

58, *Inovtek*, Volume 4, Nomor 1, April 2014, hlm. 58 - 66

## IMPLEMENTASI FPGA UNTUK SISTEM *SPREAD-SPECTRUM* DAN ANALISA SIMULASI KANAL MULTIPATH MENGGUNAKAN CODE PN GOLD

Anang Budikarso, Arifin  
Jurusan Teknik Telekomunikasi  
Politeknik Elektronika Negeri Surabaya  
Kampus PENS Keputih Sukolilo Surabaya 60111  
e-mail : anang\_bk@pens.ac.id, arifin@pens.ac.id

### Abstrak

Dengan berkembangnya teknologi IC terprogram yang semakin canggih dan modern seperti *FPGA*, *FPA* dan yang lainnya, maka dibuat suatu modul sistem *spread-spectrum* dengan *FPGA* sebagai implementasinya. Dalam penelitian ini dilakukan rancang bangun sistem *spread spectrum* dan simulasi kanal *multipath* berbasis *FPGA* (*Field Programmable Gate Array*), sebagai kode acak semu dipilih jenis *kode Gold*. Untuk melihat kinerja sistem tersebut dilakukan uji coba dan pengukuran hasil terhadap pengaturan penundaan (*delay*) pada lintasan jamaknya. Dari hasil percobaan yang dilakukan sifat kanal frekwensi selektif muncul saat *rms delay spreadnya* 215,723 nS dengan *time symbol* 125 nS, perbedaan yang agak jauh dikarenakan keterbatasan pembuatan delay dalam skala kecil. Modul peralatan yang dibuat dapat digunakan sebagai simulasi secara hardware untuk membantu praktikum *spread-spectrum*, dimana untuk melihat sinyal-sinyal *spreading-despreading*, sinyal-sinyal PN serta simulasi sederhana kanal *multipath.spread spectrum*

**Kata kunci** : *FPGA*, PN Code, *Spreading-Despreading*, *Multipath*

### Abstract

*As the IC program technology continuously develops to be more sophisticated and modern such as FPGA, FPA and others, a module of spread-spectrum system with FPGA as an implemantory is made. For this research, a design of spread-spectrum system and multipath canal simulation with FPGA (Filed Programmable Gate Array) was made and the Gold code type was chosen as a pseudo random code. To observe the performance of the system, a test and result measurement toward the arrangement of delay on its multipath was conducted. The result of the test indicated that the selective canal frequency appeared when the rms delay spread is at 215,732 nS with time symbol of 125 nS: a quite far differences occurred because of the limitation of delay production in a small scale. The tools module made can be used as a hardware simulation to help spread-spectrum practice in which to see the spreading-despreading signals, PN signals and also simple simulation of the spread spectrum multipath canal.*

**Keywords** : *FPGA*, PN Code, *Spreading-Despreading*, *Multipath*

### PENDAHULUAN

Pada era informasi sekarang ini, masyarakat dunia membutuhkan sebuah teknologi sistem komunikasi yang memiliki beberapa kehandalan diantaranya adalah kebal terhadap interferensi, memiliki tingkat keamanan data yang tinggi, bisa bekerja pada S/N yang rendah, dan memiliki kemampuan untuk digunakan secara *multiple acces*.

Atas dasar inilah para ilmuwan mengembangkan sebuah sistem teknologi komunikasi yang saat ini banyak dikenal orang sebagai teknologi *Spread Spectrum*. Prinsip kerja dari sistem ini adalah dengan memadukan

antara data informasi yang akan dikirimkan dengan sebuah PN kode yang berbentuk random atau yang lebih dikenal dengan nama *Pseudo Noise Code* (PN-Code), ini merupakan hasil dari pembangkitan *Pseudo Noise Generator* (PN-Generator).

Teknologi *Spread Spectrum* sendiri memiliki 2 proses penting yang tak terpisahkan antara satu dengan yang lainnya yaitu Proses *Spreading* dan *Despreading*. *FPGA* (*Field Programmable Gate Array*) pada tesis ini adalah sebagai Implementator dari keseluruhan sistem yang di buat dan *software* yang digunakan adalah *Xilinx Foundation*.

## DASAR TEORI

### Spread Spectrum

Komunikasi *spread spectrum* lahir pada pertengahan tahun 1950 dilatar belakangi oleh kebutuhan akan sistem komunikasi yang dapat mengatasi masalah interferensi, jaminan kerahasiaan informasi yang dikirim dan beroperasi pada tingkat S/N (signal to noise ratio) yang rendah.

