.011689 - 0.0087413i
314			0	-0.018603 - 0.017403i	-0.018603 - 0.017403i
315			0	-0.022372 + 0.02779i	-0.022372 + 0.02779i
316			0	0.020628 - 0.037399i	0.020628 - 0.037399i
317			0	-0.0022873 + 0.030597i	-0.0022873 + 0.030597i

318			0	$0.02547 - 0.0090245i$	$0.02547 - 0.0090245i$
319			0	$-0.030742 + 0.06294i$	$-0.030742 + 0.06294i$
320			0	$0.027739 - 0.073057i$	$0.027739 - 0.073057i$



Lampiran 2:

Tabel Perbandingan FFT dengan Modulasi untuk  $\Delta t = 1$  detik

elemen	Modulasi		IFFT	FFT
1	0		0.0097656 - 0.0082031i	-2.7756e-017 -8.3267e-017i
2	0		-0.038048 - 0.026865i	1.3878e-016 +1.1102e-016i
3	0		-0.011425 + 0.066521i	-2.2204e-016 +5.5511e-017i
4	0		0.046744 + 0.043903i	-2.7756e-016 -8.3267e-017i
5	0		-0.027007 - 0.046223i	2.2204e-016 - 2.498e-016i
6	0		0.0091414 - 0.053075i	-2.2204e-016 +1.9429e-016i
7	0		-0.053149 - 0.010411i	1.6653e-016 +1.1102e-016i
8	0		0.056548 + 0.051758i	2.2204e-016 + 3.747e-016i
9	0		0.0014114 + 0.019404i	-2.7756e-016 -1.9429e-016i
10	0		0.034336 + 0.018292i	0 +2.2204e-016i
11	0		-0.021213 - 0.05259i	0
12	0		-0.035846 + 0.045736i	3.3307e-016 -1.9429e-016i
13	0		0.0055276 - 0.052372i	-1.6653e-016 +3.3307e-016i
14	0		-0.038748 - 0.040905i	-1.1102e-016 +3.3307e-016i
15	0		0.018281 + 0.0085237i	-4,44E-12
16	0		0.021655 + 0.10503i	-1,39E-12
17	0		0.05468 - 0.014626i	2.2204e-016 -5.5511e-017i
18	0		0.014696 - 0.021846i	-2.498e-016 -1.1102e-016i
19	0		-0.084838 - 0.011663i	-4.996e-016 +8.3267e-017i
20	0		0.021169 - 0.0072077i	1.6653e-016 -5.5511e-017i
21	0		0.012163 - 0.015629i	5,55E-13
22	0		-0.081672 - 0.058678i	1.1102e-016 -2.2204e-016i
23	0		0.048918 + 0.08699i	-1.9429e-016 -1.1102e-016i
24	0		0.019336 + 0.013497i	2.7756e-016 -4.0246e-016i
25	0		0.049978 - 0.0010305i	1.1102e-016 -1.3878e-016i
26	0		-0.016035 - 0.021056i	-1.6653e-016 -3.8858e-016i
27	0		0.00054881 + 0.0090598i	3.3307e-016 -2.2204e-016i
28	0		-0.058598 + 0.021224i	-4.4409e-016 +2.7756e-017i
29	0.15 +	0.45i	-0.013666 - 0.040993i	0.15 + 0.45i
30	-0.75 -	0.15i	0.043725 - 0.057655i	-0.75 - 0.15i
31	-1.05 -	0.15i	-0.031448 + 0.021191i	-1.05 - 0.15i
32	-0.15 +	0.75i	0.029394 + 0.058563i	-0.15 + 0.75i
33	0.45 +	0.45i	0.026479 + 0.00852i	0.45 + 0.45i
34	1.05 +	0.45i	-0.036814 + 0.031008i	1.05 + 0.45i
35	0.15 +	0.45i	0.066665 - 0.022905i	0.15 + 0.45i
