National Library of Canada
Canadian Theses Service
Ottawa, Canada
K1A 0N4

Bibliothèque nationale du Canada
Service des thèses canadiennes

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

NL-339 (r. 88/04) c

Canada

UNIVERSITY OF ALBERTA

A STUDY OF THE 3B16P1G PIM CODE FOR A LED SINGLE-MODE OPTICAL FIBER COMMUNICATION SYSTEM.

BY

STEPHEN ROY VANDENBRINK

©

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

EDMONTON, ALBERTA
SPRING 1990

National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

ISBN 0-315-60263-5

NL-339 (r. 88/04) c

Canada

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: STEPHEN ROY VANDENBRINK

TITLE OF THESIS: A STUDY OF THE 3B16P1G PIM CODE FOR A LED SINGLE-MODE OPTICAL FIBER COMMUNICATION SYSTEM.

DEGREE: MASTER OF SCIENCE

YEAR THIS DEGREE GRANTED: 1990

Permission is hereby granted to the UNIVERSITY OF ALBERTA LIBRARY to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

Signature: S. VandenBrink

Permanent Address: #1505 11111 87 Ave.
Edmonton, Alberta.
Canada. T6G 0X9

DATE: April 19th 1990

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled A STUDY OF THE 3B16P1G FIM CODE FOR A LED SINGLE-MODE OPTICAL FIBER COMMUNICATION SYSTEM submitted by STEPHEN ROY VANDENBRINK in partial fulfilment of the requirements for the degree of MASTER OF SCIENCE.

Paul A. Goud
Supervisor - Dr. P.A. Goud

C. G. Englefield.
Supervisor - Dr. C.G. Englefield

Committee - Dr. A.E. Kamal

Vaneldik
Committee - Dr. J.F. Vaneldik

DATE: 12th April 1990

DEDICATION

To my parents

and

my wife, Eleanor

ABSTRACT

To increase the viability of light-emitting diodes (LED) as light sources for single-mode fiber systems, a new family of binary codes has been developed, which would allow the LEDs to be overdriven, thus increasing the peak power coupled into the fiber. This family of codes is based on an asynchronous pulse interval modulation (PIM) scheme, in which the encoded data are represented by the length of the interval contained between the transmitted pulses. In this thesis, the development of the "nB2·2^nPmG" family of PIM codes is discussed and their potential power gain over the NRZ code is determined. After further analysis, the 3B16P1G PIM code was selected to be implemented into hardware in order to test the feasibility of this family of codes. An experimental communication system was constructed to operate at 90 Mbit/s over an electrical link. Because of the high speed of operation, a unique design was used for the asynchronous buffers which are required in the PIM encoder and decoder. This design allows simultaneous read and write operations to the first-in-first-out buffer, with an access time of less than 4 nsec. Experimental measurements were taken to study the effect of threshold level, system bandwidth and SNR on the performance of the system. The effective power gain of the 3B16P1G PIM code for an encoded NRZ 2^{23}-1 PRBS in a bandwidth optimized system was found to be 1.12 or 75% of the maximum theoretical value of 1.5. This degradation in the power gain was due to two factors: the code's error multiplication factor of 12.5 and an optimum bandwidth of 3.45 instead of 3.0 times the optimum bandwidth of the NRZ code. The results demonstrate that the "nB2·2^nPmG" family of PIM codes can be implemented at high speed. It is also shown that these codes are bit sequence independent and have excellent resynchronization characteristics. These features, together with their potential power gain over the NRZ code, make PIM a suitable candidate for short haul LED optical single-mode fiber systems.

ACKNOWLEDGEMENTS

I wish to thank both Dr. C.G. Englefield and Dr. P.A. Goud for their advice, encouragement and guidance during the course of this project. Similarly, I wish to extend my thanks to the technical staff, in particular Bert Telder, Ken Campbell and Dave Rudyk for their technical assistance throughout the project. I would also like to thank the members of the examining committee for reviewing this work.

Appreciation is also extended to the following:

- To the Alberta Telecommunication Research Centre for the use of their facilities and equipment.
- To my fellow graduate students in the optical communication group, in particular Mr. V.A. Anand, Ms. Zhijing Chen, Mr. S. Lohtia and Mr. K.H. Teo for their useful discussions and mutual encouragement.
- The National Science and Engineering Research Council of Canada for the operating grants made to Dr. P.A. Goud and Dr. C.G. Englefield that supported this research work.
- The Department of Electrical Engineering for teaching assistantships.

As well, I would like to express thanks to my parents, wife and all others, whose support has made it possible to complete the work reported in this thesis.

TABLE OF CONTENTS

CHAPTER PAGE

1 INTRODUCTION.....1
  1.1 History.....2
  1.2 Single-Mode Fiber.....5
    1.2.1 Characteristics of Single-Mode Fiber.....6
      1.2.1.1 Attenuation.....6
      1.2.1.2 Chromatic Dispersion.....8
    1.2.2 Types of Single-Mode Fiber.....10
  1.3 Light-Emitting Diodes.....14
    1.3.1 Characteristics of LEDs.....14
      1.3.1.1 Coupled Power.....14
      1.3.1.2 Spectral Width.....15
      1.3.1.3 Frequency Response.....15
      1.3.1.4 Temperature Dependence.....16
    1.3.2 Types of LED.....16
      1.3.2.1 Surface-emitting LED.....16
      1.3.2.2 Edge-emitting LED.....17
      1.3.2.3 Super-radiant LED.....19
  1.4 Coding.....20
    1.4.1 Code Requirements.....21
    1.4.2 Code Selection.....23
      1.4.2.1 Pulse Position Modulation.....27
      1.4.2.2 Pulse Interval Modulation.....30
        1.4.2.2.1 "nB2·2nPmG" PIM.....33

1.5 Thesis Objectives and Organization....38

  1.5.1 Objectives....38

  1.5.2 Organization....39

2 THE 3B16P1G PIM CODE....41

2.1 Coding Rules....41

  2.1.1 Mapping Coding Rules....42

  2.1.2 RDS Coding Rules....44

2.2 Buffer Requirements....47

2.3 Power Spectral Density....54

2.4 Error Multiplication....55

2.5 Theoretical Bit Error Rate....63

  2.5.1 Optimum Threshold Level....67

2.6 Power Gain....71

3. EXPERIMENTAL SETUP....74

3.1 PIM Encoder....74

  3.1.1 Clock Stage....76

  3.1.2 Input Stage....82

  3.1.3 Buffer Stage....84

  3.1.4 Output Stage....92

3.2 PIM Decoder....95

  3.2.1 Clock Stage....98

  3.2.2 Input Stage....101

  3.2.3 Buffer Stage....104

  3.2.4 Output Stage Circuitry....106

3.3 NRZ driver and detector....108

3.4 Channel....110

3.5 Receiver....110

3.5.1 Amplifier 111

3.5.2 Noise Generator Stage 113

3.5.3 Lowpass Filter and the DC Level Shifter 115

4. EXPERIMENTAL RESULTS 120

4.1 Experimental Setup 120

4.3 Coder and Decoder Operation 122

4.3 System Bandwidth 123

4.4 Power Spectrum 127

4.5 Bit Error Rate Measurement 131

4.5.1 Threshold Level 131

4.5.2 Bandwidth 135

4.5.3 SNR 139

4.5.3.1 Error Multiplication Factor 141

4.5.3.2 Power Gain 144

5. SUMMARY AND CONCLUSIONS 148

BIBLIOGRAPHY 155

LIST OF TABLES

Table 2.1 Examples of the mapping assignments for the 3B16P0G and the 3B16P1G PIM code........43

Table 2.2 The threshold violation error analysis of the NRZ data sequence "1111..."........61

Table 2.3 Error multiplication factors for the NRZ data streams "1111..." and "0000..."........64

Table 2.4 Examples of the normalized optimum threshold levels for the 3B16P1G PIM and NRZ binary code........70

Table 3.1 List of the PIM encoder components........78

Table 3.2 Definition of the schematic symbols........79

Table 3.3 State tables for the design of an asynchronous buffer cell........86

Table 3.4 List of the PIM decoder components........99

Table 4.1 The bandwidth of the system........128

LIST OF FIGURES

Figure 1.1 Attenuation characteristics of single-mode fibers.......7

Figure 1.2 Bandwidth-distance characteristics of single-mode fibers....... 11

Figure 1.3 Dispersion characteristics of single-mode fibers.......13

Figure 1.4 Power coupled into a single-mode fiber from an E-LED....... 18

Figure 1.5 Spectral width versus drive current of an E-LED....... 18

Figure 1.6 Peak power versus duty cycle.......25

Figure 1.7 Examples of the NRZ, PPM and PIM code waveforms.......28

Figure 1.8 An example of the "nB2·2nPmG" PIM intervals for a 3-bit sequence.......34

Figure 2.1 Examples of the exact and approximate coding rules for the 3B16P1G PIM code.......46

Figure 2.2 State diagrams for the 3B16P1G PIM code, implemented using the approximate RDS coding rules.......48

Figure 2.3 State diagrams for the 3B16P1G PIM code, implemented using the exact RDS coding rules.......50

Figure 2.4 Probability distribution of the RDS for the approximate coding rules.......52

Figure 2.5 Probability distribution of the RDS for the exact coding rules.......52

Figure 2.6 Impulse power spectrum for the 3B16P1G PIM code using the approximate RDS coding rules.......56

Figure 2.7 Examples of the creation and destruction of data due to errors.......59

Figure 2.8 Error multiplication factors of the 3B16P1G PIM code for the encoded data stream "11111...".......61

Figure 2.9 Normalized optimum threshold level versus the effective duty cycle.......69

Figure 2.10 A comparison of the average power spectra of the 3B16P1G PIM code and the NRZ code....73

Figure 3.1 Block diagram of the 3B16P1G PIM encoder....77

Figure 3.2 The encoder's clock stage circuitry....80

Figure 3.3 The encoder's input stage circuitry....83

Figure 3.4 Karnaugh maps for the design of an asynchronous buffer cell....88

Figure 3.5 The encoder's buffer stage circuitry....91

Figure 3.6 The encoder's output stage circuitry....93

Figure 3.7 Block diagram of the 3B16P1G PIM decoder....97

Figure 3.8 The decoder's clock stage circuitry....100

Figure 3.9 The decoder's input stage circuitry....102

Figure 3.10 The decoder's buffer stage circuitry....105

Figure 3.11 The decoder's output stage circuitry....107

Figure 3.12 The NRZ driver and detector circuitry....109

Figure 3.13 The amplifier circuit....112

Figure 3.14 The three port adder circuit....114

Figure 3.15 The lowpass filter and level shifter circuit....116

Figure 4.1 The experimental setup....121

Figure 4.2 The experimental output waveforms of the 3B16P1G PIM encoder and decoder for the encoded NRZ data sequence "1010..."....124

Figure 4.3 The normalized frequency response of the system for the lowpass filter setting of "L000"....126

Figure 4.4 The experimental power spectrum of the 3B16P1G PIM code for an encoded NRZ 223-1 PRBS....129

Figure 4.5 The theoretical average power spectrum of the 3B16P1G PIM code....130

Figure 4.6 The BER versus the normalized threshold level for the NRZ binary code....133

Figure 4.7 A comparison of the the normalized threshold levels for the NRZ binary and the 3B16P1G PIM codes....133

Figure 4.8 The BER versus the bandwidth of the system....137

Figure 4.9 The BER versus the SNR for the 3B16G1P PIM and the NRZ binary code....140

Figure 4.10 The experimental error multiplication factors for the 3B16P1G PIM code....143

Figure 4.11 The peak power gain versus the BER for the 3B16P1G PIM code....145

Figure 4.12 The effective power gain versus the BER for the 3B16P1G PIM code....145

LIST OF SYMBOLS

A	Amplitude of the signal
B_x	Bandwidth of the code "x"
D_{EFF}	Effective duty cycle
D_x	Duty cycle of the code "x"
E_{m1}	Error multiplication factor
E_{m0}	Error multiplication factor due to threshold violation errors
E_{m1}	Error multiplication factor due to erasure errors
E_{mE}	Effective error multiplication factor
f_H	3 dB upper frequency limit of the system
f_L	3 dB lower frequency limit of the system
m	The number of slots that are required for the guardtime
n	The length in bits of the encoded binary sequence
N	Noise power
N_O	One-sided noise power spectral density
$\overline{N_S}$	The average number of slots contained in the average PIM interval
N_{Sx}	The number of slots contained in the PIM interval
n_x	Reflective index of the material "x"
P_E	Probability of error
P_{E_0}	Probability of error due to threshold violation errors
P_{E_1}	Probability of error due to erasure errors
P_{EG}	Effective power gain
P_{NP}	Noise power penalty
P_{PG}	Peak power gain
P_{Px}	Peak power of the code "x"

P_{Si}	The probability of occurrence of the next RDS state "i"
P_{Sj}	The probability of occurrence of the current RDS state "j"
P_{Dk}	The probability of occurrence of the data sequence "k"
$RDS_{(+)}$	The next value of the RDS
S	Status of the buffer cell
T	Temperature
T_{INT}	The average PIM interval width
T_L	Threshold level
T_{Lopt}	Optimum threshold level
T_{Px}	Pulse width of the code "x"
α	Attenuation of the fiber
λ	wavelength

LIST OF ABBREVIATIONS

AWGN	Additive white guassian noise
BER	Bit error rate
E-LED	Edge-emitting LED
ECL	Emitter-coupled logic
Erfc	Complementary error function
FFT	Fast fourier transform
FIFO	first-in-first-out
FWHM	Full width at half maximum
HPBW	Half-power beam width
IC	Integrated circuit
ISI	Intersymbol interference
LD	Laser diode
LED	Light-emitting diode
MECL	Motorola emitter-coupled logic
MMF	Multi-mode fiber
NA	Numerical aperture
NRZ	Non-return to zero modulation
OH	Hydroxyl ion
PIM	Pulse interval modulation
PPM	Pulse position modulation
PRBS	Pseudo-random binary sequence
RDS	Running digital slip
RMS	Root-mean-square
S-LED	Surface-emitting LED

SMF	Single-mode fiber
SNR	Signal to noise ratio
SNR_P	Peak signal to power ratio
SR-LED	Super-radiant LED

CHAPTER 1

INTRODUCTION

In the last decade, there has been an almost total transition from multi-mode fiber (MMF) to single-mode fiber (SMF) as the standard for optical communications systems, due to its lower dispersion and cost. The decrease in dispersion of the single-mode fiber dramatically increases its information carrying capacity. Initially light-emitting diodes (LED) were not considered as sources for single-mode fiber, because of the small amount of optical power that could be coupled into the fiber. However, in an effort to produce a low cost reliable optical fiber system, researchers have been carrying out research into the use of LEDs as sources for single-mode fiber over the last few years. Optical communication systems using LEDs instead of laser diodes (LD) as sources have a lower launch power and a smaller bit-rate-distance product, causing a decrease in either the transmission distance and/or bit-rate. So, for use in short loop or low bit-rate application, the LED's advantages of higher reliability, higher temperature stability, simpler drive circuitry and lower cost, as compared to a laser diode, outweigh its disadvantages of lower coupled power and wider spectral width.

The four main components of an optical fiber communication system are: the source, the optical channel, the detector and the coding scheme. Each one of these can be modified to improve the overall performance of the system, either by increasing the transmission distance and/or information rate. Since the use of LEDs instead of LDs as sources for single-mode will decrease the performance of the system, we will examine the use of coding techniques to offset this drop in performance.

For a better understanding of optical fiber communication systems, a short history will be presented, followed by an examination of the characteristics of single-mode fiber and LEDs. This information will be then used in the development of a coding scheme that will increase the viability of using LEDs on single-mode fiber.

1.1 History

The major events that have precipitated our current growth in high capacity optical communications were initiated by the invention of the laser thirty years ago in 1960[1]. This spurred research and development into unguided optical communication systems by supplying an appropriate source. These atmospheric optical communication systems suffered from several disadvantages. First, they required a line of sight transmission link between the transmitter and receiver. This limited the range, and the localities in which the system could be installed. Second, the characteristics of the communication link which depend on the atmospheric conditions (weather) will vary over time, reducing the reliability of the system.

In order to overcome these disadvantages, interest was shifted to guided optical communication systems. Because of their large information carrying potential, Bell laboratories in New Jersey redirected research efforts from millimeter-waveguides to guided-wave optical communication in 1963. During this time, their colleagues at Standard Telecommunication laboratories in England also established a guided-wave optical communication group.

Before a practical optical fiber system could be implemented, major obstacles had to be overcome. The first was the fiber. At that time the highest quality glass fiber had transmission losses of approximately 200 dB/km. It had been considered that for fiber systems to be competitive with copper cable, these transmission losses would have to be no greater than 20 dB/km. It took until

1970 before Corning Glass Works reached this goal, by producing a multi-mode fiber with an attenuation loss of less than 20 dB/km.

During the same year another major milestone was achieved by Bell labs researchers — the construction of a continuous semiconductor laser that operated at room temperature. These first LDs operated for over 20 minutes at 38°C. Four months after the announcement of a room temperature LD, results were published on the use of LEDs as sources for optical fiber systems by C. Burrus of Bell Labs. Because of the reliability problems with LDs, Bell Labs concurrently pursued research on LEDs as sources for multi-mode fiber. It took more than five years of further research and development before the reliability of the laser was increased to the point where it could be field tested.

By the late seventies, improvements in optical fibers decreased the attenuation loss at the wavelengths of 0.8 to 0.9 μm, to 2 to 3 dB/km, and the first commercial optical fiber system was installed by General Telephone and Electronics Corporation in 1977. It was a first generation system which consisted of a 0.83 μm LED transmitting at a data-rate of 1.544 Mbit/s over an eight kilometer span of multi-mode fiber.