Pada sistem komunikasi *spread spectrum* dilakukan proses penyebaran (*spreading process*) lebar bidang frekuensi bit informasi oleh sebuah deretan sinyal penyebar yang dikenal dengan *spreading waveforms* atau PN (*pseudo noise*) code. Lebar bidang PN (*pseudo noise*) code jauh lebih besar dibanding dengan lebar bidang bit atau sinyal informasi yang hendak dikirimkan.

Sistem komunikasi *spread spectrum* sebagai salah satu sistem komunikasi digital, memiliki beberapa kelebihan dibandingkan sistem komunikasi konvensional yaitu :

1. Lebih kebal terhadap *jamming*
2. Mampu menekan interferensi
3. Dapat dioperasikan pada level daya yang rendah
4. Kemampuan multiple access secara CDMA (*Code Division Multiple Access*)
5. Kerahasiaan lebih terjamin

### Pembangkitan Pseudo Noise (PN)

#### Kode Pseudonoise Gold

Metode pembangkitan PN *sequences* dengan periode crosskorelasi yang lebih baik dari pada *m*-sequence telah dibuat oleh Gold (1967,1968). Teori Gold membuktikan bahwa pasangan yang cocok dari kode panjang maksimal dengan periode  $L = 2^N - 1$  memiliki tiga nilai fungsi crosskorelasi yaitu  $\{-1, -t(N), t(N)-2\}$  dimana  $t(N) = 2^{(N+1)/2} - 1$  untuk  $N$  bernilai ganjil dan  $t(N) = 2^{(N+2)/2} + 1$  untuk  $N$  bernilai genap. Dua deretan panjang maksimal dengan periode  $L$  dan mempunyai crosskorelasi periodik yang dibatasi harga  $\{-1, -t(N), t(N)-2\}$  dinamakan *deretan preferred*. Dari pasangan deretan preferred

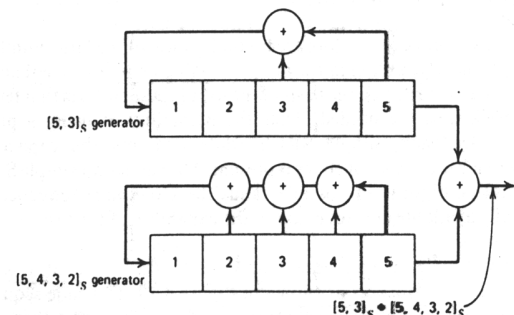
ini kita sebut saja  $a = \{a_1, a_2, a_3, \dots\}$  dan  $b = \{b_1, b_2, b_3, \dots\}$  membentuk suatu deretan dengan mengambil penjumlahan modulo-2 dari  $a$  dengan pergeseran  $N$  siklik dari  $b$ . Sehingga, diperoleh  $N$  deretan periodik yang baru dengan periode  $L = 2^N - 1$ . Jadi dengan memasukkan deretan asal  $a$  dan  $b$  maka jumlah totalnya menjadi  $L = 2$  deretan dan deretan  $L+2$  inilah yang disebut **kode Gold**.

Pembangkitan kode Gold dengan nilai  $N=5$ , sehingga panjang periode  $L = 2^5 - 1 = 31$ . Berdasarkan teori Peterson dan Weldon pasangan preferred yang dapat terbentuk adalah :

$$g_1(D) = 1 + D^3 + D^5$$

$$g_2(D) = 1 + D^2 + D^3 + D^4 + D^5$$

Bentuk blok diagram dari persamaan di atas adalah seperti gambar di bawah ini :



Gambar 1. Ilustrasi Kode Gold  $[5,3]-[5,4,3,2]$

Beberapa keuntungan yang diperoleh dari penggunaan kode ini antara lain adalah :

1. Kode-kode yang ditambahkan melalui proses penambahan *chip per chip* dengan *clock* yang disinkronkan mempunyai panjang periode yang sama, sehingga kedua pembangkitan kode panjang maksimal tersebut dapat menjaga agar fasenya selalu sama.
2. Dengan konfigurasi kode *Gold* akan dihasilkan konfigurasi yang panjangnya ditambahkan dua deretan dari maksimal dasarnya  $L = 2^N - 1$ .
3. Crosskorelasi dari kode-kode panjang maksimalnya yang rendah dan terbatas.

4. Untuk teknologi CDMA yang menginginkan kerja simultan banyak digunakan kode *Gold* ini untuk membawa banyak sinyal pada frekuensi pembawa yang sama.

Untuk memperoleh nilai korelasi yang baik pada sebuah kode *Gold* terdapat tap-tap khusus yang khusus yang disediakan oleh *Gold* (*preferred pair*).