36	1.05 -	1.05i	-0.056302 - 0.032781i	1.05 - 1.05i
37	0.15 +	0.15i	-0.058304 - 0.05699i	0.15 + 0.15i
38	-0.75 -	0.45i	0.033729 + 0.053906i	-0.75 - 0.45i
39	1.05 -	0.75i	0.048161 - 0.028104i	1.05 - 0.75i
40	-0.45 -	0.15i	-0.026189 + 0.03306i	-0.45 - 0.15i
41		1	-0.0028357 + 0.02261i	1
42	-0.15 +	0.75i	0.02544 + 0.014638i	-0.15 + 0.75i
43	-0.15 -	0.75i	-0.011459 - 0.043418i	-0.15 - 0.75i

44	-0.75 - 0.75i	-0.018507 + 0.011543i	-0.75 - 0.75i
45	0.75 - 1.05i	-0.019801 + 0.01529i	0.75 - 1.05i
46	0.45 + 0.75i	0.03262 - 0.022968i	0.45 + 0.75i
47	-0.45 - 0.75i	0.013509 - 0.060059i	-0.45 - 0.75i
48	-0.45 + 0.45i	-0.003917 + 0.040413i	-0.45 + 0.45i
49	-1.05 - 1.05i	-0.034055 + 0.011947i	-1.05 - 1.05i
50	1.05 + 0.75i	0.017041 + 0.051066i	1.05 + 0.75i
51	0.75 - 0.15i	-0.02857 + 0.016644i	0.75 - 0.15i
52	-0.45 - 0.75i	0.051014 - 0.05345i	-0.45 - 0.75i
53	-0.45 + 0.15i	0.013627 - 0.027335i	-0.45 + 0.15i
54	0.45 - 0.45i	-0.0090801 + 0.0098926i	0.45 - 0.45i
55	1.05 + 0.75i	-0.046388 + 0.014823i	1.05 + 0.75i
56	-0.45 - 0.45i	-0.022804 - 0.051136i	-0.45 - 0.45i
57	-1.05 - 1.05i	0.041813 + 0.031125i	-1.05 - 1.05i
58	0.75 - 0.75i	0.024943 + 0.036297i	0.75 - 0.75i
59	-0.75 + 0.45i	-0.020598 + 0.043433i	-0.75 + 0.45i
60	0.75 + 0.15i	0.011754 - 0.027784i	0.75 + 0.15i
61	0.15 - 0.45i	-0.0042432 - 0.031987i	0.15 - 0.45i
62	0.15 - 0.75i	0.0035018 - 0.062901i	0.15 - 0.75i
63	0.45 + 0.45i	-0.041555 + 0.019912i	0.45 + 0.45i
64	-0.15 - 1.05i	0.019757 + 0.045964i	-0.15 - 1.05i
65	0.75 + 1.05i	0.0035156 + 0.043359i	0.75 + 1.05i
66	-1	0.016081 - 0.052i	-1
67	-0.15 + 0.75i	-0.008611 + 0.012673i	-0.15 + 0.75i
68	-1.05 + 0.15i	0.0075523 - 0.027744i	-1.05 + 0.15i
69	0.45 + 1.05i	0.012789 + 0.019288i	0.45 + 1.05i
70	0.15 - 0.15i	-0.020822 + 0.018916i	0.15 - 0.15i
71	-1.05 - 1.05i	-0.035773 - 0.014217i	-1.05 - 1.05i
72	1.05 - 0.15i	0.036291 - 0.014487i	1.05 - 0.15i
73	-0.15 + 0.15i	0.00097944 + 0.0048581i	-0.15 + 0.15i
74	0.45 - 0.45i	0.0013273 - 0.02135i	0.45 - 0.45i
75	0.45 - 0.45i	-0.0042786 + 0.0019564i	0.45 - 0.45i
76	0.75 + 0.75i	-0.0098631 + 0.0695i	0.75 + 0.75i
77	-0.45 - 0.15i	0.034801 - 0.017556i	-0.45 - 0.15i
78	1.05 + 0.75i	-0.0016013 - 0.035183i	1.05 + 0.75i
79	0.15 + 1.05i	-0.04801 + 0.0023718i	0.15 + 1.05i
80	0.15 - 0.45i	0.028197 + 0.0076228i	0.15 - 0.45i
81	-0.45 + 1.05i	-0.030835 - 0.023811i	-0.45 + 1.05i