The fiber's lower attenuation losses at higher wavelengths, as low as 0.2 dB per kilometer at 1.55 μm, and the higher effective bandwidth at the zero dispersion point of 1.3 μm, triggered interest in longer wavelength devices. The first practical long-wavelength LD was produced in 1976 by J. Hsieh of MIT. This shift to longer wavelengths not only gave performance improvements due to better fiber characteristics but also, as it was later discovered, the properties of the materials used in their construction increased the life span and reliability of the devices. This paved the way to second generation optical systems. They consisted of LD or LED sources operated at the zero chromatic dispersion point, 1.3 μm, on multi-mode fiber. The lower attenuation and dispersion in the 1.3 μm transmission window increased the functionality of LEDs in fiber systems. But modal dispersion of the multi-mode fiber limited the maximum performance of these systems due to pulse broadening.

The push for increased bit-rates and transmission span has led to the development of third generation systems. These were initially single-mode fiber systems that used 1.3 μm LDs as sources. The use of LEDs as sources for single-mode fiber was considered impractical due to their low launch power. But as interest grew in the application of single-mode fiber to short haul communication links, research in LEDs was renewed. Using LEDs instead of LDs as a source would decrease cost and increase reliability. Also, the transmission capacity of the system could later be upgraded with LDs as they became more economical or if higher data-rates were required[2,3,4].

Research into the use of LEDs in SMF systems was concentrated on the short haul or local loop domain. To be useful, these systems had to operate at data rates of up to 140 Mbit/s, over a 10 kilometer span. Research in the early 1980s on LEDs was focused on increasing the launch power into the fiber. This was accomplished by increasing the coupling efficiency between the LED and fiber[5,6] or by increasing the LED's brightness[7]. Several experimental LED single-mode fiber systems were built and tested during the mid 1980s that matched or exceeded these goals[8,9,10,11].

By the late 1980s, research on LED single-mode fiber systems had diversified into different areas. One area is the continued improvement in the system's bit-rate and transmission distance. Bit-rates of 1 Gbit/s or greater for distances up to 20 kilometers have been reported[12,13]. These systems are being targeted for high information capacity and medium length communication links. A second area of interest is the transmission of multiple information channels through one fiber. Experimental systems using wavelength division multiplexing have been built and tested[14,15]. These will be important in adding flexibility and functionality to local area networks and local loop applications. Finally, the third area is in the development of LED single mode fiber systems. These projects examine the environmental, hardware compatibility and economic factors that will be encountered in a commercial system[16,17,18]. Research and development in this area is being encouraged by current trials to provide optical fiber to the home. In the first half of 1989 there were plans for at least six different trials across the continental USA, that would eventually connect fiber to over 5,000 homes[19]. This potential market may be well matched to LED single-mode fiber systems in terms of reliability, costs and the initially required data rate.

1.2 Single-Mode Fiber

Single-mode fiber acts as a waveguide that limits the propagation of light to one mode due to its small core diameter. Because of its much larger bandwidth than multi-mode fiber, it is becoming the standard for optical fiber communication systems. This increase in bandwidth is caused by a decrease in the dispersion of the single-mode fiber, as compared to multi-mode fiber, because of the elimination of modal dispersion. Modal dispersion occurs because the modes of light contained in the fiber propagate at different velocities, causing pulse broadening. Also, the production of single-mode fiber requires less dopant and has looser tolerances than multi-mode fiber, resulting in lower manufacturing cost. Major disadvantages of single-mode fiber are its small numerical aperture (NA) and core diameter. The small NA of the fiber decreases the amount of power that can be coupled into it from a non-coherent source (such as a LED), while the small core diameter increases the difficulty of aligning the fibers to sources or for splicing.

1.2.1 Characteristics of Single-Mode Fiber

Single-mode fiber has two main characteristics that limit the maximum bit-rate and transmission distance. The maximum bit-rate is limited by intersymbol interference (ISI) caused by pulse broadening due to the dispersion of the fiber. However, pulse broadening is a function of both dispersion and fiber length. Therefore, dispersion forces a trade-off between transmission distance and bit-rate. The maximum transmission length is also limited by the attenuation of the fiber. Thus, depending on the characteristics of the system, it may be either attenuation or dispersion limited.

1.2.1.1 Attenuation

The attenuation of the fiber can be accounted for by two different mechanisms, Rayleigh scattering and absorption. Rayleigh scattering is caused by the interaction of light with the small density variations that occur in the glass fiber. Scattering losses are greatest when the wavelength of the light approaches the width of the the density imperfections. As the wavelength of the light increases, the scattering losses decrease as a function of the fourth power of the wavelength.

Absorption of the light also occurs due to the physical properties of the fiber material — silica, and the impurities that it contains. The most serious impurity is the hydroxyl ion (OH). It causes absorption peaks, in the window of interest, at wavelengths of 1.23 µm and 1.37 µm. With strict quality control and specialized manufacturing processes, the magnitude of these peaks can be reduced to below 1.5 dB/km[20]. The material of the fiber, silica, absorbs light in the infrared region. These losses become noticeable at 1.6 µm and increase with wavelength. The losses due to Rayleigh scattering and absorption are combined to give the attenuation profile of the fiber. This is illustrated in Figure 1.1 over

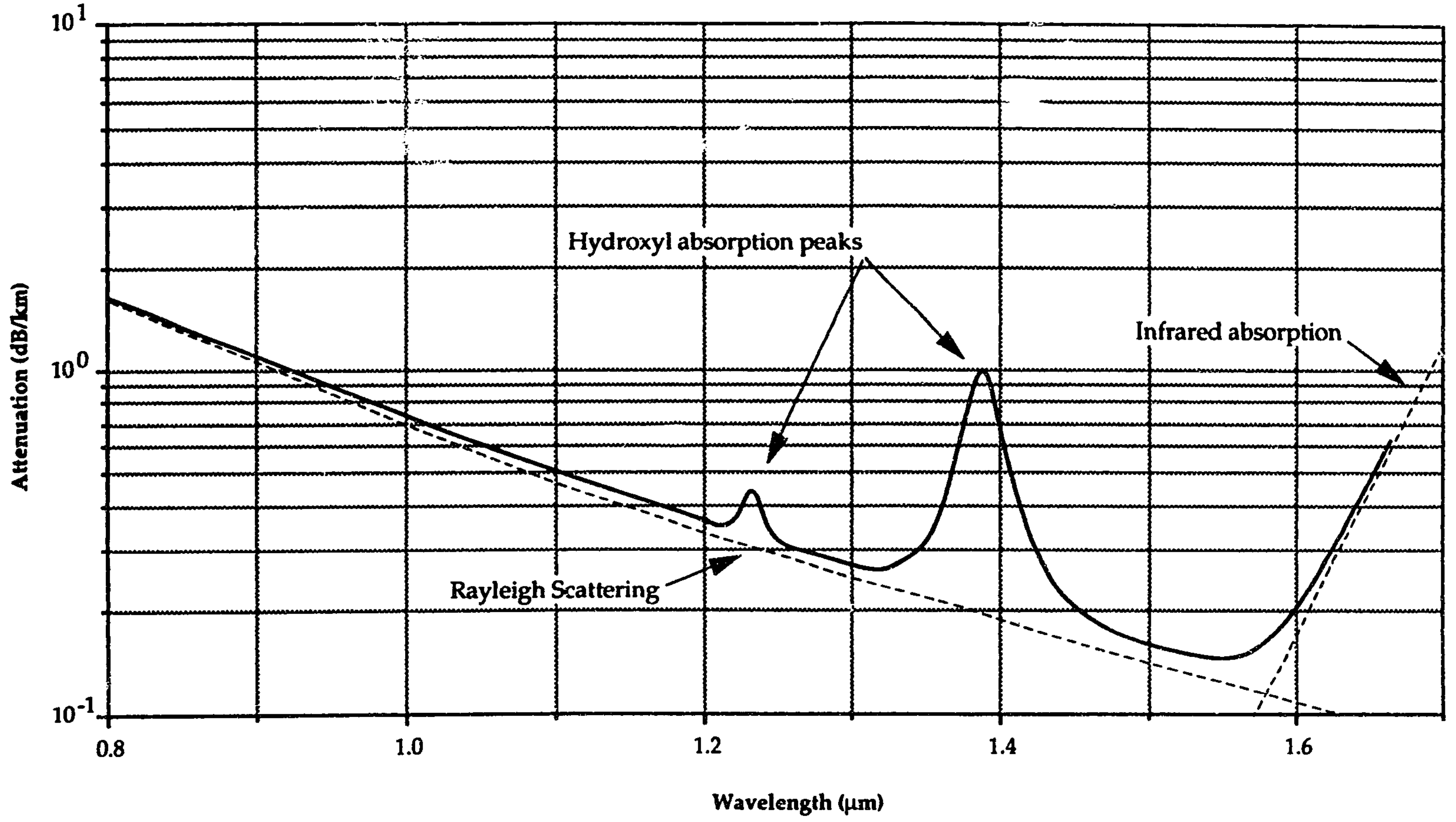


Figure 1.1 Attenuation characteristics of single-mode fibers.[21]

the range of wavelengths that are pertinent to single-mode fiber systems. It can be seen that the attenuation curve contains two minima: a local minimum at 1.31 μm with an attenuation of approximately 0.3 dB/km, and the absolute minimum at 1.55 μm with an attenuation of less than 0.2 dB/km. Both of these regions of minimum attenuation give wavelength "windows" that are important in the development of SMF optical communication systems.

The attenuation of a fiber (α) is a function of wavelength (λ) and is given by

$$\alpha(\lambda) = \frac{10}{L}\log_{10}\left(\frac{P_0(\lambda)}{P_L(\lambda)}\right) \qquad \mathrm{dB}/\mathrm{km} \qquad 1.1$$

where $P_0(\lambda)$ and $P_L(\lambda)$ represent the spectral power distribution of the light launched into the fiber, and that measured at the output of the fiber of length L, respectively[22]. Because of the wider spectral width of the light emitted from LEDs, using the attenuation measured at the central wavelength will not necessarily give a good approximation to the overall attenuation. However, once the value for the average attenuation (α_{av}) has been calculated, one can use

$$\alpha(L) = \alpha_{av}L \qquad \mathrm{dB} \qquad 1.2$$

to obtain a good approximations of the total attenuation ($\alpha(L)$) for a range of different fiber lengths.

1.2.1.2 Chromatic Dispersion

Chromatic dispersion is the sum of waveguide and material dispersion. Waveguide dispersion is dependent on the characteristics of the fiber channel while material dispersion depends on the properties of silica, the material used in the manufactu [illegible] ingle-mode fiber. In both cases, the dispersion is the result of a change in [illegible] ve refractive index of the fiber as a function of wavelength. The refractive index of a material is the ratio of the velocity of light in that material to its velocity in a vacuum. Thus, if a pulse of light containing multiple wavelengths is coupled into a fiber, the different wavelengths of light will propagate down the fiber at different velocities which causes pulse broadening. This pulse broadening limits the bit rate of the fiber and hence its effective bandwidth. The lower the bit rate, the wider the transmitted pulses, and thus the larger the amount of pulse broadening that can be tolerated before ISI occurs.

The effective bandwidth or bit-rate of the fiber is a function of the distance. The longer the transmission span, the greater the difference in arrival times, and hence, the greater the pulse broadening. Thus, the dispersion limited performance of an optical fiber system is usually defined in terms of the bandwidth-distance product.

The fiber dispersion $D(\lambda)$ is expressed as

$$D(\lambda) = \frac{1}{L} \times \frac{d\tau_g}{d\lambda} \qquad 1.3$$

where "τ_g" is the group delay of the wavelengths of light emitted by the source. The following empirical formula for dispersion

$$D(\lambda) = \frac{A_o\lambda}{4}\left(1 - \left(\frac{\lambda_o}{\lambda}\right)^4\right) \qquad \begin{array}{l} A_o = 0.1 \text{ ps/km.nm}^2 \\ \lambda_o = 1.31\ \mu\text{m} \end{array} \qquad 1.4$$

is widely used[23].

Ideally, for a source that emits light at the zero dispersion wavelength λ_o, no pulse spreading will occur and the system will be attenuation limited. Unfortunately, all sources emit a range of wavelengths with typical spectral widths of 1 to 5 nm for LDs and 30 to 120 nm for LEDs. If a source with a wide spectral width is centered over the zero dispersion wavelength of the fiber, the effects of pulse spreading still occur, although to a smaller extent. Thus, the larger the spectral width of the source, the lower the expected performance in a dispersion limited system. This is confirmed in Figure 1.2 where the bandwidth-distance product is plotted versus the center wavelength of the source for a variety of spectral widths. It also shows that the ability of an optical fiber system to transmit information at high bit-rates over long distances increases substantially for sources located at the zero dispersion point of the fiber. For an accurate assessment of the dispersion penalties of a system the spectral power distribution of the source, the transfer function of the fiber, and the characteristics of the receiver must be taken into account[24].

1.2.2 Types of Single-Mode Fiber

There are three major types of single-mode fiber[25]. These fibers are classified by their dispersion characteristic, which is a combination of material and waveguide dispersion. The dispersion characteristics of single-mode fiber can be changed by modifying the waveguide dispersion. This is accomplished by altering the doping profile and/or the physical shape of the core of the fiber. The three different types of single-mode fiber are:

1) Dispersion Non-Shifted: The non-shifted dispersion borosilicate fiber has a zero dispersion wavelength of approximately 1.31 μm. Of the three types of fiber, it is the most common and the easiest to manufacture, due to the constant refractive index profile of the core.
2) Dispersion-Shifted: The dispersion-shifted fiber has its zero dispersion point shifted to a wavelength of 1.55 μm. This is accomplished by decreasing the waveguide dispersion, which shifts the zero dispersion point of the fiber, to higher

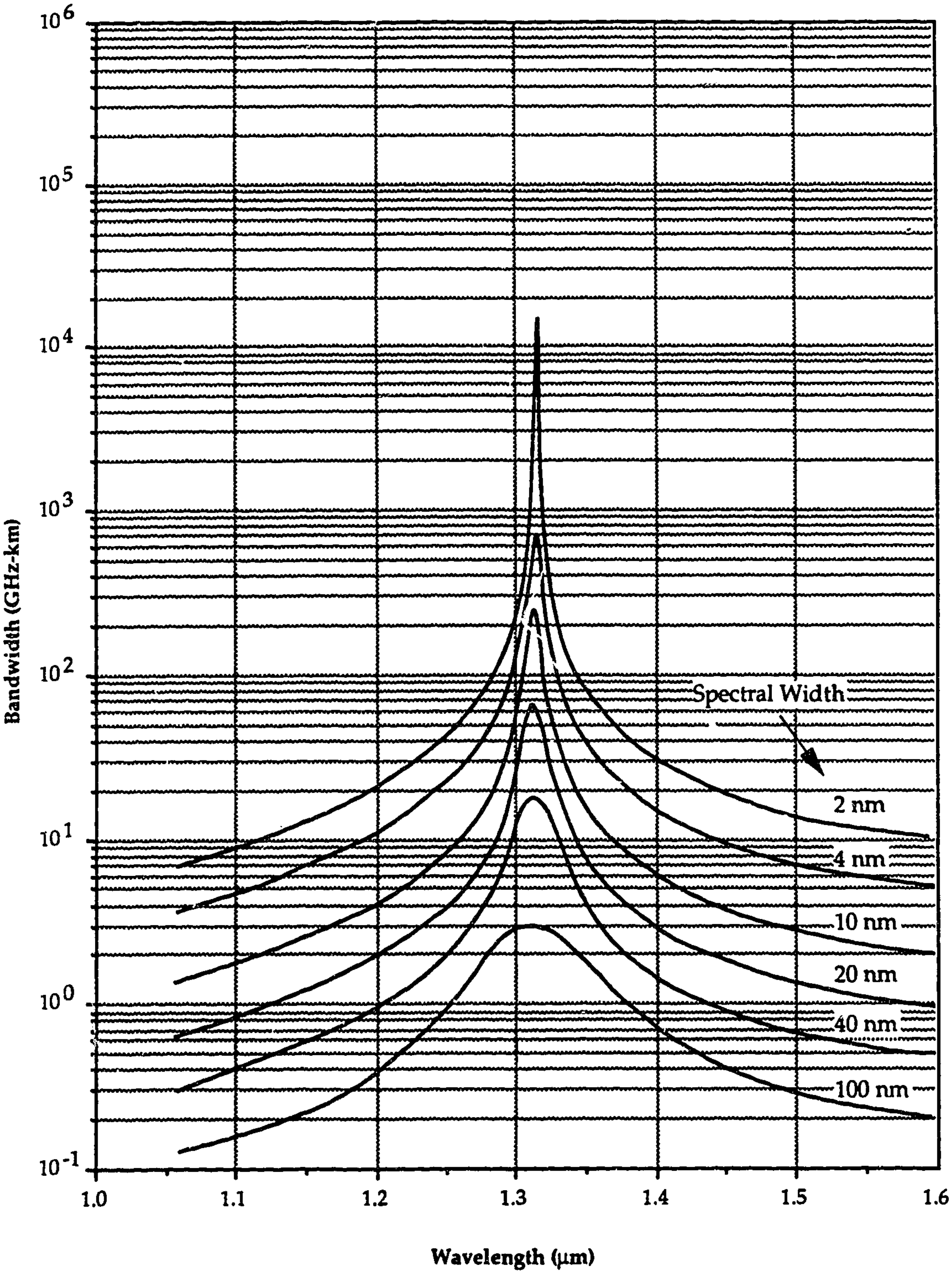


Figure 1.2 Bandwidth-distance characteristics of single-mode fibers.[20]

wavelengths. This allows designers to take advantage of the lower theoretical attenuation of the fiber, thus increasing the transmission distance for attenuation-limited optical fiber communication systems.