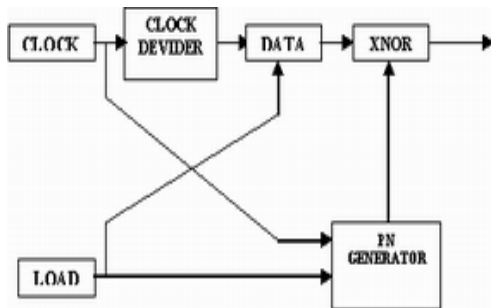
### FPGA

Penggunaan FPGA pada penelitian ini karena fleksibilitas dari device ini. *Flexibel* artinya kita bisa menggunakan *device* ini berdasarkan keinginan, tergantung dari program yang dibuat. Setelah itu di-*download*-kan ke FPGA. Dan proses pengerjaannya melalui *software Xilinx Foundation F2.1i*.

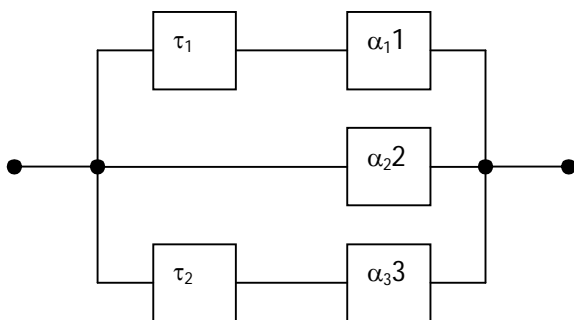
### Desain

#### 1. Blok Diagram

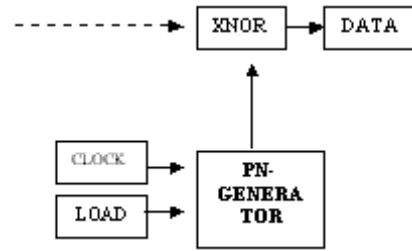
Desain dari sistem yang akan dibuat adalah sebagai berikut :



Gambar 2. Blok Diagram Proses *Spreading*



Gambar 3. Blok Diagram simulasi kanal *multipath*



Gambar 4. Blok Diagram Proses *despreading*

#### 2. Prinsip Kerja

Prinsip kerja dari sistem ini adalah Sinyal pulsa yang masuk kedalam sistem ini yang merupakan data informasi yang telah diinisialisasi sebelumnya akan mengalami proses XNOR-ing dengan kode Pseudonoise yang dibangkitkan. Agar dapat dilakukan proses XNOR-ing ini, *clock* yang dihasilkan harus diolah terlebih dahulu karena frekuensi yang digunakan berbeda. *Pseudonoise Code* mendapatkan *clock* secara langsung dari *oscillator*, sedangkan *clock* yang akan masuk ke data informasi harus dibagi terlebih dahulu dengan *clock divider*. Pembagian *clock* disesuaikan dengan banyaknya jumlah register dari kode *pseudonoise* yang mengkodekan. Misal : untuk *Pseudonoise Code* dengan 5 register maka pembagi jam-nya adalah  $2^5 - 1 = 31$ .

Setelah proses *spreading* kemudian dilewatkan kanal *multipath* yaitu dengan melalui proses delay dan attenuasi baru setelah itu di *despreading* yang merupakan proses kebalikan dari sistem *Spreading*, data informasi yang telah diacak pada sistem *spreading* (sinyal *spread*) dan proses *multipath* di XNOR lagi dengan *PN code*. *Output* dari XNOR merupakan sinyal informasi yang diperoleh dari proses yang terjadi.

#### 3. Komponen Pendukung

Sistem yang dibuat berupa suatu paket *spreading-despreading* yang mempunyai konfigurasi sebagai berikut :

Perangkat keras:

1. FPGA type XC 4005 XL
2. Rangkaian load (monostabil)
3. Rangkaian power supply

Perangkat lunak :

Perangkat lunak yang digunakan dalam membuat sistem ini adalah *Xilinx Foundation*.

### PENGUJIAN DAN PENGUKURAN

#### Peralatan yang digunakan

Dalam pengukuran dibutuhkan alat-alat penunjang untuk melihat sinyal-sinyal yang diproses pada sistem. Alat-alat yang digunakan :

1. FPGA type XC 4005 XL.
2. *Oscilloscope Kenwood CS4125A*.
3. *Logic Analyzer National VP-3621A*.
4. *Spectrum Analyzer Agilent E4403B*.
5. Power supply.
6. *Mono stable multivibrator*.