82	-1.05 - 1.05i	0.013596 + 0.00681i	-1.05 - 1.05i
83	1.05 - 0.45i	0.040252 - 0.0067411i	1.05 - 0.45i
84	0.45 + 1.05i	0.012653 + 0.026659i	0.45 + 1.05i
85	-0.75 - 0.75i	-0.076025 + 0.061726i	-0.75 - 0.75i
86	-0.75 - 0.45i	0.043223 - 0.0614i	-0.75 - 0.45i
87	0.15 + 0.15i	0.01729 - 0.016237i	0.15 + 0.15i
88	0.45 + 0.45i	-0.044361 - 0.013911i	0.45 + 0.45i
89	0.15 + 1.05i	0.030717 + 0.0010164i	0.15 + 1.05i

90	-1.05 + 1.05i	-0.033388 - 0.0077333i	-1.05 + 1.05i
91	1	-0.0067837 + 0.071362i	1 - 1.3878e-016i
92	-1.05 - 0.45i	0.095795 - 0.0014092i	-1.05 - 0.45i
93	-0.15 - 0.75i	-0.080198 - 0.0693i	-0.15 - 0.75i
94	0.75 - 0.45i	0.028251 + 0.048708i	0.75 - 0.45i
95	1.05 + 0.15i	-0.028693 - 0.037947i	1.05 + 0.15i
96	0.45 - 0.15i	0.014053 + 0.017553i	0.45 - 0.15i
97	1.05 + 0.75i	-0.011021 + 0.019503i	1.05 + 0.75i
98	-0.15 + 1.05i	0.026179 - 0.027011i	-0.15 + 1.05i
99	-1.05 - 1.05i	-0.0025413 - 0.01035i	-1.05 - 1.05i
100	1.05 + 0.15i	-0.014782 + 0.016169i	1.05 + 0.15i
101	-0.45 - 0.45i	-0.0060301 + 0.0071914i	-0.45 - 0.45i
102	0.15 + 1.05i	-0.034312 + 0.032894i	0.15 + 1.05i
103	-0.45 - 0.15i	0.08485 - 0.022396i	-0.45 - 0.15i
104	-0.45 + 1.05i	-0.03791 - 0.052136i	-0.45 + 1.05i
105	0.45 + 0.75i	0.045476 + 0.053179i	0.45 + 0.75i
106	0.75 + 1.05i	-0.033235 - 0.045058i	0.75 + 1.05i
107	0.75 - 1.05i	-0.085314 + 0.038608i	0.75 - 1.05i
108	0.15 + 0.75i	0.014352 - 0.00698i	0.15 + 0.75i
109	0.75 - 1.05i	0.069288 - 0.0095515i	0.75 - 1.05i
110	-0.45 + 0.15i	0.058566 + 0.056352i	-0.45 + 0.15i
111	0.45 - 0.45i	-0.064908 - 0.05984i	0.45 - 0.45i
112	0.45 - 0.45i	-0.024936 - 0.011992i	0.45 - 0.45i
113	-0.45 - 0.15i	0.032273 + 0.023247i	-0.45 - 0.15i
114	0.75 + 0.75i	-0.03148 - 0.017605i	0.75 + 0.75i
115	0.15 + 0.75i	0.0061982 + 0.045142i	0.15 + 0.75i
116	-1	-0.0057016 - 0.024936i	-1 + 3.747e-016i
117	0.45 + 0.75i	0.016917 - 0.014919i	0.45 + 0.75i
118	-1.05 - 0.45i	0.034264 + 0.024777i	-1.05 - 0.45i
119	1.05 + 0.15i	-0.067733 - 0.032972i	1.05 + 0.15i
120	0.15 + 0.45i	0.064864 + 0.026468i	0.15 + 0.45i
121	0.75 + 0.75i	-0.018637 + 0.024711i	0.75 + 0.75i
122	0.45 - 1.05i	-0.035676 - 0.048758i	0.45 - 1.05i
123	0.15 - 0.15i	0.012898 + 0.02863i	0.15 - 0.15i
124	-1.05 - 0.15i	0.018192 - 0.065645i	-1.05 - 0.15i
125	0.45 + 0.45i	-0.0036823 + 0.083564i	0.45 + 0.45i