3) Dispersion-Flattened: The dispersion-flattened fiber has a dispersion characteristic with a maximum dispersion of less than 3.5 ps/nm.km for the range of wavelengths between 1.3 and 1.6 μm and is characterized by two zero dispersion points. In this case the slope of the waveguide dispersion, as a function of wavelength is decreased, which causes the chromatic dispersion curve to reverse itself at higher wavelengths. Because of the low dispersion over this range, this type of fiber can be used with sources having wide spectral widths and/or for systems using wavelength multiplexing. As an example, if non-shifted fiber was used for a wavelength multiplexing system at 1.3 μm and 1.5 μm wavelengths, the ratio of the maximum bit-rates would be approximately 3:1 because of the higher dispersion encountered at 1.55 μm[26]. For dispersion-flattened fiber, a maximum bit-rate ratio of approximately 1:1 could be achieved in a dispersion-limited system[14].

To better compare the dispersion characteristics of the different types of fiber, they are plotted in Figure 1.3 over the range of wavelengths of interest[27]. It should be mentioned that the complexity of the refractive index profile of the core for both the dispersion-shifted and dispersion-flattened fiber will increase the difficulty and cost of manufacturing.

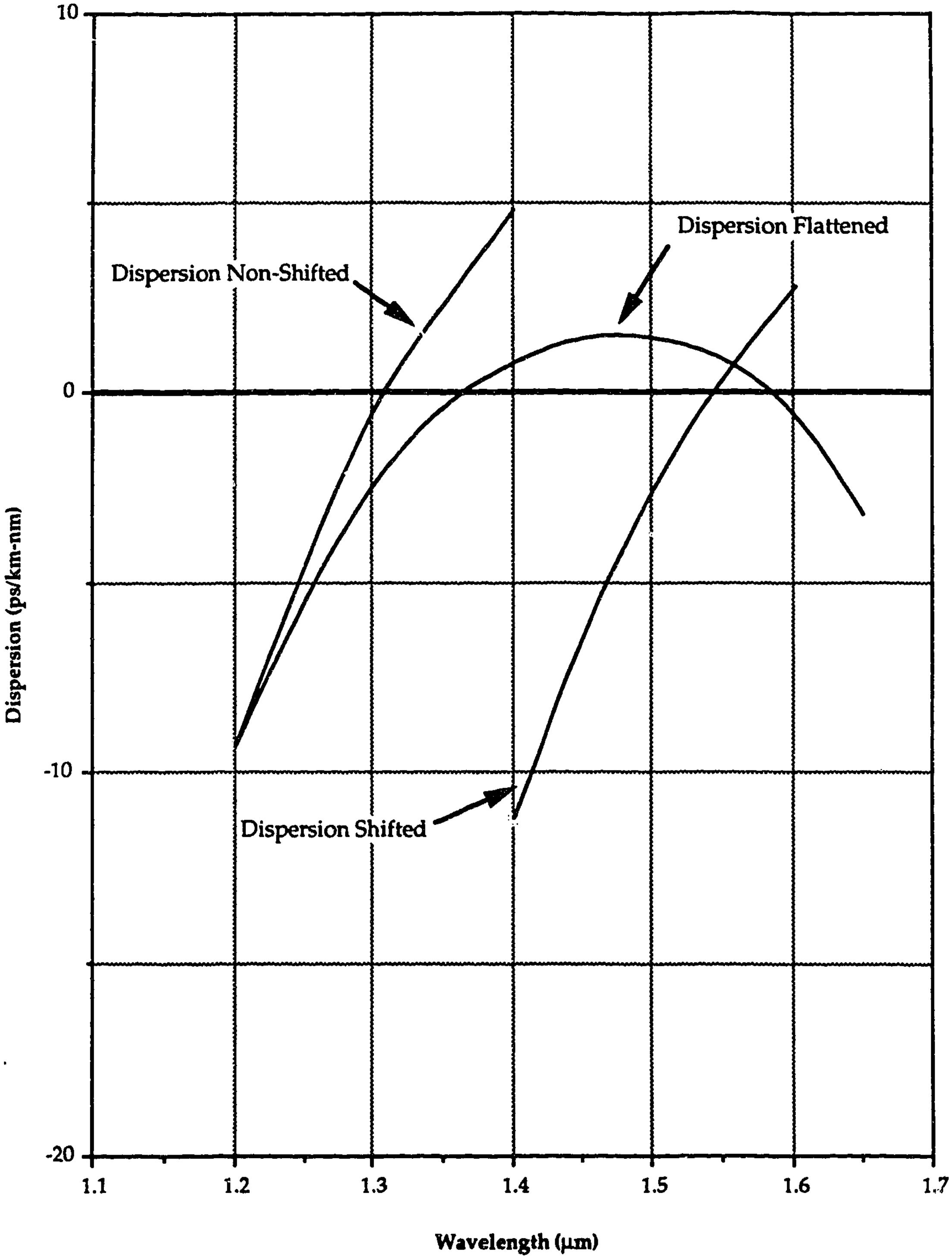


Figure 1.3 Dispersion characteristics of single-mode fibers.[28]

1.3 Light-Emitting Diodes

Sources for optical fiber communication systems must meet certain requirements. They must have high coupled power, high reliability, high modulation speed, a narrow spectral width and the ability to operate over a range of environmental conditions. This narrows the field down to two basic types of sources — LDs and LEDs. Even with the advent of single-mode fiber, LEDs were initially considered impractical for use as optical sources because of their large spectral width and low launch power. But for short distance applications where initially only medium to low bit-rates are required, the advantages that LEDs have over LDs make them the logical choice. LEDs are generally more reliable, are less expensive, require simpler drive circuitry, have a lower temperature dependence and lack the interference effects and instabilities of LDs[28].

1.3.1 Characteristics of LEDs

The characteristics of the LED that define its performance are described below. They include coupled power, spectral width, frequency response and temperature dependence.

1.3.1.1 Coupled Power

The power of the light from the LED that illuminates the end of the single-mode fiber is dependent on two main factors — the intensity and the half-power beam width (HPBW) of the source. The intensity of the emitted light is effected by the the quantum efficiency, the geometry of the device, and is almost directly proportional to the amount of drive current across the P-N junction. The HPBW of the LED is mainly determined by the device geometry. The percentage of the power from the LED coupled into the fiber is dependent on both the HPBW of the LED and the numerical aperture of the fiber. The numerical aperture is defined as

$$NA = \sin a_o = \sqrt{n_1^2 - n_2^2} \qquad 1.5$$

where "a_0" is the acceptance angle of the fiber, which is a function of the refractive index of the core "n_1" and the cladding "n_2". For single-mode fiber, typical values for the NA and acceptance angle are 0.2 and 12°, respectively. By either increasing the NA of the fiber or by decreasing the HPBW of the source the percentage of power coupled into the fiber can be increased. If the active area of the source is smaller than that of the core of the fiber, then optics can be used to concentrate the light onto the core to increase the power coupled into the fiber.

1.3.1.2 Spectral Width

The light emitted from a LED contains a range of wavelengths. This spectral power distribution of the emitted beam is defined by its central wavelength and spectral width. The spectral width is defined either by its full width at half maximum (FWHM) or its root-mean-square (RMS) value. This value is dependent on the device structure and decreases with increased active layer thickness due to the band filling effect[29].

1.3.1.3 Frequency Response

The maximum modulation frequency or speed of the LED is dependent on the recombination rate of the electrons and holes in the junction of the device. The average time that it takes for the charges injected into the junction to recombine is the carrier lifetime τ_c

$$\tau_c \propto \frac{1}{R_C \cdot N_A} \qquad 1.6$$

where "R_C" is equal to the radiative recombination rate and "N_A" is the number of acceptors[30]. In practice, the frequency response also depends upon the capacitance of the device. A device with a small active volume will have a smaller capacitance and thus a higher frequency response.

1.3.1.4 Temperature Dependence

Temperature affects both the output power and the spectral power distribution of the LED. The output power P_0 decreases with rising temperature and can be modeled by the following empirical formula

$$P_0 = ke^{-T/T_0} \qquad 1.7$$

where k and T_0 are constants that depend on the structure and operating current of the LED[31]. The spectral power distribution of the LED shifts to higher wavelengths as the temperature increases due to narrowing of the energy bandgap. The magnitude of the shift is in the order of 0.4 nm/°C. The temperature dependence of the LED is not as large as that found in LDs and in many practical systems, no active method of temperature stabilization is required. Usually, the more complicated the structure of the source, the higher its temperature dependence.

1.3.2 Types of LED

Basically, there are three different types of LEDs — surface-emitting, edge-emitting and super-radiant. They can be differentiated from each other by their physical structure and the internal mechanisms used to generate light.

1.3.2.1 Surface-emitting LED

Surface-emitting LEDs (S-LED) consist of a planar structure in which light is emitted perpendicular to the grown surface. The Burrus LED is the most common type of S-LED. Its structure is modified by etching a well into the surface of the LED, which is used to concentrate the light from the active layer located below in order to increase the coupled power[32]. Because of the simplified structure of the S-LEDs, they were the first LED sources to be used in optical fiber systems. They are relatively inexpensive and have the smallest temperature dependence and the highest reliability of any of the sources. Their disadvantages are that they have the widest spectral width and lowest coupled power of the three types of LEDs.

Typical values for the FWHM spectral width and coupled power into single-mode fiber for the S-LED are in the order of 120 nm and 0.25 μW, respectively[33]. High-speed surface emitting LEDs were initially developed for use in short loop multi-mode fiber systems, but they are also attracting limited interest for some single-mode fiber systems.

1.3.2.2 Edge-emitting LED

The structure of the edge-emitting LEDs (E-LED) is similar to that of the S-LEDs except that the light is emitted parallel to the junction and is confined to a buried channel in the device. This structure increases the intensity and reduces the HPBW of the emitted light beam. Both of these effects will increase the amount of power coupled into the fiber. It is also possible to increase the coupled power by using optics to concentrate the light into the core of the fiber for some E-LEDs which have a channel width smaller than the diameter of the core. A comparison between butt-coupling and optical-coupling for an E-LED is shown in Figure 1.4. In this case there was approximately a four fold increase in the coupled power when a tapered lens was used. A typical value for the optical power that a commercial E-LED will couple into a single-mode fiber is in the order of 7 μW.

The spectral width of the light emitted from the E-LED is narrower than that emitted from the S-LEDs. Typical values of the FWHM spectral widths are

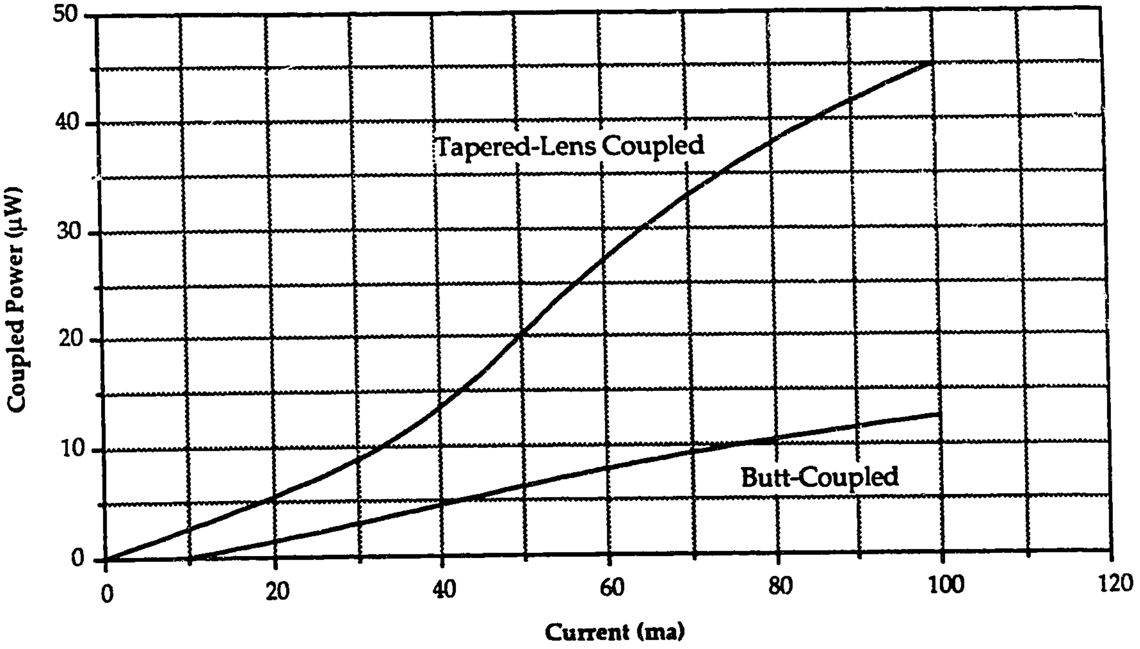


Figure 1.4 Power coupled into single-mode fiber from an E-LED.[5]

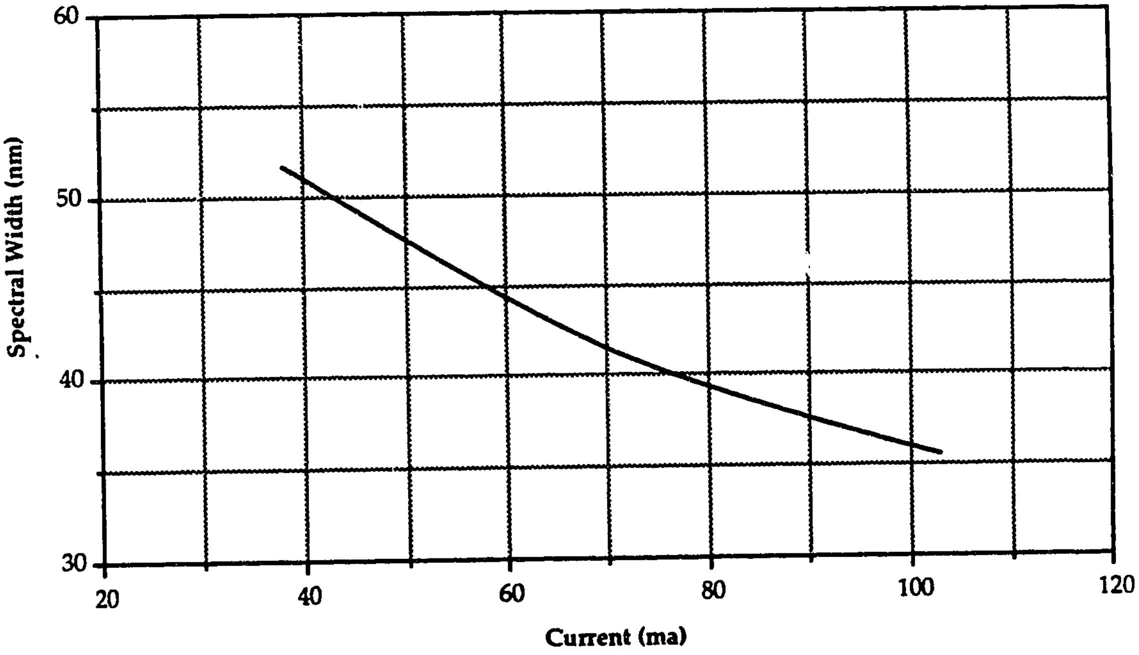


Figure 1.5 Spectral width versus drive current of an E-LED.[5]

in the order of 70 nm. It is also important to note that the spectral width of the E-LED narrows with increased drive current as shown in Figure 1.5. This indicates that operating the E-LEDs at higher drive currents or optical power levels may be desirable. It should also be noted that the spectral width of the light coupled into the single-mode fiber is about 25-35 nm smaller than the spectral width of the light emitted from the E-LED. This is because the light from the center of the active area of the E-LED is more homogeneous and constitutes a larger part of the light coupled into the single-mode fiber due to the small acceptance angle[29].

E-LEDs have a slightly higher temperature dependence and a lower reliability than S-LEDs. Their projected life spans, however, are greater than 10^8 hours at room temperature, which is three orders of magnitude larger than the life spans predicted for LDs[34,35]. Because of the E-LEDs' high coupled power, smaller spectral width, and longer lifetimes, they are the most prevalent type of LED to be used as a source for single-mode fiber.

1.3.2.3 Super-radiant LED

The super-radiant LEDs (SR-LED) are a subset of the E-LEDs. They have a similar structure to the E-LEDs except that the channel has a larger active length. Increasing the active length of the channel can be accomplished by either creating a facet on the backface of an E-LED or alternatively replacing the front facet of a LD with a non-reflective surface. The advantages that the SR-LEDs have over the other types of LEDs are an increase in coupled power due to the higher intensity and the smaller HPBW of the emitted beam, and a narrower spectral width. This comes at the cost of even higher temperature dependence, which necessitates temperature stabilization circuitry and lower reliability.

As we can see, the SR-LEDs fall in between the LDs and the E-LEDs not only in structural design but also in performance. Because of the major deviation of the SR-LEDs from the characteristics that made the LEDs desirable as sources, they are not used extensively in single-mode fiber systems.

1.4 Coding

Coding is used to optimize the performance of communication systems. This optimization may be required for one or more of the following reasons[36]:

1) to increase the information capacity,

2) to reduce transmission limitations,

3) to implement error detection and correction.

One of the purposes of coding is to increase the amount of information that can be transmitted over a channel. The maximum information rate for a bandlimited channel with additive noise was derived theoretically by C.E. Shannon in 1948[37]. He defined the maximum information rate as the channel capacity (C), which is a function of the channel bandwidth (B) and the signal to noise ratio (SNR). The channel capacity is given by

$$C = B \cdot \mathrm{Log}_2(SNR+1) \qquad 1.8$$

Though this expression gives the maximum information rate that can be achieved for the channel, it does not in any way define the code required to achieve this rate.

Another major use of coding is to modify the information being transmitted to increase the performance of the system which may be limited by physical characteristics of the system or the information. These limitations may include AC coupling of the signal, bandwidth limitations of the channel or the devices, linearity and sensitivity of the devices, and jitter sensitivity[38,39].

Coding is also used to implement error detection and error correction capabilities into the transmitted information. This is accomplished by increasing the redundancy of the information transmitted, making it possible to detect errors and for some codes, correct them. An example of the use of error detection in a code is parity. By sending an extra bit that indicates the parity of the code word transmitted, it is possible for the decoder to detect if a single error has occurred. Error detection and error correction increases the complexity of the necessary electronic circuitry, which tends to limit the high speed implementation of the code. Also, the redundancy of the code increases the required bandwidth of the system, due to an increase of the transmitted bit-rate.