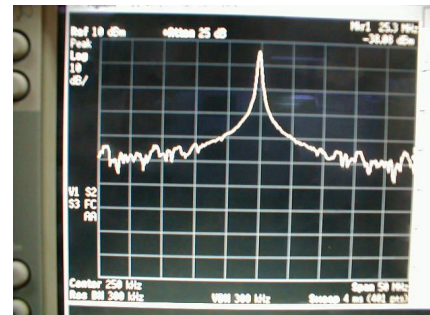
#### Pengujian

Tujuan dari pengujian ini adalah untuk mengetahui kemampuan dari sistem. Frekuensi kode *Pseudonoise* yang digunakan adalah 8 MHz, berasal dari internal IC XILINX XC 4005PC84 (FPGA). Dalam pengujian ini dilakukan juga dengan mengatur laju data hingga sistem tidak mampu lagi memproses data, artinya data hasil proses *despreading* menjadi kacau (terdapat *noise*) atau tidak sama lagi dengan data masukan.

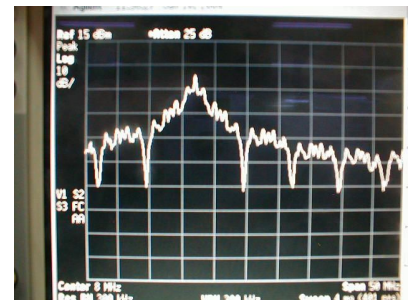
#### Hasil Pengujian



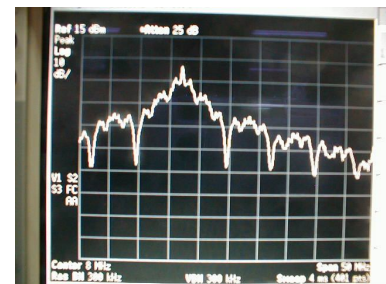
Gambar 5. Hasil Pengujian proses *Spreading-Despreading PN-Code Gold* [5,3]-[5,4,3,2].



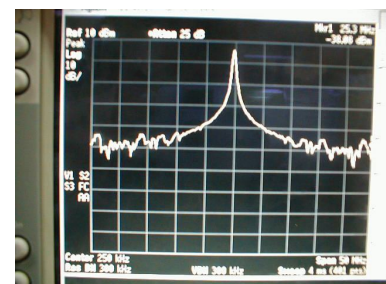
Gambar 6. Hasil pengukuran Data Input



Gambar 7. Hasil pengukuran PN Code



Gambar 8. Hasil pengukuran sinyal spreading

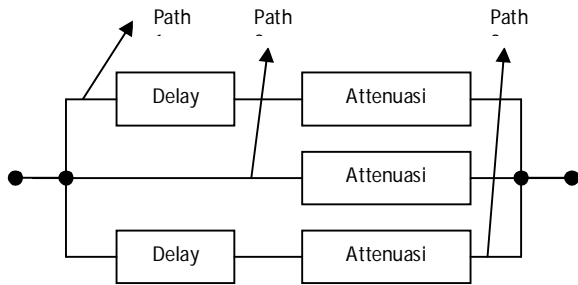


Gambar 9. Hasil pengukuran sinyal Output

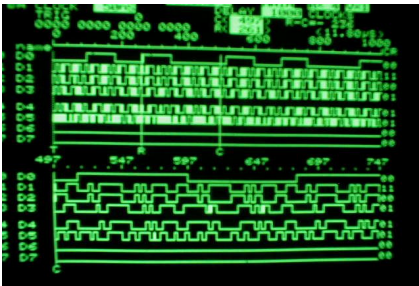
#### Pengukuran di simulasi kanal multipath

1. 3 element path
2. PN Gold 5 register
3. Path 1 di delay = 750 nS, attenuasi = -8,794 dB.

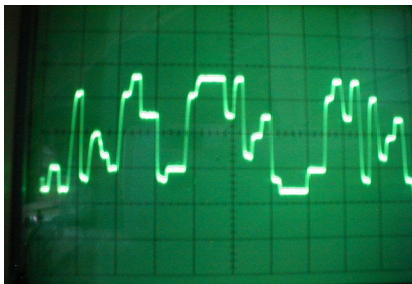
4. Path 2 tanpa di tunda, attenuasi = - 0,830 dB.
5. Path 3 di tunda = 1000 nS, attenuasi = - 13,872 dB.



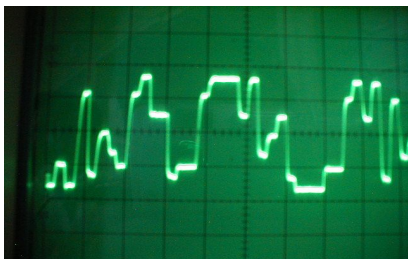
Gambar 10. Diagram rangkaian simulasi kanal multipath



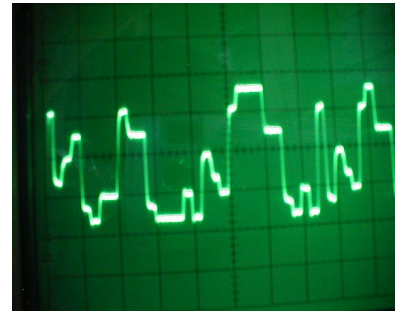
Gambar 11. Hasil pengukuran percobaan 3 di simulasi kanal *multipath* dengan PN Gold [5,3]-[5,4,3,2].