126	0.15 + 1.05i	-0.039641 - 0.0014885i	0.15 + 1.05i
127	0.45 + 1.05i	0.077472 - 0.014893i	0.45 + 1.05i
128	-0.75 - 0.75i	-0.054238 - 0.046249i	-0.75 - 0.75i
129	0	0.0050781 + 0.029297i	2.7756e-017 - 2.7756e-017i
130	-0.15 + 1.05i	-0.00059939 + 0.009261i	-0.15 + 1.05i
131	-1.05 - 0.45i	0.0021674 + 0.0087578i	-1.05 - 0.45i
132	-0.45 - 0.45i	0.065736 - 0.0018524i	-0.45 - 0.45i
133	-1.05 + 0.45i	-0.054359 - 0.0048i	-1.05 + 0.45i
134	0.75 - 0.45i	-0.073168 - 0.047457i	0.75 - 0.45i
135	-0.75 + 1.05i	0.018322 + 0.031246i	-0.75 + 1.05i

136	-1.05 - 0.15i	0.061676 + 0.035128i	-1.05 - 0.15i
137	-0.45 + 0.15i	0.063083 - 0.032312i	-0.45 + 0.15i
138	-0.45 + 1.05i	-0.022593 + 0.068932i	-0.45 + 1.05i
139	0.15 + 1.05i	-0.10052 - 0.092136i	0.15 + 1.05i
140	-1.05 + 0.45i	-0.035783 + 0.0041935i	-1.05 + 0.45i
141	-0.45 + 1.05i	0.092733 + 0.0045326i	-0.45 + 1.05i
142	1	0.02978 + 0.03167i	1 -1.1102e-016i
143	-1.05 + 1.05i	-0.026901 - 0.0086987i	-1.05 + 1.05i
144	-0.15 - 1.05i	0.013774 + 0.0082726i	-0.15 - 1.05i
145	0.75 - 0.75i	-0.053062 + 0.020049i	0.75 - 0.75i
146	-0.45 + 0.75i	0.0045613 + 0.011065i	-0.45 + 0.75i
147	0.15 + 0.15i	-0.0083451 - 0.093604i	0.15 + 0.15i
148	-0.15 - 0.75i	0.051831 - 0.021918i	-0.15 - 0.75i
149	0.75 + 1.05i	-0.0067394 + 0.094074i	0.75 + 1.05i
150	0.45 - 1.05i	-0.0011411 + 0.049946i	0.45 - 1.05i
151	0.45 + 0.75i	0.029491 - 0.077985i	0.45 + 0.75i
152	0.45 - 0.15i	-0.095256 - 0.004114i	0.45 - 0.15i
153	0.75 - 0.15i	0.040666 + 0.008508i	0.75 - 0.15i
154	-0.15 + 0.75i	0.0081824 + 0.025529i	-0.15 + 0.75i
155	-1.05 - 1.05i	-0.01533 - 0.036378i	-1.05 - 1.05i
156	1.05 + 0.45i	0.012904 + 0.01575i	1.05 + 0.45i
157	-0.45 - 0.15i	0.04596 - 0.06017i	-0.45 - 0.15i
158	-1.05 + 1.05i	-0.0082669 + 0.091874i	-1.05 + 1.05i
159	-0.45 - 0.75i	-0.045057 + 0.0068135i	-0.45 - 0.75i
160	-1.05 - 0.45i	-0.004679 - 0.016823i	-1.05 - 0.45i
161	0.75 - 0.15i	0.043052 - 0.038989i	0.75 - 0.15i
162	-0.75 - 1.05i	-0.10888 - 0.010983i	-0.75 - 1.05i
163	-0.45 + 0.15i	0.08057 + 0.00074537i	-0.45 + 0.15i
164	0.75 - 1.05i	0.061156 + 0.027557i	0.75 - 1.05i
165	0.15 + 0.45i	-0.049442 + 0.058762i	0.15 + 0.45i
166	1.05 + 1.05i	-0.0043653 - 0.097547i	1.05 + 1.05i
167	1	0.0015754 + 0.057719i	1 -5.5511e-016i
168	-0.15 - 0.75i	-0.024552 - 0.017438i	-0.15 - 0.75i
169	1.05 + 1.05i	-0.0044086 - 0.012634i	1.05 + 1.05i