1.4.1 Code Requirements

To optimize the performance of LEDs, coding is to be implemented to increase their viability as a source for single-mode fiber. The two primary factors that limit the performance of LEDs are:

1) Maximum brightness: This limits the amount of optical power that can be coupled into a single-mode fiber, thus restricting the distance that the signal can propagate down the fiber due to attenuation.
2) Large spectral width: This increases the amount of pulse broadening that occurs due to the chromatic dispersion of the fiber. This limits the the bit-rate and/or transmission distance due to ISI.

Besides the limitations due to the characteristics of the LED, there are other secondary factors which limit the performance of a system but which coding can alleviate. Some of the main secondary factors are:

1) Baseline wander: The baseline of the received data stream shifts due to the AC coupling of the signal in the receiver. This decreases the performance of the system by varying the

effective spacing between the decision threshold level of the electronics and the amplitude of the signal. The AC coupling acts as a highpass filter which removes the DC and attenuates the low frequency components that are contained in the data stream. So optimally, coding should eliminate or attenuate these DC and low frequency components in order to minimize the baseline wander.

2) Clock Content: To regenerate the clock from the received data stream, there should be sufficient clock content contained in the signal. As the transition rate increases, so does the clock content. A larger clock content simplifies the recovery of the clock and tends to reduce the magnitude of the timing jitter.

3) Simplified coding rules: As the data rate increases, the time available to encode the data sequences decreases. In order to implement a code at a high data rate, simplified coding rules are required to minimize the complexity of the encoder and decoder electronics and the delay times associated with them, thus allowing the data to be encoded in the time available.

4) Bit sequence independence: This means that the code's performance in the communication system is independent of the data sequence encoded. An example is the NRZ binary code that is AC coupled at the receiver. Due to the AC coupling, the magnitude of the baseline wander of the received data sequences varies and depends on the number of sequential ones or zeros transmitted. This ties the performance of the system to the bit sequences encoded, indicating that the code is not bit sequence independent.

5) Error multiplication: This is the increase in the error rate due to incorrect decoding of an erroneously received encoded data sequence. In coding, there are two main sources of error multiplication. One is due to the multi-level characteristic of coding, where several bits of data are encoded into one symbol. An error in detecting the symbol will be multiplied in the decoded data by a factor that is dependent on the code and on the number of bits encoded. The other source of error multiplication is the propagation of an error to the other encoded bit sequences. This occurs if information contained in the erroneously decoded bit sequence is required for the decoding of the other encoded sequences.

In selecting a code, it may not be possible for it to be optimized with respect to all of the above factors. But because of the objectives, priority will be given to those factors that limit the use of the LEDs in a SMF system.

1.4.2 Code Selection

A code that will improve the performance of LEDs in a SMF system should counteract the effects of the limited amount of power that can be coupled into the fiber and the pulse broadening due to dispersion. Depending on the code used and the characteristics of the SMF system, the system's performance will either be attenuation or dispersion limited.

For a SMF system that is attenuation limited, the main requirement is to increase the effective power coupled into the fiber. This can be accomplished by overdriving the LED to increase its peak output power. This technique can only be applied when the pulse width is much smaller than the thermal time constant of the LED and the average power constraints of the device are not exceeded.

To increase the peak output power while keeping the average power constant requires a coding scheme which has a low duty cycle. If there is a linear relationship between the power applied and the power emitted from the LED, then the maximum peak power (P_P) that can be obtained will be equal to the inverse of the duty cycle of the code which is the ratio of the LED's on time to the total time of operation. This relationship between the maximum peak power and the duty cycle of a code is shown in Figure 1.6.

It should also be mentioned that decreasing the duty cycle will correspond to a decrease in pulse width by a factor that is dependent on the code. If the pulse width is decreased, then a corresponding increase in the bandwidth of the system is required to transmit the data. This increase in bandwidth decreases the receiver sensitivity due to an increase in the noise power at the output of the receiver. Thus, the increase in noise power imposes a bandwidth related power penalty on the system which will decrease the peak power related power gain of the code.

In order to compare the increase in the effective power for the different codes, which is a function of their peak power and required bandwidth, the effective power gain of each code will be calculated with respect to the effective power of a NRZ pseudo-random binary sequence (PRBS). The NRZ PRBS is to be used as a reference level because it is well defined in the literature and is used as a standard in the testing of communication systems. The effective power gain (P_{EG}) of a low duty cycle code is then given by

$$P_{EG} = \frac{P_{PG}}{P_{NP}} = \frac{P_{P\,CODE}/P_{P\,NRZ}}{P_{NP}} \qquad 1.9$$

where "P_{PG}" is the peak power gain of the code, which is equal to the ratio of the maximum peak power of the code (P_{PCODE}) to that of the maximum peak

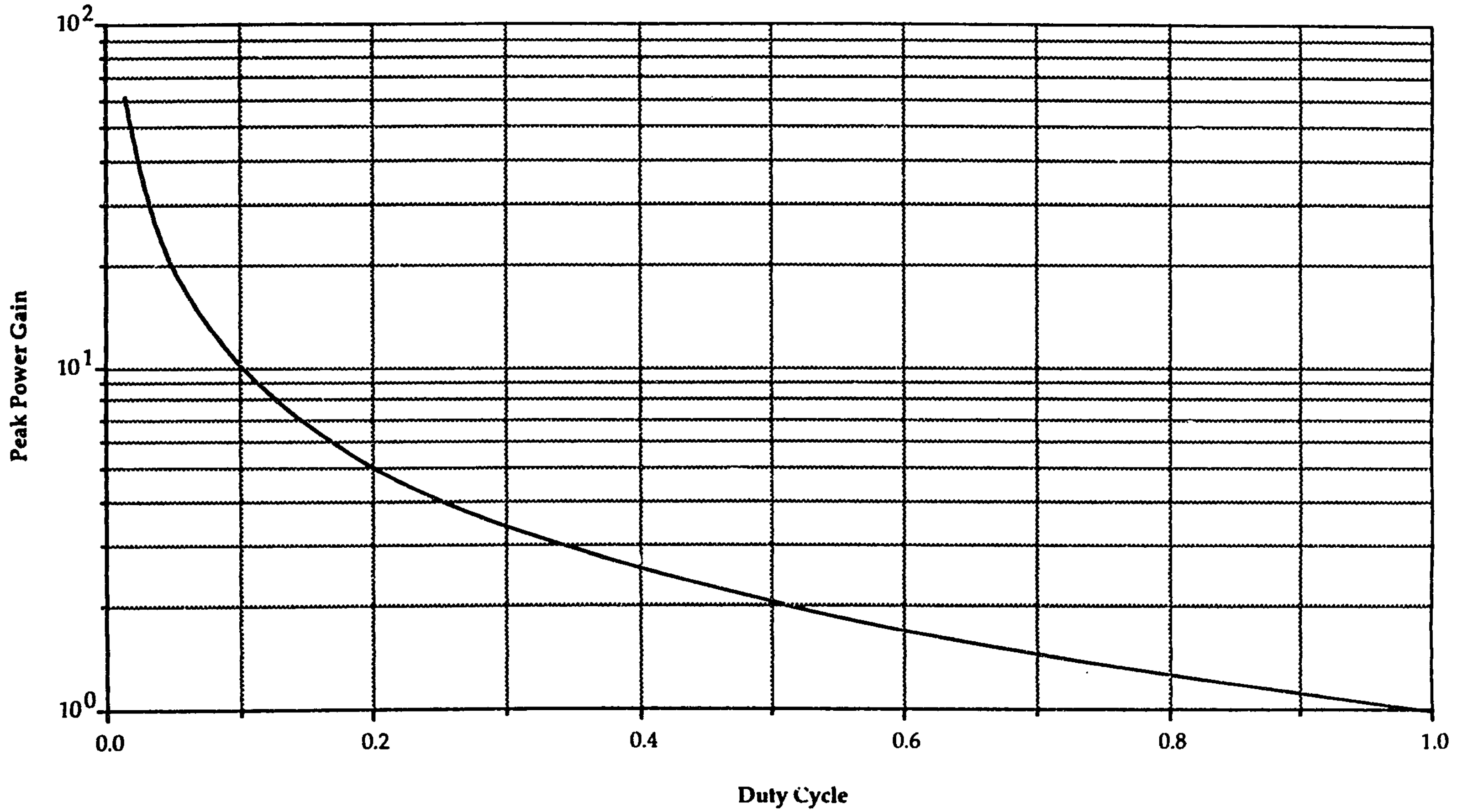


Figure 1.6 Peak power versus duty cycle.

power of the NRZ PRBS (P_{PNRZ}), and "P_{NP}" is the noise power penalty due to the increase bandwidth required for the code. This expression for the effective power gain of a low duty cycle code can be simplified to

$$P_{EG} = \frac{\dfrac{1}{D_{CODE}}}{\dfrac{1}{D_{NRZ}}} \cdot \frac{1}{P_{NP}} = \frac{1}{2 \cdot D_{CODE} \cdot P_{NP}} \qquad 1.10$$

where the maximum peak powers are represented by the inverse of their duty cycles, which in the case of the NRZ PRBS is 0.5.

If the SMF system is dispersion limited, then the main requirement of the code is to decrease the ISI due to pulse broadening. There are two alternate approaches that coding can take to reduce this effect. One, is to increase the width of the transmitted pulses which will reduce the required bandwidth and the effects of pulse broadening. This will either decrease the bit-rate or increase the duty cycle of the code and since neither result is desirable this approach was rejected.

The other is to follow each the transmitted pulse with a space or guardtime. This would increase the amount of pulse broadening that could be tolerated before ISI occurred. The disadvantage of this approach is that, since the guardtime does not convey any information, it reduces the efficiency of the code, causing an increase in the required bandwidth needed to transmit the data. This increase in the bandwidth is caused by a reduction in the pulse width because the time interval in which the encoded data is transmitted is decreased by the size of the guardtime. This approach to reduce the effects of dispersion, complements the technique used to increase the peak power. Since, the narrowing of the pulse width due to the guardtime results in a further decrease in the duty cycle of the code which corresponds to an increase in the peak power gain. It should be mentioned that in Section 1.3 it was shown that the spectral width of the E-LED decreases with increased output power, which indicates that high peak powers may be desirable.

From the above discussion, it can be seen that the two main requirements of a code for a LED SMF system is a low duty cycle and the ability to follow the transmitted pulse with a guardtime. Also, for the code to be implemented into a practical system, it must be able to operate at high data rates, thus indicating the use of simplified coding rules.

1.4.2.1 Pulse Position Modulation

Pulse position modulation (PPM) was initially examined because of its low duty cycle and its ability to implement a guardtime. Its other advantages include minimal DC baseline wander and the ability to trade bandwidth for SNR for improved performance[40]. Theoretically, PPM is shown to have a potential 10 to 12 dB improvement over equivalent PCM systems[41,42], while experimental gains in performance of 4.2 dB have been demonstrated[43].

Digital PPM is represented by "nB2^nPmG" where a binary data sequence of length "n" is encoded by the positioning of a pulse within a fixed time frame. This time frame is divided into "2^n+m" equal width slots, where each of the "2^n" slots represents a different binary sequence and the "m" slots are used for either a guardtime, a framing pulse or both. An example of 3B8P1G PPM is shown in Figure 1.7b. Here three binary bits are encoded into eight position PPM with a framing pulse. The framing pulse is an extra pulse used to mark the beginning of each frame to synchronize the data stream. This synchronizing pulse increases the duty cycle of the code by a factor of two, which decreases the amount of the peak signal power that could be coupled into the fiber. Eliminating the framing pulse would increase the difficulty of regenerating the clock and resynchronizing the decoder after an error.

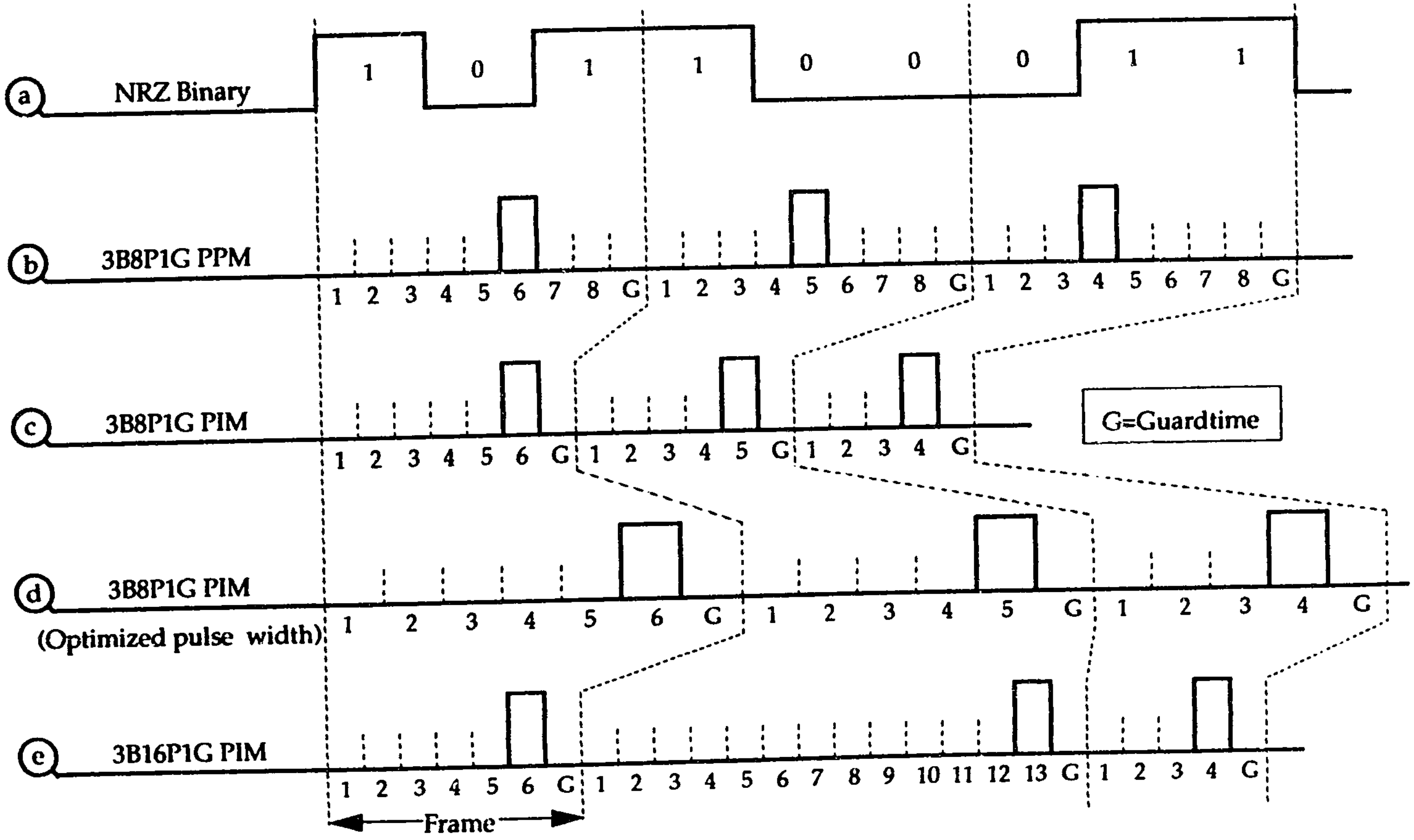

a NRZ Binary
1 0 1 1 0 0 0 1 1
b 3B8P1G PPM
1 2 3 4 5 6 7 8 G 1 2 3 4 5 6 7 8 G 1 2 3 4 5 6 7 8 G
c 3B8P1G PIM
1 2 3 4 5 6 G 1 2 3 4 5 G 1 2 3 4 G
G=Guardtime
d 3B8P1G PIM
(Optimized pulse width)
1 2 3 4 5 6 G 1 2 3 4 5 G 1 2 3 4 G
e 3B16P1G PIM
1 2 3 4 5 6 G 1 2 3 4 5 6 7 8 9 10 11 12 13 G 1 2 3 4 G
Frame

Figure 1.7 Examples of the NRZ, PPM and PIM code waveforms.

In the case where no framing pulse is used, the effective power gain of the PPM code, as defined by Equation 1.10, can be expressed in terms of the length of the encoded data sequence "n" and the number of slots "m" used for the guardtime. Since, only one pulse is transmitted in each frame, the duty cycle for PPM (D_{PPM}) code is equal to the inverse of the number of slots that the frame contains. This is shown by

$$D_{PPM} = \frac{1}{2^n + m} \qquad 1.11$$

To derive the noise power penalty P_{NP} for the PPM code, certain assumptions have to be made. First, the noise is white with a constant one-sided power spectral density of "N_O", thus the noise power "N" is a linear function of bandwidth. Second, the power spectral density curves for the two codes are identical or similar in shape. This permits a direct comparison between the bandwidth (B) required for a code and its transmitted pulse width (T_P). From these suppositions, the noise power penalty (P_{NP}) can be derived as a function of the transmitted pulse widths of the NRZ and PPM codes, as shown by

$$P_{NP} = \frac{N_{PPM}}{N_{NRZ}} = \frac{N_O \cdot B_{PPM}}{N_O \cdot B_{NRZ}} = \frac{T_{P_{NRZ}}}{T_{P_{PPM}}} \qquad 1.12$$

Since the width of the PPM frame (T_{FR}) is equal to the width of the NRZ binary sequence encoded, the ratio of pulse widths between the NRZ and PPM code can be expressed as

$$P_{NP} = \frac{T_{P_{NRZ}}}{T_{P_{PPM}}} = \frac{T_{FR}/n}{T_{FR}/(2^n + m)} = \frac{2^n + m}{n} \qquad 1.13$$

which gives the noise power penalty in terms of the number of binary bits encoded "n" and the number of slots required for the guardtime "m". By substituting the expressions for the duty cycle and the noise power penalty for the PPM code into Equation 1.10, the effective power gain for the PPM code is found to be

$$P_{EG} = \frac{1}{2 \cdot D_{PPM} \cdot P_{NP}} = \frac{2^n + m}{2} \times \frac{n}{2^n + m} = \frac{n}{2} \qquad 1.14$$

which is equal to half the number of binary bits encoded per PPM symbol. Because the effective power gain is independent of the guardtime, which is used to decrease the effects of pulse broadening, it is possible to optimize the code for both the attenuation and dispersion characteristics of the fiber.