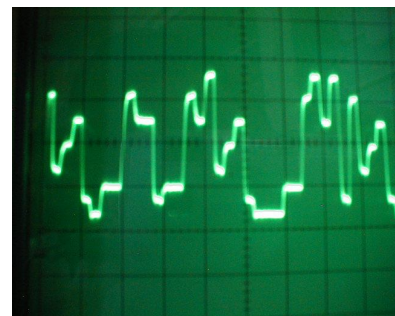
Gambar 12. Hasil pengukuran percobaan 3 path 1.



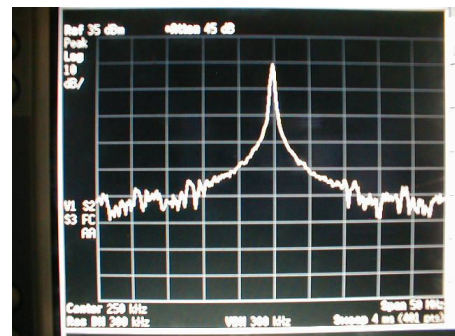
Gambar 13. Hasil pengukuran percobaan 3 path 2.



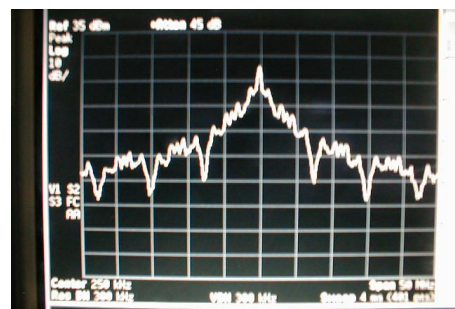
Gambar 14. Hasil pengukuran percobaan 3 path 3.



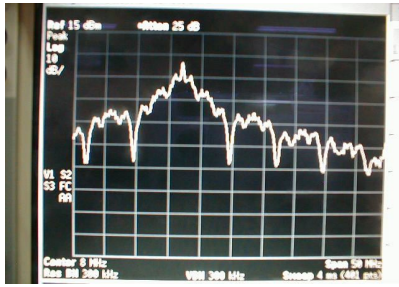
Gambar 15. Hasil pengukuran percobaan 3 penjumlahan ke 3 path.



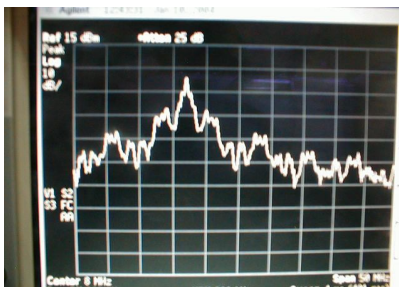
Gambar 16. Hasil pengukuran percobaan 3 Data input.



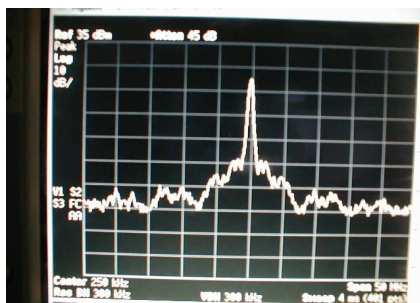
Gambar 17. Hasil pengukuran percobaan 3 PN Code.



Gambar 18. Hasil pengukuran percobaan3 sinyal spreading



Gambar 19. Hasil pengukuran percobaan penggabungan ke 3 sinyal spreading



Gambar 20. Hasil pengukuran percobaan 3 Output

**ANALISA PERHITUNGAN**

1. Laju bit data = 256 kHz = 4 uS.
2. Laju PN = 8 MHz = 125 nS.
3. Laju Spreading = 125 nS.
4. Laju Despreading = 256 kHz = 4 uS.
5. Delay yang digunakan = 40 nS, 80, nS, 250 nS, 500 nS, 750 nS, 1000 nS, 1250 nS, 1500 nS dan 1750 nS.
6. Resistor yang dipasang pada path (untuk redaman) = 5 Ohm, 20 Ohm dan 100 Ohm.
7. Arus keluaran dari pin output = 5mA,
8. Dari perhitungan redaman untuk daya P = 10 log Pout/Pin.