170	0.15 - 1.05i	0.013919 + 0.0030631i	0.15 - 1.05i
171	-1.05 + 1.05i	0.037954 + 0.0046073i	-1.05 + 1.05i
172	-0.75 + 0.15i	-0.030403 + 0.011566i	-0.75 + 0.15i
173	0.45 + 0.15i	0.0071488 - 0.0093468i	0.45 + 0.15i
174	-0.75 + 0.15i	-0.025675 + 0.0062086i	-0.75 + 0.15i
175	1.05 + 1.05i	0.040054 + 0.0081059i	1.05 + 1.05i
176	0.75 + 0.15i	-0.03736 - 0.046286i	0.75 + 0.15i
177	-0.15 - 1.05i	0.035941 + 0.011089i	-0.15 - 1.05i
178	0.45 - 0.45i	-0.042291 + 0.067485i	0.45 - 0.45i
179	-0.45 - 0.15i	0.034084 - 0.022814i	-0.45 - 0.15i
180	-0.15 - 0.45i	-0.00076852 - 0.040811i	-0.15 - 0.45i
181	0.15 + 0.75i	-0.00057319 - 0.00032371i	0.15 + 0.75i

182	0.15 - 1.05i	-0.0091969 - 0.002218i	0.15 - 1.05i
183	0.15 + 0.15i	-0.017959 + 0.034872i	0.15 + 0.15i
184	-0.45 - 0.75i	-0.034382 - 0.032743i	-0.45 - 0.75i
185	-0.75 - 0.15i	0.057748 + 0.027101i	-0.75 - 0.15i
186	-0.75 + 0.75i	0.068695 + 0.052422i	-0.75 + 0.75i
187	-1.05 - 0.75i	-0.063603 - 0.06589i	-1.05 - 0.75i
188	1.05 + 1.05i	-0.0041156 - 0.022918i	1.05 + 1.05i
189	0.75 + 0.75i	-0.028914 - 0.026532i	0.75 + 0.75i
190	0.15 - 0.45i	-0.0066917 + 0.087177i	0.15 - 0.45i
191	0.45 - 0.75i	0.009758 - 0.055601i	0.45 - 0.75i
192	1	-0.021485 + 0.046676i	1 -7.7716e-016i
193	0.75 + 0.15i	0.073828 + 0.0058594i	0.75 + 0.15i
194	0.15 - 1.05i	-0.0011708 - 0.040306i	0.15 - 1.05i
195	0.45 - 0.75i	0.050448 - 0.00446i	0.45 - 0.75i
196	0.45 - 1.05i	-0.081157 + 0.035412i	0.45 - 1.05i
197	0.45 - 1.05i	-0.073817 - 0.029611i	0.45 - 1.05i
198	-0.75 - 0.45i	0.029806 - 0.0045455i	-0.75 - 0.45i
199	-0.75 - 0.45i	0.038341 - 0.029274i	-0.75 - 0.45i
200	0.45 + 1.05i	-0.014028 + 0.024689i	0.45 + 1.05i
201	-0.15 - 1.05i	0.081309 + 0.097113i	-0.15 - 1.05i
202	0.75 + 0.45i	-0.047314 - 0.034416i	0.75 + 0.45i
203	-0.75 - 0.75i	-0.022023 - 0.065277i	-0.75 - 0.75i
204	1.05 - 0.45i	0.01696 - 0.022464i	1.05 - 0.45i
205	-0.75 + 0.45i	0.0077832 + 0.011522i	-0.75 + 0.45i
206	0.45 - 0.45i	-0.044186 + 0.037059i	0.45 - 0.45i
207	-0.75 - 0.45i	0.0074525 - 0.0021534i	-0.75 - 0.45i
208	-1.05 - 0.15i	-0.01349 + 0.031182i	-1.05 - 0.15i
209	-0.75 + 0.15i	0.021404 - 0.023799i	-0.75 + 0.15i
210	1.05 - 1.05i	0.017297 - 0.024683i	1.05 - 1.05i
211	-0.75 + 0.45i	0.04781 + 0.0066912i	-0.75 + 0.45i
212	0.75 - 0.45i	-0.037274 + 0.0069036i	0.75 - 0.45i