1.4.2.2 Pulse Interval Modulation

Due to the disadvantages associated with framing pulses, a modified form of PPM — pulse interval modulation (PIM), which maintains the low duty cycle of ideal PPM — is considered as a superior alternative. This is because the data in the PIM code are represented by the interval between two pulses. Thus, a separate framing pulse is not required, since the data pulse of the preceding PIM frame acts as the framing pulse for the current frame. As a result, PIM can operate with one pulse per frame without any elaborate scheme for synchronization. If an error occurs (e.g. due to noise), the decoder will be able to resynchronize on the next pulse.

The simplest form of PIM encoding is represented by "$nB2^nPmG$". This is similar to the PPM code except that the "2^nP" represents the number of intervals instead of the number of positions that are required. An example of the 3B8P1G PIM code is shown in Figure 1.7c using the same pulse width as for the 3B8P1G PPM code. It consists of encoding a sequence of three NRZ binary bits to an interval with eight possible spacings, and where the pulse that represents the end of the current interval is followed by a guardtime.

In examining the PIM waveform, shown in Figure 1.7c, we can see two possible problems arising from the code, one due to error propagation and the other due to the variable frame width. If an error occurs either due to false triggering from noise exceeding the threshold level or the attenuation of the received pulse to such a degree that it falls below the threshold level, then this error will produce two erroneous PIM intervals. This is because each PIM pulse is used to define an end or start point for two adjacent time frames, thus giving the PIM code an error propagation rate of two.

To examine the effect of the variable frame width of the PIM code, we first define the frame as the interval from the end of the guardtime of the previous pulse to the end of the guardtime of the present pulse. Unlike PPM, where the frame width is constant, the PIM frame varies in width according to the data encoded. This means that the encoded PIM data stream is asynchronous with respect to the original NRZ binary data stream. To permit the continuous transmission of data, buffers will be required in both the encoder and decoder to interface between the the binary data stream and the variable frame width of the PIM signal. The size of these buffers will depend on the statistical nature of the data, the acceptable error rate due to buffer overflow, and the PIM coding scheme.

To limit the size of the buffers, two conditions must be fulfilled. First, the average length of the PIM interval must be equal to the width of the encoded NRZ binary sequence. Second, the deviation between the end of the current PIM frame and the end of its encoding NRZ binary data sequence must be finite. This deviation is expressed as the running digital slip (RDS), which is defined as the number of PIM slots that the current PIM frame is ahead of the NRZ encoded data stream.

In comparing the PIM to the PPM code, as shown in Figure 1.7, we see that the magnitude of the RDS for the 3B8P1G PIM increases with each encoded NRZ binary sequence. This is because the average interval of the PIM code, which is dependent on the statistical characteristics of the data, is smaller than the encoded NRZ binary sequence. Since the average PIM interval should be equal in length to the encoded NRZ binary sequence, the width of the PIM pulse is obviously too narrow. The correct PIM pulse width ($T_{P_{PIM}}$) can be found by dividing the average PIM interval (T_{INT}) by the average number of slots ($\overline{N_S}$), as shown by

$$T_{P_{PIM}} = \frac{T_{INT}}{\overline{N_S}} \qquad 1.15$$

The average number of slots can be calculated by summing the number of slots that are contained in each possible "2^n" encoded PIM intervals (In_x), which are weighted by the probability of their occurrence (PR_x). This gives

$$\overline{N_S} = \sum_{x=1}^{x=2^n} PR_x \cdot In_x = \sum_{x=1}^{x=2^n} PR_x \cdot (x + m) \qquad 1.16$$

where the number of slots that are contained in the PIM interval are equal to "x+m". In the case where a NRZ PRBS is encoded, each PIM interval will occur with an equal probability and the expression for the average number of slots can be simplified into

$$\overline{N_S} = \frac{2^n + 1}{2} + m \qquad 1.17$$

where "n" is the number of binary bits that were encoded, and "m" is the number of slots required for the guardtime.

For the 3B8P1G PIM code, the average number of slots per encoded NRZ binary data sequence is 5.5. Using the correct number of slots increases the width of the pulse and decreases the magnitude of the RDS, as shown in Figure 1.7d. Since the average PIM interval is now equal to the width of the encoded NRZ binary bit sequence, the only variations that occur in the RDS will be due to statistical variations in the encoded data. Since there is no coding mechanism in the "nB2^{n}PmG" PIM code to limit the RDS, the buffers that are required would have to be infinite in size to prevent errors due to buffer overflow from occurring. But by increasing the complexity of the PIM coding scheme, it is possible to reduce the buffers to a finite size or to eliminate them, thus preventing buffer overflow errors.

1.4.2.2.1 "nB2·2^{n}PmG" PIM

In the past, a synchronous version of the double length "nB2·2^{n}PmG" PIM code has been implemented into a low speed atmospheric optical communication system[44]. It operates within a fixed frame width which contains only a single pulse. This implementation of the PIM code solves the problem of the variable frame width but only at the loss of its ability to resynchronize on the next received pulse.

Thus, the asynchronous version of the double length "nB2·2^{n}PmG" PIM code was developed to eliminate the problems that occur due to buffer overflow in the "nB2^{n}PmG" PIM code while retaining the ability to resynchronize on the next received pulse. Here a binary data sequence of length "n" is encoded into "2·2^n" possible intervals with a possible guardtime "m". This creates twice as many intervals than are required to encode all the possible combinations of the binary sequences. By encoding each binary sequence to two intervals, the RDS can be modified by sending either a short interval to increase the RDS or a long interval to decrease the RDS. Thus, the range of RDS values could be limited, allowing finite buffers to be used.

To better illustrate this, Figure 1.8 shows an example of the two encoded PIM intervals for the 3-bit binary sequence "011". Since no coding rules have yet been defined, two intervals shall be arbitrarily chosen to represent this sequence — one interval longer and one interval shorter than the average time frame width

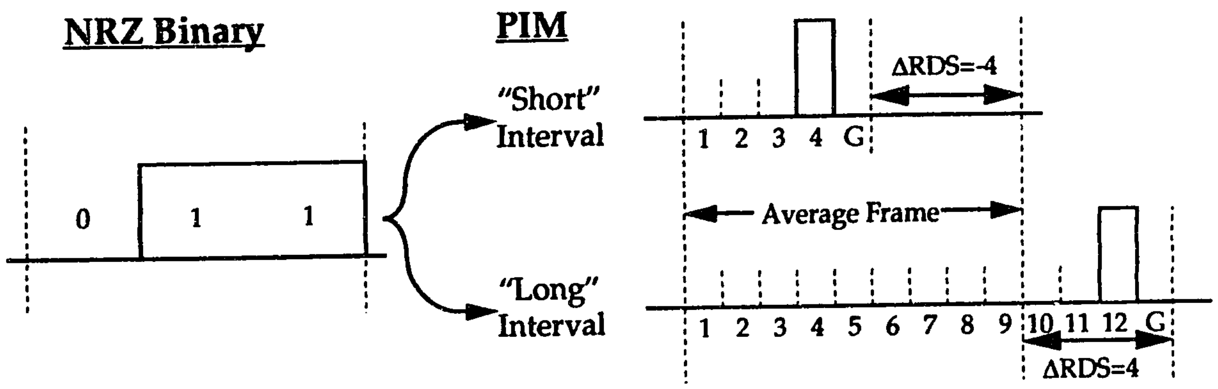


Figure 1.8 An example of the "nB2·2^nPmG" PIM intervals for a 3-bit sequence.

of nine slots. The encoder has the choice of either encoding the PIM pulse into the fourth interval, which decreases the current value of the RDS by four, or the twelfth interval, which increases the current value of the RDS by four. The interval selected depends on the previous value of the RDS and on a decision criterion that is defined by the coding rules.

The average number of slots ($\overline{N_S}$) that are contained in the "nB2·2^nPmG" PIM interval does not rely on the probability of occurrence of the bit sequence encoded, but is dependent on the coding scheme used to control the RDS. It can be calculated by taking the average of the maximum "2·2^n" and minimum "1" number of slots that are contained in the PIM intervals and adding the number of slots required by the guardtime "m". This gives

$$\overline{N_S} = \frac{2 \cdot 2^n + 1}{2} + m = 2^n + m + 1/2 \qquad 1.18$$

An example of the 3B16P1G PIM code is shown in Figure 1.7e. In this case a 3-bit binary sequence is encoded with a one-slot guardtime. This gives an average interval length of 9.5 slots. Because of the nature of the code, it is possible to vary the number of slots in the average interval by ±1/2 slot, or in this case between 9 and 10 slots without disrupting the ability of the code to control the RDS. In the example for the 3B16P1G PIM code, an average interval of 9 slots was used, which gives it the same pulse width as the 3B8P1G PPM code. Thus, the peak power gain and the bandwidth power penalty are identical for these two codes.

Similarly, it can be shown that the "nB2·2^nPmG" PIM and PPM code have the same effective power gains of

$$P_{EG} = \frac{n}{2} \qquad 1.19$$

where the assumptions for the noise power penalty concerning the shape of the spectral power density curves still hold. Although this indicates that the effective power gain can be increased indefinitely by encoding larger and larger NRZ binary bit sequences, the maximum power gain will be limited by such factors as the bandwidth or the maximum peak power of devices used in the SMF system.

The double encoding of the data in the "nB2·2ⁿPmG" PIM code eliminates errors due to buffer overflow by limiting the RDS to finite values. It also has the same effective power gain as the ideal PPM code with the added advantages of being self synchronizing. Thus, the "nB2·2ⁿPmG" PIM code was selected for implemention into an experimental test system.

The performance of the "nB2·2ⁿPmG" PIM code is summarized in the following sections by examining how it addresses the factors that limit the performance of codes.

1) Maximum brightness: The increase in brightness is inversely proportional to the duty cycle (D_{PIM}). This gives a maximum peak power of

$$P_{PIM} = \frac{1}{D_{PIM}} = 2^n + m + \frac{1}{2} \qquad 1.20$$

where "n" is the the length of the encoded binary sequence and "m" is the number of slots required for the guardtime. To obtain an accurate estimate of the effective power gain, the noise power penalty due to the increase in bandwidth has to be taken into account. Thus, the effective power gain of the PIM code over an equivalent NRZ binary code is given by

$$P_{EG} = \frac{n}{2} \qquad 1.21$$

2) Large spectral width: This factor causes a decrease in system performance because of intersymbol interference caused by

pulse broadening. The code allows a guardtime to be placed after the data pulse to act as a buffer, increasing the amount of pulse broadening that can occur before the ISI occurs.

3) Baseline wander: Because of the [illegible]y cycle of the code it is expected to contain small DC and low frequency components. This is confirmed in Section 2.3 where the theoretical power spectrum is calculated. Thus, the code will have a smaller magnitude of baseline wander as compared to the NRZ binary code.
4) Clock Content: There is a decrease in the number of transitions in the signal after encoding, indicating a decrease in the clock content of the code. But accurate clock recovery may not be required for the PIM code. Since only the time between adjacent pulses is needed rather than the absolute time of each pulse and the interval needs only to be measured to the nearest whole number, it may be sufficient to use a free running clock that is resynchronized at each pulse[45].
5) Simplified coding rules: The coding rules that map the different NRZ bit sequences to the discrete intervals and those that define the decision criteria for controlling the RDS are examined in Section 2.1. But for now it is sufficient to indicate that there are coding rules for the code which will allow it to be implemented into high speed systems.
6) Bit sequence independence: The performance of the code is expected to be independent of the NRZ binary data sequence encoded. This is because there is an average of one pulse per encoded data sequence, with the maximum spacing between

pulses being limited by the maximum value of the RDS.

7) Error multiplication: An increase in the error rate occurs due to the multi-level and the error propagation characteristics of the code. The increase in the error rate because of the multi-level coding scheme depends on the coding rules, number of bits encoded and the hardware implementation. This error rate will be examined for specific examples of the code in Section 2.4. The error propagation factor of the PIM code increases the overall error rate by a factor of two. Errors can be detected at the decoder by an eventual overflow of the buffers, but due to the statistical nature of the received data, it is not possible to identify the PIM frame responsible for the error. Thus, error correction could be very difficult to implement.

1.5 Thesis Objectives and Organization

1.5.1 Objectives

The purpose of the thesis is the development and the experimental implementation of a code that would increase the viability of LEDs on SMF systems. The "$nB2{\cdot}2^nPmG$" family of PIM codes is shown to be suitable for this application. These codes increase the power coupled into the fiber by allowing the LED to be overdriven and counteract pulse broadening by the placement of guardtimes after the information pulses. The 3B16P1G PIM code was selected for implementation into an experimental system. The specific goals of the research are:

1) To examine and select the coding rules for the 3B16P1G PIM code and to calculate their characteristics.
2) To design and construct a 90 Mbit/s 3B16P1G PIM encoder and

decoder for testing over a coaxial channel.

3) To compare the theoretically derived power spectrum with the measured power spectrum of the 3B16P1G PIM code.
4) To experimentally examine the effect on the bit error rate (BER) for variations in SNR, threshold level and bandwidth for the 3B16P1G PIM code.
5) To measure the experimental power gain and error multiplication factors and compare them to the theoretically derived values of the 3B16P1G PIM code.

1.5.2 Organization

The remainder of this thesis is divided into four major areas. Each one represented by a chapter which contains the following material:

1) Chapter 2: Gives a detailed description of the 3B16P1G PIM code. Then we investigate the coding rules, theoretical power spectrum, error multiplication and the BER of the code.
2) Chapter 3: Discusses the design and construction of the hardware required for the experimental system. This includes the PIM encoder, the PIM decoder, the channel and the receiver, which consists of a noise generator, an amplifier, a low-pass filter, and a DC level adjuster.
3) Chapter 4: Contains the experimental results on the performance of the 3B16P1G PIM code. These include the power spectral density as well as the BER as a function of system bandwidth, threshold level and SNR of the PIM code. These are compared to similar results obtained on the performance of the NRZ code in the same experimental system.

4) Chapter 5: Consists of a summary of the results and the conclusions derived from them. Also, recommendations are given for potential areas of further research.

CHAPTER 2

THE 3B16P1G PIM CODE

In the previous chapter, the properties of the "nB2·2^{n}PmG" family of PIM codes was explained, as well as why they were selected for implementation into the experimental system. Of the different possible variations of the "nB2·2^{n}PmG" family of PIM codes, the 3B16P1G PIM code was selected. In this code each PIM interval represents a 3-bit NRZ binary sequence which is buffered from the following PIM interval by a one-slot guardtime. These are considered as the minimum values required to test the feasibility of this family of codes, while still keeping the hardware required for the coder and decoder to a reasonable size.

The average interval for the 3B16P1G PIM code, calculated using Equation 1.18, is nine and one-half slots. To simplify the clock generation circuitry required in both the encoder and decoder, an average PIM interval of nine slots was selected. With this choice, the width of a NRZ binary bit is equal to the width of three PIM slots, so that the clock signal required for the logic contained in the NRZ section of the hardware can be generated from the PIM clock signal with a simple divide-by-three digital circuit. The average interval length of nine slots will be used for all further analysis of the 3B16P1G PIM code, unless otherwise indicated.

2.1 Coding Rules

In Chapter 1, the example of the 3B16P1G PIM code, displayed in Figure 1.7, was generated with little explanation of the coding rules used. The coding rules that define a specific PIM code can be divided into two groups. First, there are the coding rules required to map each of the NRZ binary sequences to their corresponding two PIM intervals. Second, there are the coding rules required to select one of the two PIM intervals in order to constrained the RDS to a finite value.

2.1.1 Mapping Coding Rules

For the "nB2·2^nPmG" PIM family of codes, each NRZ bit sequence combination is mapped to two different intervals, one longer and one shorter than the average PIM interval. In normal mapping assignments, the intervals selected for each bit sequence are used to optimize the code for either its transmission through a system or its implementation into hardware.

In commercial systems, the bit sequences which represent the information to be encoded do not have an equal probability of occurrence. Thus, it is possible to modify the power spectral density of the encoded PIM data stream by changing the mapping assignments of the binary bit sequences. This would allow the performance of the system to be optimized by matching the power spectral density of the PIM code to the channel characteristics. However, since the 3B16P1G PIM code is to be compared and tested against a NRZ PRBS, where each bit sequence occurs with equal probability, mapping assignments will only be examined for the purpose of simplifying the implementation of the PIM code into hardware.

To better illustrate the mapping process, three examples of mapping assignments for the 3B16P0G (no guardtime) and the 3B16P1G PIM code (guardtime=1 slot) are given in Table 2.1. All of the mapping assignments fulfil the basic requirement that each bit sequence must be represented by two unique PIM intervals, one shorter and one longer than the average PIM interval. The first mapping example, labelled "Random", demonstrates the flexibility of selecting PIM intervals for the individual bit sequences. The other two examples of the mapping assignments are structured in the sense that the two PIM

Table 2.1 Examples of the mapping assignments for the 3B16PmG PIM code.