Dari perhitungan diperoleh nilai-nilai besaran arus, daya dan redaman dari masing-masing path seperti dihitung dibawah :

$$i_t = 5 \text{ mA}, R_t = R_{\text{path1}} // R_{\text{path2}} // R_{\text{path3}} = 5 // 20 // 100 = 3,87 \text{ Ohm.}$$

$$P_t = i_t^2 \times R_t = (5)^2 \times 3,87 = 96,8 \text{ mWatt.}$$

$$i_{\text{path1}} = R_{\text{path3}} / (R_{\text{path1}} + R_{\text{path2}} + R_{\text{path3}}) \times i_t = 100 / (5 + 20 + 100) \times 5 = 4 \text{ mA.}$$

$$P_{\text{PATH1}} = I_{\text{PATH1}}^2 \times R_{\text{PATH1}} = (4)^2 \times 5 = 80 \text{ mWatt.}$$

$$P_{\text{att1}} = 10 \log P_{\text{out}} / P_{\text{in}} = 10 \log 80 / 96,8 = 10 \log 0,826 = -0,830 \text{ Db}$$

Dari hasil perhitungan percobaan 1 sampai 7 dapat ditabelkan sebagai berikut :

Tabel 1. Data lintasan jarak percobaan 1

Path	Excess delay (nS)	Resistensi (ohm)	Autensuasi (dB)	Power (mWatt)
1	0	5	-0,830	80
2	40	20	-8,794	12,8
3	250	100	-13,872	4

Sumber : data olahan (2014)

Tabel 2. Data lintasan jarak percobaan 2

Path	Excess delay (nS)	Resistensi (ohm)	Autensuasi (dB)	Power (mWatt)
1	0	5	-0,830	80
2	40	20	-8,794	12,8
3	500	100	-13,872	4

Sumber : data olahan (2014)

Tabel 3. Data lintasan jarak percobaan 3

Path	Excess delay (nS)	Resistensi (ohm)	Autensuasi (dB)	Power (mWatt)
1	0	5	-0,830	80
2	500	20	-8,794	12,8
3	750	100	-13,872	4

Sumber : data olahan (2014)

Tabel 4. Data lintasan jarak percobaan 4

Path	Excess delay (nS)	Resistensi (ohm)	Autensuasi (dB)	Power (mWatt)
1	0	5	-0,830	80
2	750	20	-8,794	12,8
3	1000	100	-13,872	4

Sumber : data olahan (2014)

Tabel 5. Data lintasan jarak percobaan 5

Path	Excess delay (nS)	Resistensi (ohm)	Autensuasi (dB)	Power (mWatt)
1	0	5	-0,830	80
2	1000	20	-8,794	12,8
3	1250	100	-13,872	4

Sumber : data olahan (2014)

Tabel 6. Data lintasan jarak percobaan 6

Path	Excess delay (nS)	Resistensi (ohm)	Autensuasi (dB)	Power (mWatt)
1	0	5	-0,830	80
2	1250	20	-8,794	12,8
3	1500	100	-13,872	4

Sumber : data olahan (2014)

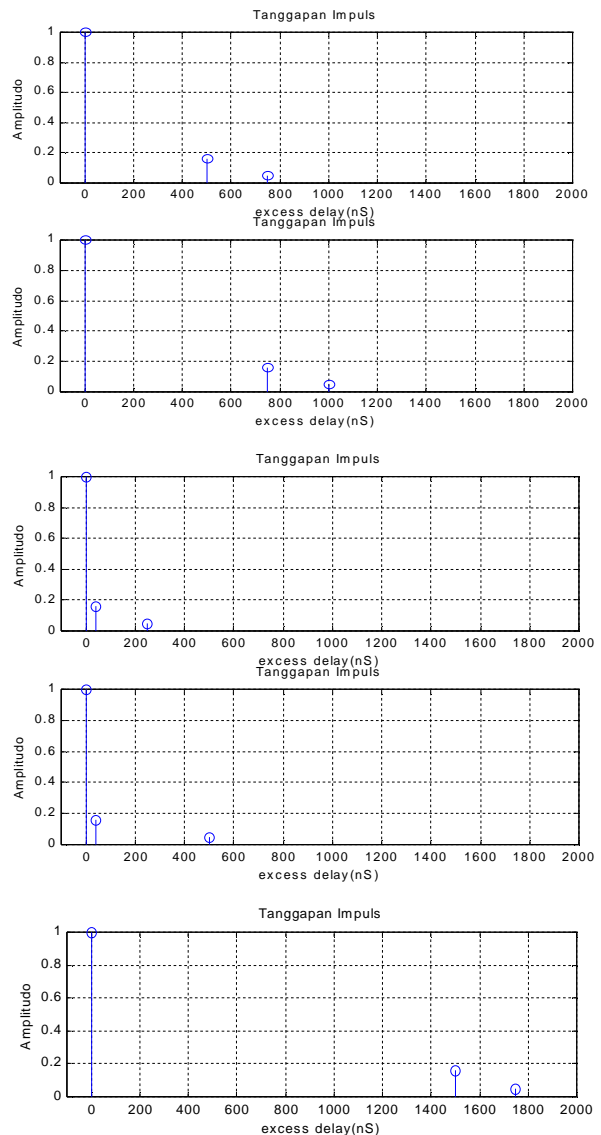
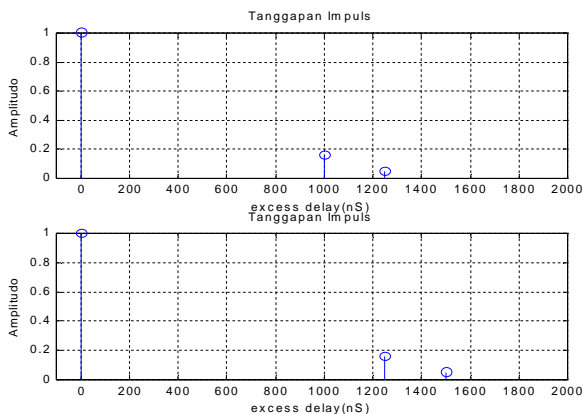
Tabel 7. Data lintasan jarak percobaan 7