213	-0.75 - 0.75i	0.0082512 - 0.008361i	-0.75 - 0.75i
214	-0.15 - 0.45i	-0.013154 + 0.025354i	-0.15 - 0.45i
215	-1.05 - 0.45i	-0.10243 - 0.023133i	-1.05 - 0.45i
216	-1.05 - 0.15i	0.063177 - 0.019136i	-1.05 - 0.15i
217	1	0.011027 + 0.039185i	1 -2.2204e-016i
218	-0.15 - 0.75i	0.045573 - 0.0056358i	-0.15 - 0.75i
219	-0.75 + 0.45i	0.031246 + 0.010713i	-0.75 + 0.45i
220	-0.75 - 0.15i	-0.037994 - 0.020685i	-0.75 - 0.15i
221	0.45 + 1.05i	-0.038988 - 0.020978i	0.45 + 1.05i
222	-0.45 + 0.75i	7.7715e-005 + 0.021419i	-0.45 + 0.75i
223	-0.75 - 0.15i	-0.011999 + 0.018637i	-0.75 - 0.15i
224	0.45 - 0.15i	0.027896 - 0.034439i	0.45 - 0.15i
225	0.75 + 0.75i	0.005552 + 0.025028i	0.75 + 0.75i
226	0.75 + 1.05i	-0.024357 - 0.0092074i	0.75 + 1.05i
227	-0.15 - 1.05i	0.041313 - 0.0096827i	-0.15 - 1.05i

228	-0.75 - 1.05i	-0.026322 + 0.030397i	-0.75 - 1.05i
229	0.15 - 0.75i	0.014434 - 0.021674i	0.15 - 0.75i
230	0	0.023133 + 0.0065565i	3.3307e-016 +2.2204e-016i
231	0	-0.058521 + 0.010084i	1.1102e-016 -5.6899e-016i
232	0	0.021953 - 0.048703i	3.6082e-016 -2.2204e-016i
233	0	0.0024851 + 0.025907i	1.9429e-016 +1.6653e-016i
234	0	-0.012513 + 0.04166i	5.5511e-017 -2.2204e-016i
235	0	-0.019586 - 0.046004i	-1.6653e-016 -1.1102e-016i
236	0	0.047779 - 0.019299i	-3.1919e-016 +1.6653e-016i
237	0	0.0035383 + 0.080781i	-1.1102e-016 -2.2204e-016i
238	0	0.014837 - 0.048423i	-8.3267e-017 -8.3267e-017i
239	0	-0.058997 + 0.00016419i	4.4409e-016 -2.7756e-017i
240	0	0.054185 - 0.012125i	1.3878e-016 + 2.498e-016i
241	0	-0.032597 - 0.01347i	1.3878e-016 +1.6653e-016i
242	0	-0.04744 + 0.047422i	3.3307e-016 +1.6653e-016i
243	0	0.047755 - 0.034421i	-5.5511e-017 -1.1102e-016i
244	0	0.0096966 + 0.015735i	4.4409e-016 -1.5266e-016i
245	0	0.011615 + 0.022326i	-5.5511e-017 -1.1102e-016i
246	0	-0.0048364 - 0.052272i	2.2204e-016 -4.7184e-016i
247	0	-0.020717 + 0.034528i	3.3307e-016 -3.3307e-016i
248	0	0.011592 + 0.0019318i	-6.1062e-016 -8.3267e-017i
249	0	0.011689 - 0.0087413i	1.1102e-016 +2.2204e-016i
250	0	-0.018603 - 0.017403i	-1.3878e-016 +2.2204e-016i
251	0	-0.022372 + 0.02779i	2.2204e-016 -4.1633e-016i
252	0	0.020628 - 0.037399i	-5.5511e-016 +4.0246e-016i
253	0	-0.0022873 + 0.030597i	2.7756e-016 +3.0531e-016i
254	0	0.02547 - 0.0090245i	5,55E-13
255	0	-0.030742 + 0.06294i	3,33E-12
256	0	0.027739 - 0.073057i	6.1062e-016 -3.3307e-016i