Code	Binary	PIM											
Type	3-bit	3B16P0G						3B16P1G					
Mapping	-	Random		Eight-Plus		Complementary		Random		Eight-Plus		Complementary	
	000	6	15	1	9	1	16	7	16	2	10	2	17
	001	3	12	2	10	2	15	4	13	3	11	3	16
	010	1	10	3	11	3	14	2	11	4	12	4	15
	011	8	14	4	12	4	13	9	15	5	13	5	14
	100	5	9	5	13	5	12	6	10	6	14	6	13
	101	2	16	6	14	6	11	3	17	7	15	7	12
	110	7	13	7	15	7	10	8	14	8	16	8	11
	111	4	11	8	16	8	9	5	12	9	17	9	10

intervals that represent each bit sequence are interrelated. In the case of the "Eight-Plus" mapping rules, the long interval is eight slots wider than the short interval. Similarly, for the "Complementary" mapping rules, the long interval is the complement of the short interval so that the combined width of the two PIM intervals is a constant with a value dependent on the characteristics of the PIM code

Of these three schemes the "Eight-Plus" mapping rules were selected because they simplify the design of the circuitry required in the encoder and decoder. This would perhaps be better understood if an example were given where the widths of the two PIM intervals for an encoded data sequence are displayed in a 4-bit binary format. In the case of the NRZ data sequence "101", it is either encoded into a short interval containing 6 slots (0101) or a long interval containing 6+8=14 slots (1101). As can be seen, the information on whether a short or a long interval has been encoded is contained in the first binary bit. This simplifies the design of the encoder by allowing either interval to be encoded by merely modifying the first binary bit. At the same time, since this data information is contained only in the last three bits, the first bit is not required during the decoding process, thus simplifying the decoder design. Because the eight-plus mapping rules have been selected for use with the 3B16P1G PIM code, they will be used as the default mapping values for further analysis of the code.

2.1.2 RDS Coding Rules

The RDS coding rules are used to select either a short or long interval in order to limit the RDS to finite values. As previously defined, the RDS is the number of PIM slots that the current PIM frame is ahead of its expected position. There are two different sets of RDS coding rules to be examined. They are:

1) Approximate coding rules: In this case the selection of the PIM

interval depends only on the current value of the RDS. If the RDS is positive (including zero), then a short interval is selected in order to decrease the RDS; but if the RDS is negative, then a long interval is selected in order to increase the RDS.

2) Exact coding rules: Here the selection of the PIM interval depends on the effect that the interval has on the the next or new RDS ($RDS_{(+)}$). This requires that the RDS coding rules look ahead by calculating the $RDS_{(+)}$ for both the short and long intervals of the encoded binary bit sequence, and select the interval which gives the smallest absolute value for the $RDS_{(+)}$. In the case where the absolute value of the $RDS_{(+)}$ calculated for both intervals are equal, additional rules are required to select the interval. Some possible rules that could be used in this situation would be to send a short interval, a long interval, alternate between a short or long interval or implement the approximate coding rules as defined above.

The value for the $RDS_{(+)}$ is dependent on the current value of the RDS, the length of the PIM interval encoded (N_{Sx}) and the average interval length ($\overline{N_S}$). As shown in

$$RDS_{(+)} = RDS + (N_{Sx} - \overline{N_S}) \qquad 2.1$$

the RDS varies according to the difference in length between the encoded PIM interval and the average PIM interval.

To help illustrate the differences between the exact and approximate coding rules, examples of the encoded PIM data sequence for the 3B16P1G PIM code using both sets of RDS coding rules are given in Figure 2.1. To further

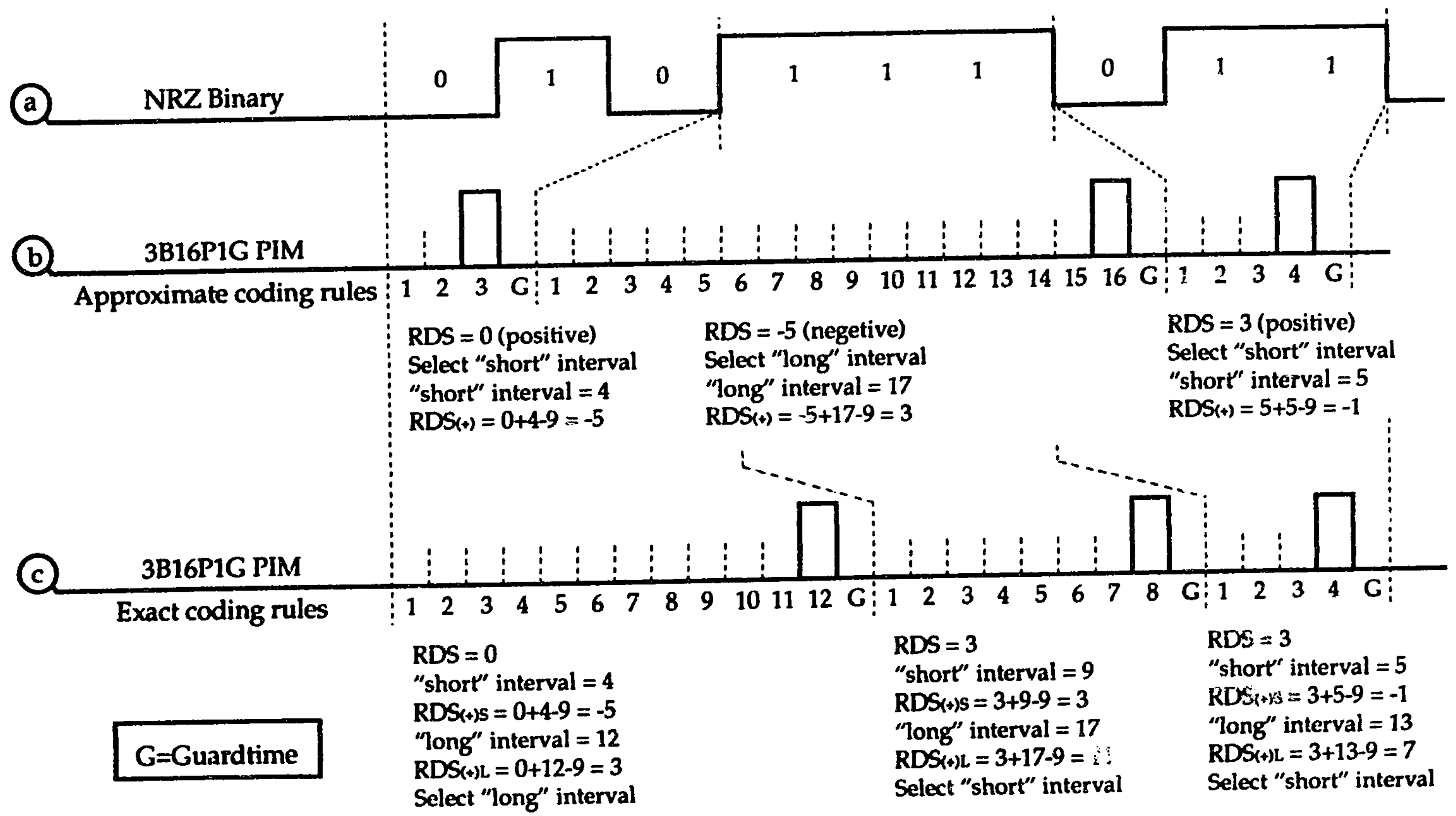


Figure 2.1 Examples of the exact and approximate coding rules for the 3B16P1G PIM code.

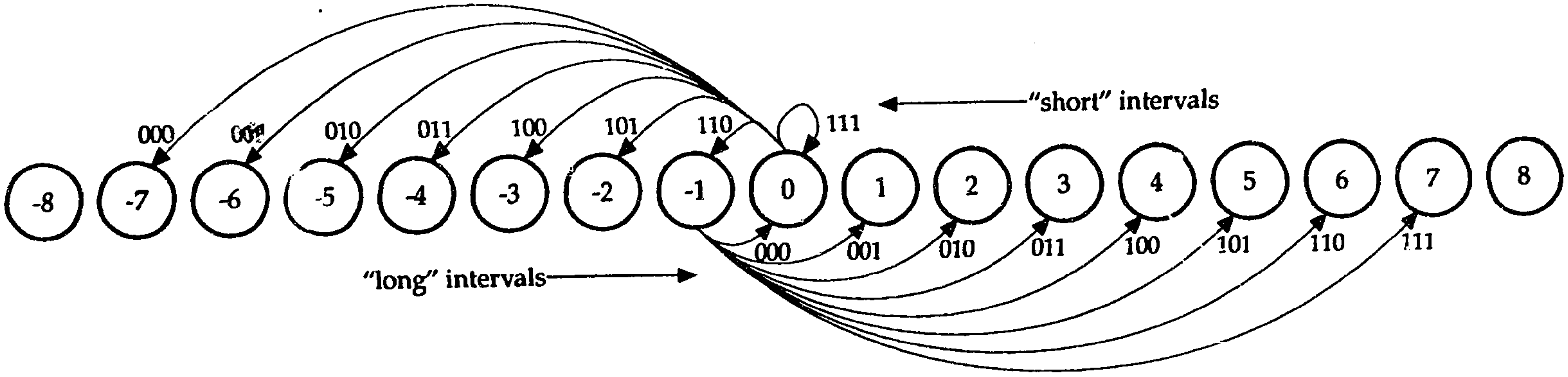
clarify the RDS coding rules, the relevant RDS calculations are displayed below their appropriate example. As can be seen, the exact coding rules require more complex calculations in order to make the correct PIM interval selection, but it appears to give less variation in the encoded interval lengths as compared to the approximate coding rules.

2.2 Buffer Requirements

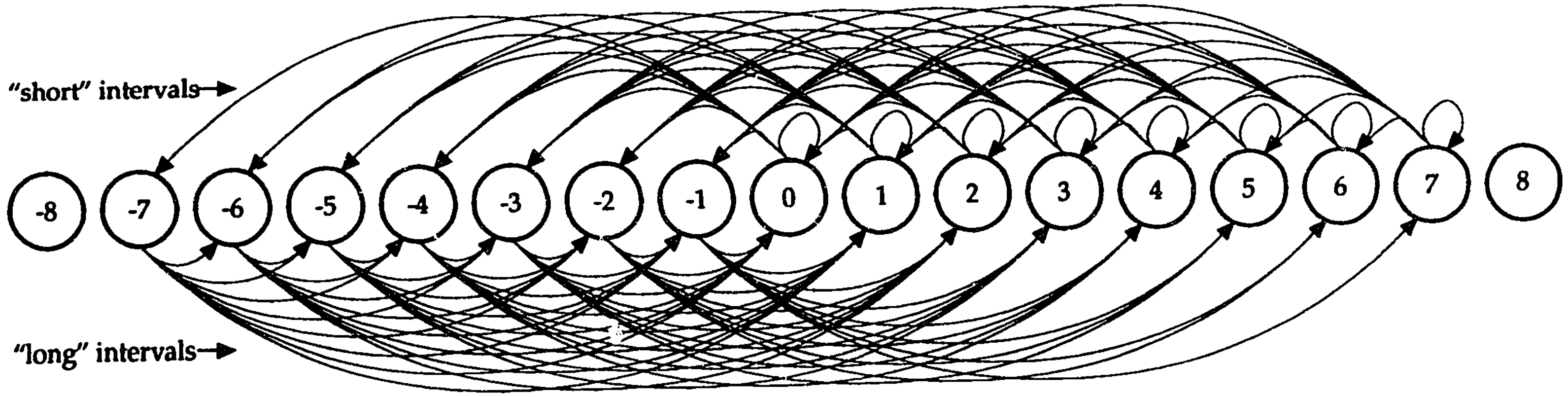
Because of the variable length of the interval in the PIM code, buffers are required in both the encoder and decoder. The use of the "nB2·2^nPmG" family of PIM codes reduces the buffers required to a finite size, where the required capacity is dependent on the length of the encoded data sequence, and the choice of the RDS coding rules. The size of the buffers required for the 3B16P1G PIM code when using either the exact or the approximate RDS coding rules will be examined. The first step is to determine the characteristics of the RDS for both sets of RDS coding rules. This is best done by constructing a state diagram where each RDS value is considered a separate state, and to map the next state or the $RDS_{(+)}$ for all possible data sequences for each state or value of the RDS.

For the approximate RDS coding rules, Equation 2.1 is used to calculate the $RDS_{(+)}$ which is the next state of the RDS. The next states for the "0" and the "-1" RDS states of the state diagram have been calculated and are displayed in Figure 2.2a for all possible data sequences. The complete state diagram of the RDS for the 3B16P1G PIM code, implemented using the approximate RDS coding rules, is shown in Figure 2.2b.

Similarly, the state diagram of the RDS for the exact RDS coding rules can be generated. Equation 2.1 is used to calculate the two next potential states, and the one with the smallest absolute value of the $RDS_{(+)}$ is selected. In the case where the absolute values of the $RDS_{(+)}$ for both of the PIM intervals are equal,



A) The state diagram for the 0 and -1 states of the RDS



B) The state diagram for all states of the RDS

Figure 2.2 State diagrams for the 3B16P1G PIM code, implemented using the approximate RDS coding rules.

the long interval will be used to calculate the next state or the value of the $RDS_{(+)}$. The next states for the "0" state of the state diagram have been calculated and are displayed in Figure 2.3a for all possible data sequences. The complete state diagram of the RDS for the 3B16P1G PIM code, implemented using the exact RDS coding rules, is shown in Figure 2.3b.

From the information contained in the state diagrams of the RDS, it is possible to calculate the probability distribution of the RDS states. The probability of the next RDS state (P_{S_i}) occurring is equal to the joint probability of the current states (S_j) and t[illegible]t state (S_i). This can be represented by

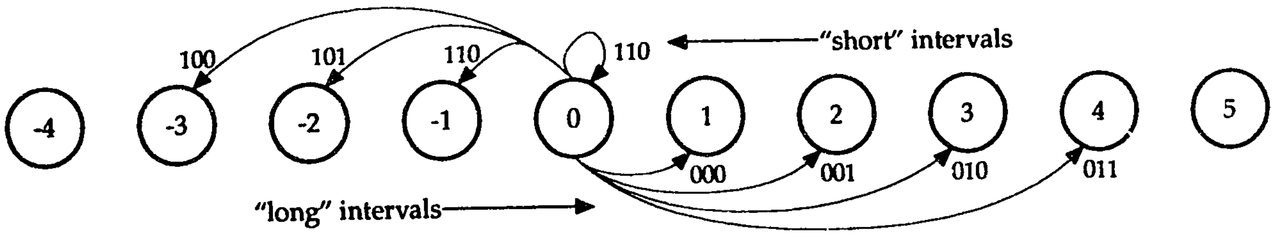
$$P_{S_i} = \sum_{j=-\infty}^{j=\infty} P_r\left(S_i \cap S_j\right) = \sum_{j=-\infty}^{j=\infty} P_{S_i/S_j} \cdot P_{S_j} \qquad 2.2$$

where "P_{S_i/S_j}" represents the conditional probability of the next RDS state "i", given the current RDS state "j". Since the encoded bit sequence is independent of the RDS, the probability of the RDS state "i" can be rewritten as

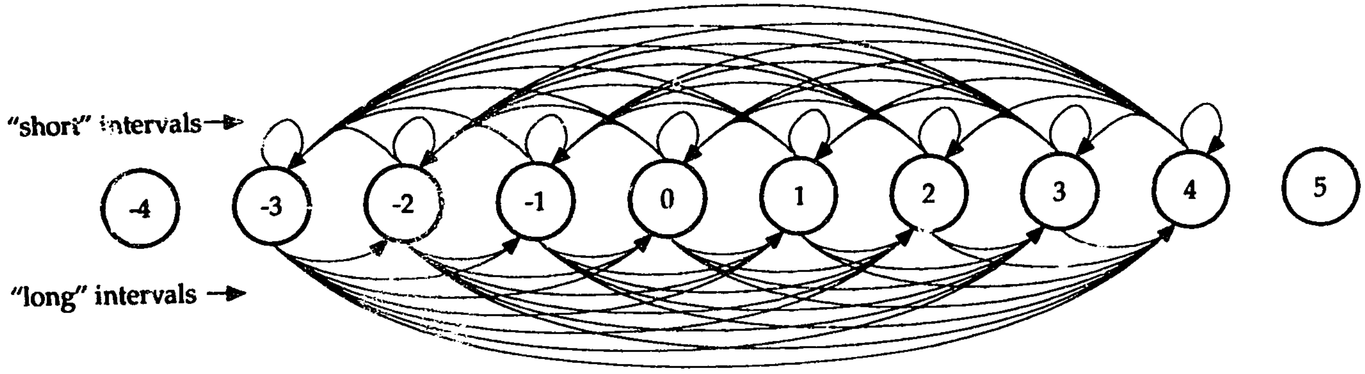
$$P_{S_i} = \sum_{j=-\infty}^{j=\infty} \sum_{k=0}^{k=2^n-1} A_{i,j,k} \cdot P_{S_j} \cdot P_{D_k} \qquad 2.3$$

where the conditional probability "P_{S_i/S_j}" is expressed in terms of "P_{D_k}" the probability of the encoded bit sequence "k", and "$A_{i,j,k}$" is the next state matrix. The next state matrix contains the information on whether or not the next state from the RDS state "j" for the data sequence "k" is the RDS state "i". This information is obtained from the state diagram of the RDS, where a link between two RDS states for a specific data sequence is represented in the next state matrix by a "1", and if no link exists by a "0".

In the case where a NRZ PRBS is to be encoded with the 3B16P1G PIM code, Equation 2.3 can be simplified. Since a pseudo-random data stream contains equi-probable data sequences, the term contained in Equation 2.3 for the



A) The state diagram for the 0 state of the RDS



B) The state diagram for all states of the RDS

Figure 2.3 State diagrams for the 3B16P1G PIM code, implemented using the exact RDS coding rules.

probability of a data sequence "P_{D_k}" becomes a constant that is equal to the inverse of the number of data sequences "2^n", where the length of the encoded NRZ data sequence is 3-bits. By substituting these expressions into the the above equation we obtain

$$P_{S_i} = \frac{1}{2^n} \sum_{j=-\infty}^{j=\infty} A_{i,j} \cdot P_{S_j} = 0.125 \sum_{j=-\infty}^{j=\infty} A_{i,j} \cdot P_{S_j} \qquad 2.4$$

Since the "nB2·2^nPmG" PIM family of codes have finite values for the RDS, the limits for "j" can be reduced to the maximum and minimum values of the RDS. These values can be found from the state diagram of the RDS and will vary depending on the coding scheme.