Path	Excess delay (nS)	Resistensi (ohm)	Autensuasi (dB)	Power (mWatt)
1	0	5	-0,830	80
2	1500	20	-8,794	12,8
3	1750	100	-13,872	4

Sumber : data olahan (2014)

### RESPON KANAL

Dari hasil perhitungan dapat dibuat respon kanal dari ke-9 sampel seperti pada gambar 21.



Gambar 21. Respon kanal dari ke 7 sampel

### PARAMETER MULTIPATH

#### 1. Maximum Excess Delay

Maximum Excess Delay ini, dapat dihitung dengan menggunakan persamaan :

$$\text{Maximum excess delay} = \tau_{(\max)} - \tau_{(1)}$$

Pada P1,

$$\begin{aligned} \text{Maximum excess delay} &= \tau_{(\max)} - \tau_{(1)} \\ &= 250 \text{ nS} - 0 \text{ nS} = 250 \text{ nS}. \end{aligned}$$

## 2. Mean Excess Delay

*Mean Excess Delay*, dihitung mulai munculnya komponen lintasan jamak pertama sampai terakhir, untuk menghitung dapat digunakan persamaan :

$$\text{Mean Excess delay} = \bar{\tau} = \frac{\frac{1}{N} \sum_{k=1}^N \tau_k P(\tau_k)}{\frac{1}{N} \sum_{k=1}^N P(\tau_k)} = \frac{\sum_{k=1}^N \tau_k P(\tau_k)}{\sum_{k=1}^N P(\tau_k)}$$

$$\text{Mean Excess delay} = \bar{\tau} = \frac{\frac{1}{N} \sum_{k=1}^N \tau_k P(\tau_k)}{\frac{1}{N} \sum_{k=1}^N P(\tau_k)} = \frac{\sum_{k=1}^N \tau_k P(\tau_k)}{\sum_{k=1}^N P(\tau_k)}$$

Berdasarkan data pada Tabel 1, maka *Mean Excess Delay* dapat dihitung sebagai berikut:

$$Pt = \{(0 \times 80) + (40 \times 12,8) + (250 \times 4)\} \times 10^{-18}$$

$$= 0 + 512 + 1000 = 1512 \times 10^{-18}$$

$$P = (80 + 12,8 + 4) \times 10^{-3} = 96,8 \times 10^{-3}$$

$$\text{Mean Excess Delay} = \bar{\tau} = \frac{Pt/N}{P/N} =$$

$$\frac{504 \times 10^{-12}}{32,27 \times 10^{-3}} = 15,618 \times 10^{-9} \text{ Second}$$

## 3. RMS Delay Spread

Berdasarkan data dari tabel 1, *RMS Delay spread* dapat dihitung berdasarkan persamaan :

$$\text{RMS delay spread} = \sigma_{\tau}$$

$$= \left[ \frac{\sum_{k=1}^N (\tau_k - \bar{\tau})^2 P(\tau_k)}{\sum_{k=1}^N P(\tau_k)} \right]^{\frac{1}{2}}$$