Equation 2.4 represents the set of linear equations that define the probability of the next RDS states. The solution for this set of linear equations for the probability of occurrence of the RDS must also satisfy

$$P_{S_i} = P_{S_j} \Big|_{i=j} \qquad 2.5$$

where the probability of a specific RDS state is independent of time, and

$$\sum_{j=-\infty}^{j=\infty} P_{S_j} = 1 \qquad 2.6$$

where the summation of the probabilities of all the possible RDS states is equal to one.

The probability distribution of the RDS states for both the approximate and exact coding rules were solved by using Equations 2.4, 2.5 and 2.6, and the results are shown graphically in Figures 2.4 and 2.5, respectively. For confirmation of these results a Monte Carlo simulation[46] of the 3B16P1G PIM code for a randomly generated data sequence was conducted. A sample size of approximately 73000 data bits were used to generate the probability distributions

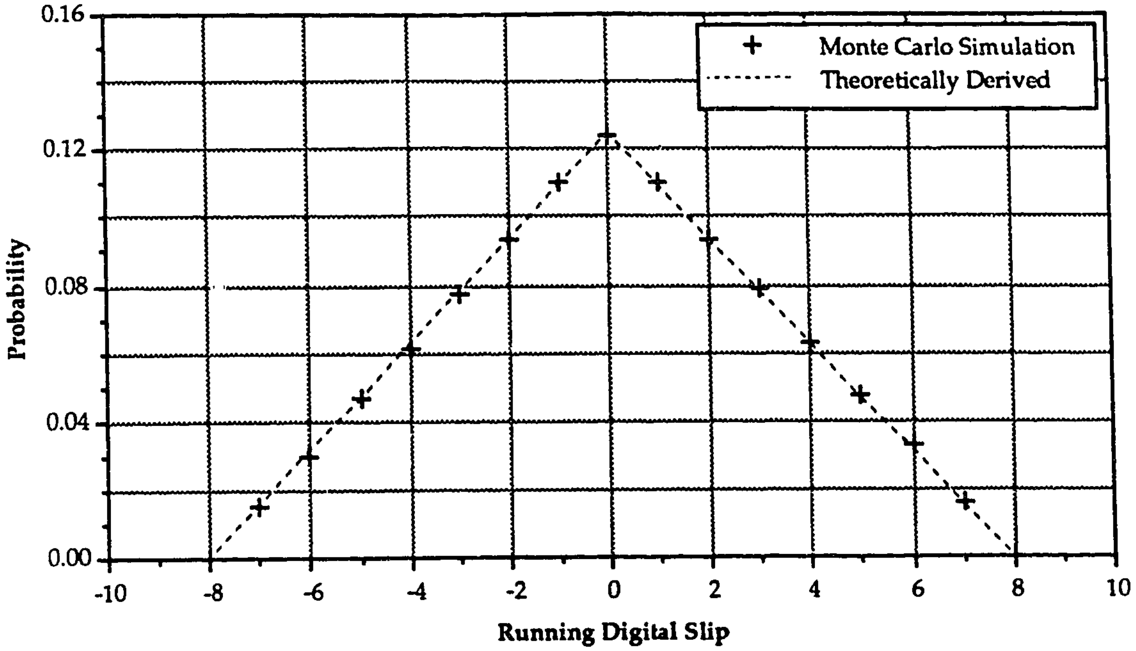


Figure 2.4 Probability distrubution of the RDS for the approximate coding rules.

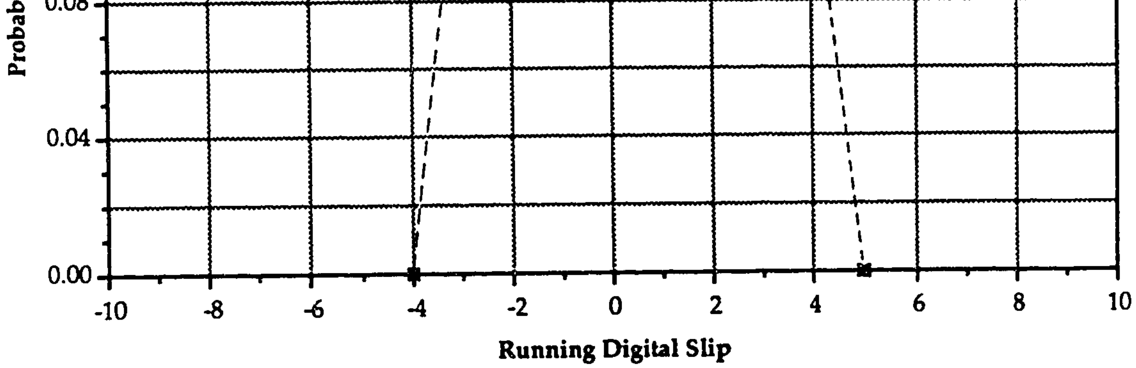


Figure 2.5 Probability distrubution of the RDS for the exact coding rules.

of the RDS for both the approximate and exact coding rules. These results were plotted against the calculated values of the probability distributions of the RDS given in Figure 2.4 and 2.5. Comparing the two methods, we find that the results of the simulation are in close agreement with the calculated theoretical probability distributions.

The probability distributions of the RDS for the approximate and exact coding rules differ not only in the number of existing RDS states they contain, but also in the shape of their probability distributions. The number of RDS states is directly related to the minimum buffer size. The probability distribution for the approximate RDS coding rules of the 3B16P1G PIM code contains fifteen RDS states and would require a buffer with the capacity to store at least fifteen PIM slots or five NRZ binary bits. Its RDS probability distribution is triangular in shape and is centered over the zero value of the RDS, thus giving this coding scheme a tendency to generate the smaller values of the RDS because of their higher probabilities of occurrence.

The probability distribution of the RDS for the exact RDS coding rules contains only eight RDS states and would require a buffer that is capable of storing at least eight PIM slots or 3 NRZ binary bits. Its probability distribution of the RDS is rectangular in shape, indicating that all the RDS states are equi-probable. This coding scheme will have a tendency to generate a more even spectrum of the RDS over a smaller range of values as compared to the coding scheme that uses the approximate coding rules.

The exact RDS coding rules require only one-half as much buffer capacity as is required for the case of the approximate RDS coding rules. This advantage that the exact RDS coding rules has over the approximate RDS coding rules is offset by an increase in the complexity of the decision logic that is required for the selection of either a short or long PIM interval. Thus, if this decision logic is implemented into hardware, it will increase the complexity of the logic circuitry, which could lead to a decrease in the maximum data rate that can be achieved (for a specific logic family).

Since the maximum data rate of the code was considered an important factor, the approximate RDS coding rules with their simpler decision logic, were selected to be implemented into the experimental encoder and decoder. They will require buffers with the capacity to store at least 5 NRZ binary bits. These buffers are relatively small in size, thus enabling the buffers to be implemented into the experimental system with minimal effects on the overall maximum data rate.

2.3 Power Spectral Density

The power spectral density is an important characteristic of a code and contains important information on its bandwidth requirements and potential performance. In the case of the PIM code, the calculation of the theoretical average power spectrum can be quite involved due to complications arising from the variable frame width and the interdependence of the encoded PIM intervals. Thus, an alternative method, using a Monte Carlo simulation[46] of a random data sequence, to determine the representative spectral characteristics, was employed

The Monte Carlo simulation algorithm that was applied in the previous section was used to generate a random NRZ data sequence which was encoded using the approximate RDS coding rules for the 3B16P1G PIM code. The length of the generated PIM data sequence was 32,768 PIM slots which is equivalent to 3,460 PIM intervals or 10,922 NRZ data bits. This PIM data stream was transformed into the frequency domain with a 65,536 point FFT to obtain the average impulse spectrum for the 3B16P1G PIM code. Squaring the spectral density components of this average impulse spectrum gives rise to the average impulse power spectrum which is shown in Figure 2.6, where the amplitude of the power spectrum has been normalized to the magnitude of the discrete DC component, and the frequency has been normalized to the PIM slot width (T_P). The shape of the average impulse power curve repeats to infinity, so only one cycle between the normalized frequencies of "0" to "1" is displayed. The advantage of the impulse power spectrum is that it is independent of the pulse shape of the data stream. But if the power spectrum of the code is required for a specific pulse shape, it can be easily obtained from the impulse power spectrum, if the power spectrum of the pulse is known. This is done by multiplying the impulse power spectrum of the code by the the power spectrum of the pulse shape in the frequency domain which corresponds to the convolution of the pulse shape with impulse data stream in the time domain.

The average impulse power spectral density of the 3B16P1G PIM code consists of both discrete and a continuous components. The discrete components of the spectrum occur at DC and at integer multiples of the normalized frequency (1,2,3,4,5...). The continuous spectrum contains only a small amount of DC and low frequency components as alluded to in Section 1.4.2.2.1. This is beneficial because it minimizes the effect that AC coupling has on the signal.

2.4 Error Multiplication

An examination of the error mechanisms of the 3B16P1G PIM code is necessary for a full understanding of how error multiplication occurs in the decoding process for an erroneously received PIM interval. As in the case of the NRZ binary code, the PIM code represents the encoded data with a series of "1's" and "0's" which correspond to two distinct levels in the transmitted signal. The two levels of the signal can be represented by an amplitude of "A" for the transmission of a "1" and an amplitude of zero for the transmission of a "0".

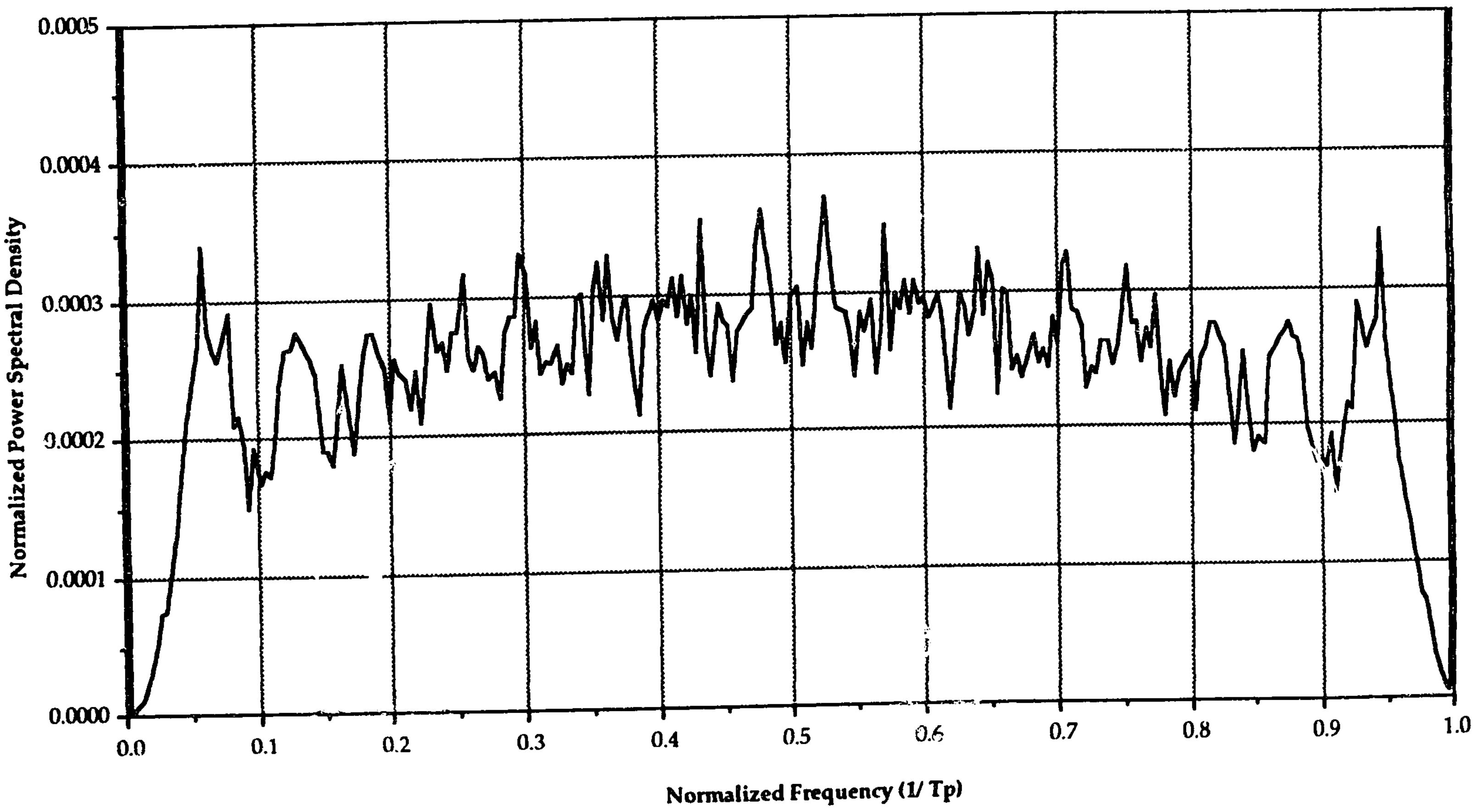


Figure 2.6 Impulse power spectrum of the 3B16P1G PIM code using the approximate RDS coding rules.

Errors occur when the amplitude of the signal is modified by noise to such a degree that an incorrect decision is made by the threshold detector in the decoder. There are two types of error, threshold violation and erasure errors. Threshold violation errors occur where a "0" is transmitted but the amplitude of the signal due to the addition of noise exceeds the threshold level at the sampling point and a "1" is detected. Likewise, erasure errors occur where a "1" is transmitted but the amplitude of the signal due to the addition of noise falls below the threshold level at the sampling point and a "0" is detected. Each error that is detected in the PIM data stream will cause several bit errors when decoded back into the original NRZ data stream. This increase in the error rate is due to a combination of three different factors: multi-level decoding errors, error propagation and data destruction or creation.

Multi-level codes take several bits of data and encode them into a symbol. If the symbol is incorrectly received, some or all of the decoded data bits will be in error. Thus, an error in detecting the symbol will be multiplied in the decoded data sequence and will increase the error rate. In the case of the 3B16P1G PIM code, each PIM interval represents three NRZ binary bits. So for every PIM interval that is incorrectly detected, up to three errors can occur in the decoded NRZ data sequence. The actual increase in the error rate is dependent on the coding rules employed and the statistics of the encoded data.

Error propagation and data destruction or creation take place in the 3B16P1G PIM code due to the interdependence of the PIM intervals. This is because the pulse of the PIM code is used not only to encode the information contained in the current PIM interval, but also to act as the framing pulse for the next PIM interval. Thus, any error will affect two PIM intervals giving an error propagation factor of two as well as creating or destroying a PIM interval, depending on the type of error.

This destruction or creation of data is illustrated by the two examples of the 3B16P1G PIM code for the encoded NRZ data sequence "110100" which are displayed in Figure 2.7. In the case of a violation error, an extra pulse is detected which divides the erroneous PIM interval into two as shown in Figure 2.7a. This creates an extra PIM interval which will increase the length of the decoded NRZ data sequence by 3-bits as compared to the encoded NRZ data sequence. Similarly, when an erasure error occurs, the two PIM intervals are combined into one as shown in Figure 2.7b. This destruction of a PIM interval will decrease the length of the decoded NRZ data sequence by 3-bits as compared to the encoded NRZ data sequence.

If the NRZ data sequence is position dependent, then once an error occurs and the decoded NRZ data stream is shifted out of position by 3-bits, any data received after this point will be registered erroneously until the the NRZ data stream is shifted back to its original position. This change in the length of the NRZ data sequence will eventually cause the RDS to exceed its prescribed limits, indicating that an error has occurred. Because the size of the buffer is directly proportional to the number of RDS states, the buffer will eventually overflow or underflow when the limits of the RDS have been exceeded. This produces an error that will shift the NRZ data stream back into its original position ending the errors due to the position dependency of the NRZ data sequence. Further analysis of position dependent errors would require extensive information on the characteristics of the implemented buffer along with a more comprehensive model of the 3B16P1G PIM code.