Sehingga diperoleh :

$$Pt = \{(-5,618)^2 \times 80 + (24,382)^2 \times 12,8 + (234,382)^2 \times 4\} \times 10^{-21}$$

$$= 246862,81 \times 10^{-21}$$

$$P = (80 + 12,8 + 4) \times 10^{-3} = 96,8 \times 10^{-3}$$

$$\text{RMS Delay Spread} = \sigma_{\tau} = \left[ \frac{Pt/N}{P/N} \right]^{\frac{1}{2}} =$$

$$\left[ \frac{246862,81 \times 10^{-21}}{96,8 \times 10^{-3}} \right]^{\frac{1}{2}} = 50,45 \times 10^{-9} \text{ Second}$$

## 4. Coherence Bandwidth

Coherence bandwidth dapat dihitung dengan persamaan :

$$B_c \approx \frac{1}{5\sigma_{\tau}}$$

Coherence bandwidth =

$$B_c = 1/5 \times (50,45 \times 10^{-9})$$

$$B_c = 3,964 \times 10 \text{ Hz}$$

Setiap percobaan diperoleh parameter-parameter lintasan jamak seperti : *maximum excess delay*, *mean excess delay* dan *rms delay spread* dan dapat ditabelkan seperti dibawah :

Tabel 8. Parameter lintasan jamak P1 sampai P7

Sampel ke	Maximum Excess Delay (nSec)	Mean Excess Delay (nSec)	RMS Delay Spread (nSec)	Coherence Bandwidth (MHz)
1	250	15,618	50,45	3,964
2	500	25,950	99,341	2,013
3	750	97,107	215,723	0,927
4	1000	140,495	309,779	0,645
5	1250	183,884	403,713	0,495
6	1500	227,272	497,929	0,402
7	1750	270,661	592,294	0,338

Sumber : data olahan (2014)

## KESIMPULAN

Berdasarkan pengukuran, perhitungan dan analisa yang dilakukan pada tesis ini, dapat diambil beberapa kesimpulan antara lain :

1. Dari hasil pengukuran pada simulasi sistem *spread spectrum* dapat dilihat dan diperoleh bahwa hasil data inputan dan bentuk *output* tidak mengalami perubahan setelah mengalami proses penebaran



sehingga bisa dianggap simulasi kanal yang digunakan adalah kanal ideal.

2. Pada hasil pengukuran perc.1 dan perc. 2 pada simulasi kanal *multipath*, belum nampak bentuk kanal frekwensi selektifnya, mulai pengukuran percobaan 3 sampai percobaan 7 mulai nampak bentuk kanal frekwensi selektifnya ketika *rms delaynya* 215,732 nS. Luarannya sudah mengalami perubahan dari bentuk inputnya.
3. Jika ditinjau dari analisa perhitungan bentuk kanal frekwensi selektif muncul, bila *rms delay spread* lebih besar 125 nS dan pada percobaan yang dilakukan diperoleh mulai 215,723 nS. Hal ini diperoleh karena keterbatasan dalam menunda komponen lintasan jamaknya.

#### DAFTAR PUSTAKA

- Alaqeeli, A and Starzyk, J (2011) Hardware Implementation for Fast Convolution with a PN Code Using Field Programmable Gate Array" IEEE.
- Budikarso, A., dan M. Syafrudin (2001) Implementasi FPGA pada Digital Logic Trainer", *Proceedings IES*.
- Budikarso, A (2010) Implementasi Verilog HDL pada rangkaian kombinasional dan sekuensial, Penelitian PENS-ITS, Surabaya.
- DJ Ridge (2006) Advance DSP on FPGAs and CPLDs, *proc. ICSPAT*.
- Han, Y., K (2005) Implementation of PIPNLMS on FPGA, *proc. ICSPAT*.
- J. G. Proakis (1989) *Digital Communication. 3th edition*, Mc Graw-Hill Book Company, New York.
- M. K. Simon., J. K Omura, R., A Scultz., and B. K. Levitt (1994) *Spread Spektrum Communication Handbook*, New York, Mc Graw-Hill.
- Santoso, I.,D., dan Endroyono (2001) Simulasi pembangkitan kode-kode acak semu dan evaluasi performansinya dalam aplikasi CDMA.
- Skahill, K (1996) *VHDL for programmable Logic*, Addison-Wesley.
- Rappaport, T., S (1996) *Wireless Communications-Principles & Practice*, IEEE Press, Vol. 1, No. 1, pp 71-131.
- R. L. Peterson, R. E. Ziemer, and D. E Borth (1995) *Introduction to Spread-spektrum Communication*, Prentice Hall, Englewood Cliffs. NJ.
- Robert C.,D (1995) *Spread spektrum Systems with Commercial Application*, 3th Edition, John Wiley and Sons, New York.