In order to examine the error multiplication of the 3B16P1G PIM code, the values of the error multiplication factors for both threshold violation and erasure errors will be calculated for the encoded NRZ data streams "1111..." and "0000...". These NRZ data sequences were selected because they are position

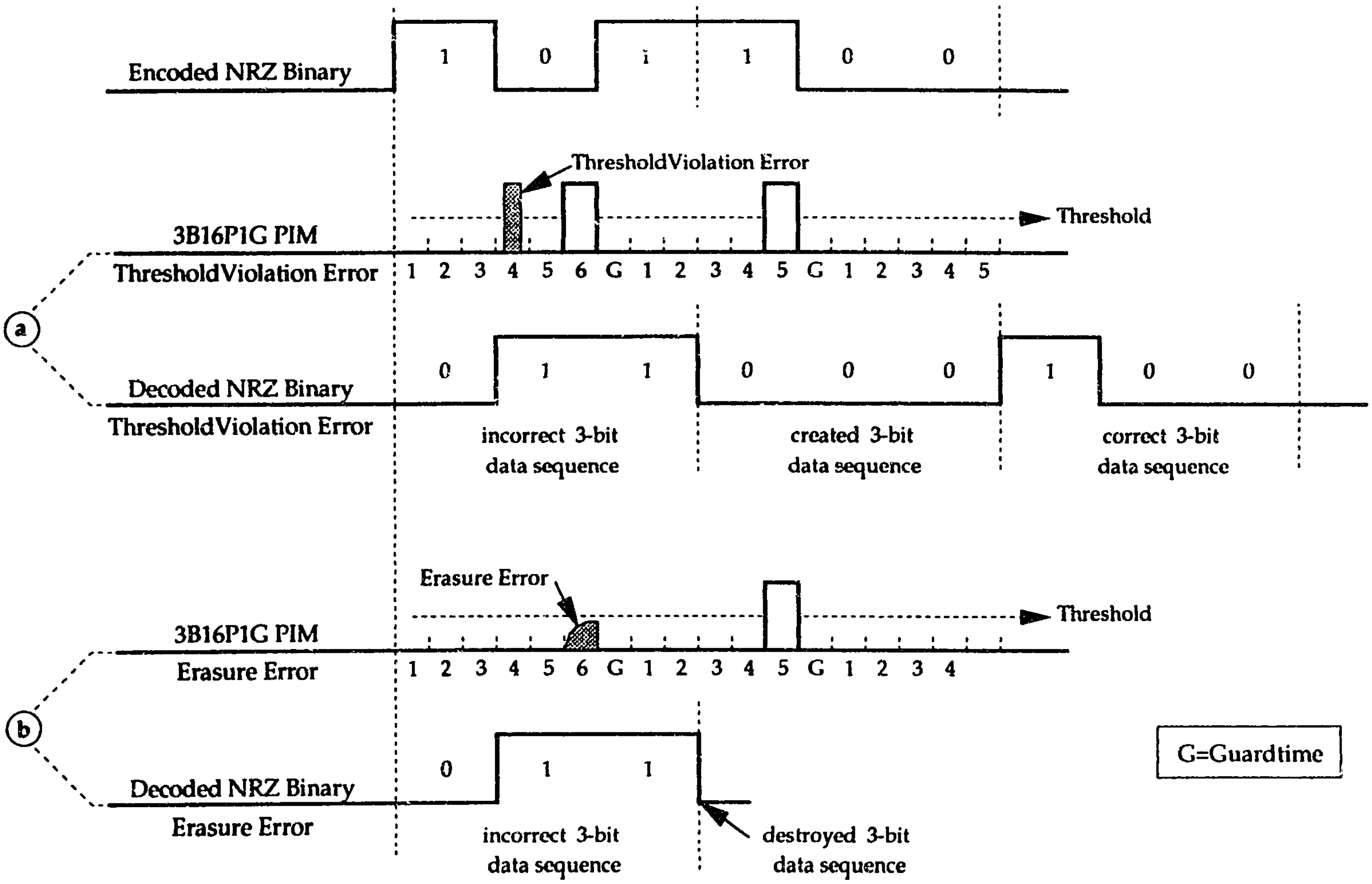


Figure 2.7 Examples of the creation and destruction of data due to errors.

independent and their error rate will not be affected by the shifting of the decoded data stream due to the creation and destruction of data.

For the encoded NRZ data stream "1111..." the PIM data sequence, displayed in Figure 2.8, contains only the short intervals of the data sequence "111". This continuous stream of short intervals occurs because the short interval is equal in width to the average PIM interval; thus the RDS remains constant. Erasure errors can only occur when the pulse that is positioned in the eighth slot of the PIM interval is attenuated to such a degree that it is not detected at the sampling point; thus the length of the interval that is detected at the decoder will be eighteen slots as shown in Figure 2.8. When decoded this gives the NRZ data sequence "000". Since the encoded NRZ data sequence was "111", this single erasure error gives rise to three errors in the decoded NRZ data stream which correlates to an error multiplication factor (E_{m1}) for erasure errors of three.

Threshold violation errors can appear at eight different positions in a PIM interval, and they occur when the the magnitude of the noise exceeds the threshold level, causing an erroneous pulse to be detected. As shown in Figure 2.8, this pulse will either decrease the width of the PIM interval by one slot if the error occurs in either the guardtime or the seventh slot, or it will divide the PIM interval into two erroneous intervals if the error is detected in any of the other remaining empty slots. In the situation where the two PIM intervals are formed, the first of these intervals incorrectly represents the original PIM interval while the second interval represents data that has been created due to the error. In either case, the location of the slot in which the threshold violation error occurs will determine the number of errors that the decoded NRZ data sequence will contain. These decoded NRZ data sequences, along with the number of erroneous bits that they contain are listed in Table 2.2 for each of the eight possible threshold violation error that can occur in the PIM interval.

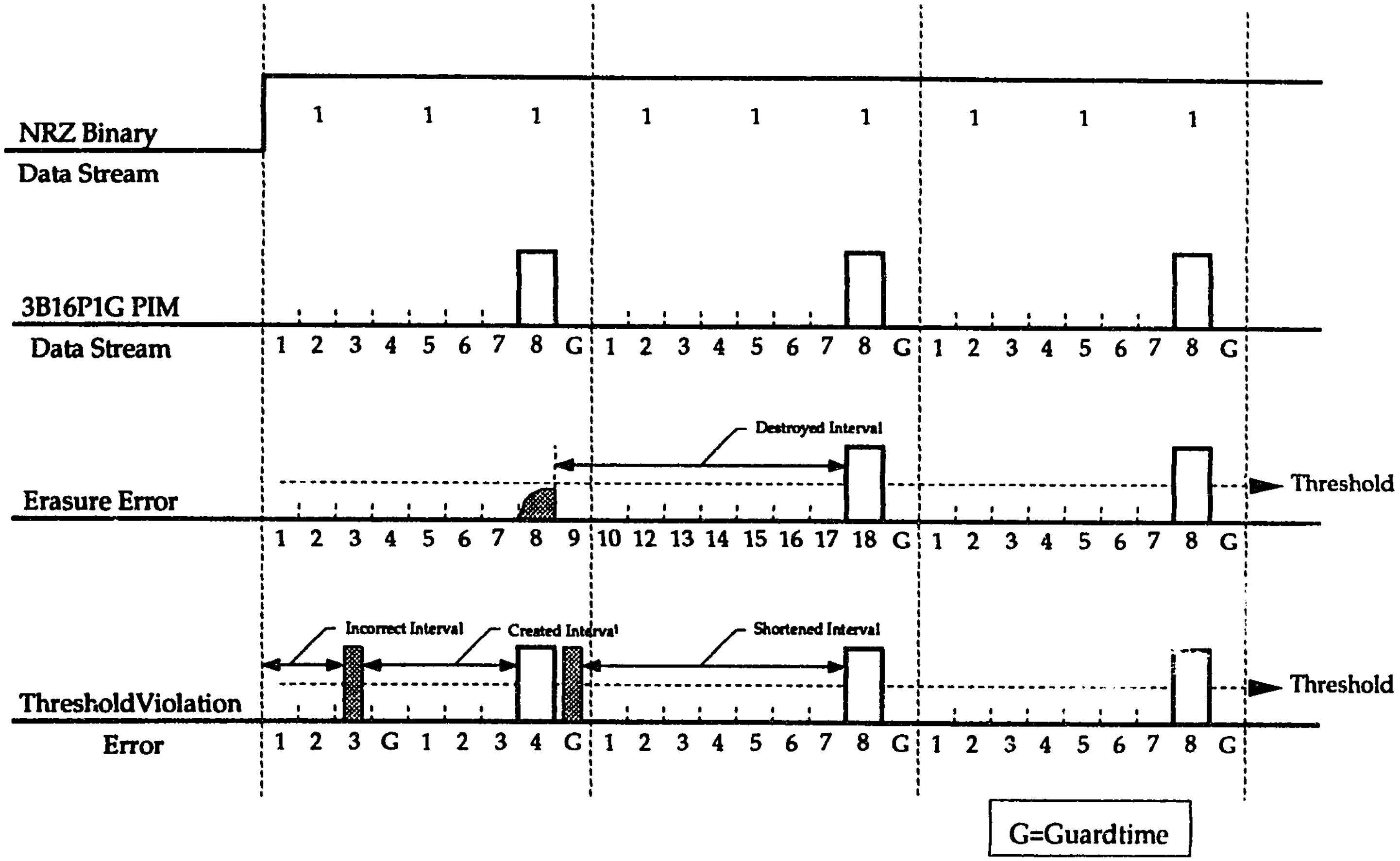


Figure 2.8 Error multiplication factors of the 3B16P1G PIM code for the encoded data stream "11111..."

Table 2.2 Threshold violation error analysis for the 3B16P1G PIM encoded "1111..." data stream.

Position of error	Incorrect 3-bit sequence	Created 3-bit sequence	NRZ bit errors
Guardtime Slot	110	-	1
1-Slot	000	101	4
2-Slot	001	100	4
3-Slot	010	011	3
4-Slot	011	010	3
5-Slot	100	001	4
6-Slot	101	000	4
7-Slot	110	-	1
Total # of Slots			8 slots
Total # of Errors			24 errors
Error Multiplication Factor E_{m_1}			3

Since there is an equal probability of a threshold violation error to occur in any of the eight slots, the value of the error multiplication factor (E_{m0}) due to these errors would be equal to the average number of errors that occur in the NRZ data stream per threshold violation error. In the case of the NRZ data stream "1111...", a total of twenty-four errors occur in the decoded NRZ data stream in response to the eight possible threshold violation errors. This gives an error multiplication factor (E_{m0}) for threshold violation errors of three. Thus, for the NRZ data stream "1111...", the value of the error multiplication factor is independent of the type of error that occurs. This is not always the case. As an example, for NRZ data stream "0000...", where a similar approach was used, the error multiplication factors are found to be "1" and "2.625" for erasure and threshold violation errors respectively. The values for the error multiplication factors for the NRZ data streams "1111..." and "0000..." are listed in Table 2.3.

2.5 Theoretical Bit Error Rate

The bit error rate (BER) is a measurement of the performance of a code and can be used for comparing the performance of different codes or for the optimization of a communication system. It is defined as the normalized number of errors that occur in the decoded binary data stream. So, for example, a BER of 10^{-6} would represent one error in a data stream that contains one million bits.

The theoretical BER, also known as the probability of error, is the predicted value of the expected BER of a system. Generally, in the case of a binary or a two level signal, the probability of error (P_E) is equal to the sum of the probability of errors due to threshold violation errors (P_{E_0}) and erasure errors (P_{E_1}). These terms are modified by the probability of a "0" (P_0) occurring, during which only threshold violation errors can occur, and the probability of a "1" (P_1) occurring during which only erasure errors can occur. Thus, the probability of

Table 2.3 Error multiplication factors for the 3B16P1G PIM encoded "1111..." and "0000..." data streams.

		Error Multiplication Factors	
		(E_{m0})	(E_{m1})
Source of Error		Threshold violation	Erasure
Encoded Data Streams	1111...	3	3
	0000...	2.625	1

error can be written as

$$P_E = P_0 \cdot P_{E_0} + P_1 \cdot P_{E_1} \qquad 2.7$$

This expression for the probability of error does not take into account the effects of error multiplication that can occur in some codes. This can be remedied by modifying Equation 2.7 to include the error multiplication factors for both the threshold violation errors (E_{m_0}) and the erasure errors (E_{m_1}). This gives

$$P_E = E_{m_0} \cdot P_0 \cdot P_{E_0} + E_{m_1} \cdot P_1 \cdot P_{E_1} \qquad 2.8$$

The probability of P_0 and P_1 can be expressed in terms of the code's duty cycle (D_{xxx}). As defined in Chapter 1, the duty cycle is the ratio of the time that the signal is high to the total time of the signal, which is equal to the probability of a "1" (P_1) occurring. Since the sum of the probabilities of occurrence are equal to unity, the probability of "0" (P_0) occurring can be expressed as one minus the duty cycle. Thus, Equation 2.8 can be rewritten as

$$P_E = E_{m_0} \cdot P_{E_0} \cdot (1-D_{xxx}) + E_{m_1} \cdot P_{E_1} \cdot D_{xxx} \qquad 2.9$$

In the situation where the noise contained in the received signal is additive white gaussian noise (AWGN) with a zero mean and a constant variance equal to the noise power (N), the probability of error for a threshold violation error (P_{E_0}) and an erasure error (P_{E_1}) for a threshold level (T_L) are given by[47]

$$P_{E_0} = \int_{T_L}^{\infty} \frac{1}{\sqrt{2\pi N}} e^{-x^2/2N} dx = \mathrm{Erfc}\left(\frac{T_L}{\sqrt{N}}\right) \qquad 2.10$$

$$P_{E_1} = \int_{-\infty}^{T_L} \frac{1}{\sqrt{2\pi N}} e^{-(x-A)^2/2N} dx$$

$$= \int_{A-T_L}^{\infty} \frac{1}{\sqrt{2\pi N}} e^{-x^2/2N} dx = \mathrm{Erfc}\left(\frac{A-T_L}{\sqrt{N}}\right) \qquad 2.11$$

To calculate the probability of threshold violation error (P_{E_0}), where the signal is at "0" amplitude, the probability density function of the noise amplitude is integrated over those values that exceed the threshold level. Similarly, the probability of erasure error (P_{E_1}), where the signal is at "A" amplitude, can be calculated by integrating the probability density function of the noise amplitude over those values that reduce the signal to below the threshold level. Using a change of variables, these integrals can be modified into the form of the complementary error function (Erfc).

The Erfc function can be obtained from Tables, although in some situations accurate results can be achieved by using the approximation[48]

$$\mathrm{Erfc}(x) \approx \left[\frac{1}{\left(1-\frac{1}{\pi}\right)x+\frac{1}{\pi}\sqrt{x^2+2\pi}}\right]\cdot\frac{1}{\sqrt{2\pi}}\,e^{-x^2/2} \qquad 2.12$$

It has a maximum error that is less than 2% of the theoretical value over the range of all "x" and becomes increasingly accurate when used to evaluate the larger values of "x".

By substituting the expressions for the probability of threshold violation and erasure errors in to Equation 2.9, the expression for the probability of error can be rewritten as

$$P_E = (1-D_{xxx})\cdot E_{m_0}\cdot \mathrm{Erfc}\left(\frac{T_L}{\sqrt{N}}\right) + D_{xxx}\cdot E_{m_1}\cdot \mathrm{Erfc}\left(\frac{A-T_L}{\sqrt{N}}\right) \qquad 2.13$$

which is a function of the duty cycle and the error multiplication factors of the code, and the Erfc function. The duty cycle and the error multiplication factors contained in the probability of error expression can be rearranged so that they can be represented by the two constants: the effective error multiplication factor (E_{m_E}) and the effective duty cycle (D_{EFF}). Thus

$$P_E = E_{m_E}\cdot\left[(1-D_{EFF})\cdot \mathrm{Erfc}\left(\frac{T_L}{\sqrt{N}}\right) + D_{EFF}\cdot \mathrm{Erfc}\left(\frac{A-T_L}{\sqrt{N}}\right)\right] \qquad 2.14$$

where

$$D_{EFF} = \frac{D_{xxx} \cdot E_{m_1}}{D_{xxx} \cdot E_{m_1} + (1 - D_{xxx}) \cdot E_{m_0}} \qquad 2.15$$

and

$$E_{m_E} = D_{xxx} \cdot E_{m_1} + (1 - D_{xxx}) \cdot E_{m_0} \qquad 2.16$$

2.5.1 Optimum Threshold Level

The threshold level is an important parameter in the calculation of the probability of error of a code. Normally, the threshold level is selected to minimize the probability of error, or in an experimental system, the BER. The threshold level that accomplishes this goal is defined as the optimum threshold level ($T_{L_{opt}}$), which can be derived by differentiating the expression for the probability of error, as given in Equation 2.14, and setting it to zero. This gives the expression

$$0 = -(1 - D_{EFF}) \cdot e^{-T_{L_{opt}}^2/N} + D_{EFF} \cdot e^{-(A - T_{L_{opt}})^2/N} \qquad 2.17$$

which defines the relationship between the optimum threshold level, and the other parameters of the code for the minimum probability of error. By rearranging the terms, it is possible to isolate the optimum threshold level and by normalizing it to the pulse amplitude "A", we get the following expression

$$\frac{T_{L_{opt}}}{A} = \frac{1}{2} + \frac{N}{A^2} \cdot \ln\left(\frac{1 - D_{EFF}}{D_{EFF}}\right) \qquad 2.18$$

for the normalized optimum threshold level. The term that contains the amplitude of the pulse "A" squared and divided by the noise power "N" is recognized as the peak signal to noise power ratio (SNR_P). By substituting this back into the Equation 2.18, it can be rewritten as

$$\frac{T_{L_{opt}}}{A} = \frac{1}{2} + \frac{1}{SNR_P} \cdot \ln\left(\frac{1 - D_{EFF}}{D_{EFF}}\right) \qquad 2.19$$

Using Equation 2.19, the normalized optimum threshold level has been plotted in Figure 2.9 as a function of the effective duty cycle for three different values of the SNR_P.

The normalized optimum threshold level is calculated for a NRZ PRBS, and the 3B16P1G PIM code for the encoded NRZ data streams "1111..." and "0000...". In the case of the NRZ PRBS, which has a duty cycle of 0.5 and an error multiplication factor of "1" for both threshold violation and erasure errors, the effective duty cycle, calculated using Equation 2.15, was found to be equal to the actual duty cycle of the code. By substituting this value into Equation 2.19, one obtains

$$\frac{T_{L_{opt}}}{A} = \frac{1}{2} + \frac{1}{SNR_P} \cdot \ln\left(\frac{1 - 0.5}{0.5}\right) = \frac{1}{2} + \frac{1}{SNR_P} \cdot \ln(1) = 0.5 \qquad 2.20$$

where the term containing the SNR_P goes to zero which eliminates the effect that the SNR_P has on the optimum threshold level. Thus, the NRZ PRBS has a normalized optimum threshold level of 0.5 which is independent of the SNR.

Similarly, the normalized optimum threshold levels for the 3B16P1G PIM code were calculated at three different values of the SNR_P for the encoded NRZ data streams "1111..." and "0000...", where a duty cycle of 1/9 and the values for the error multiplication factors as shown in Table 2.3 were used. The values for the normalized optimum threshold level are displayed in Table 2.4 along with the effective error multiplication factor and the effective duty cycle for each of the PIM encoded NRZ data sequences. In both cases, due to the low duty cycle of the code, the normalized optimum threshold level exceeds the normalized threshold level of 0.5 that was calculated for the NRZ PRBS. Also, it is noticed that the larger the SNR_P the smaller the variation of the optimum threshold level

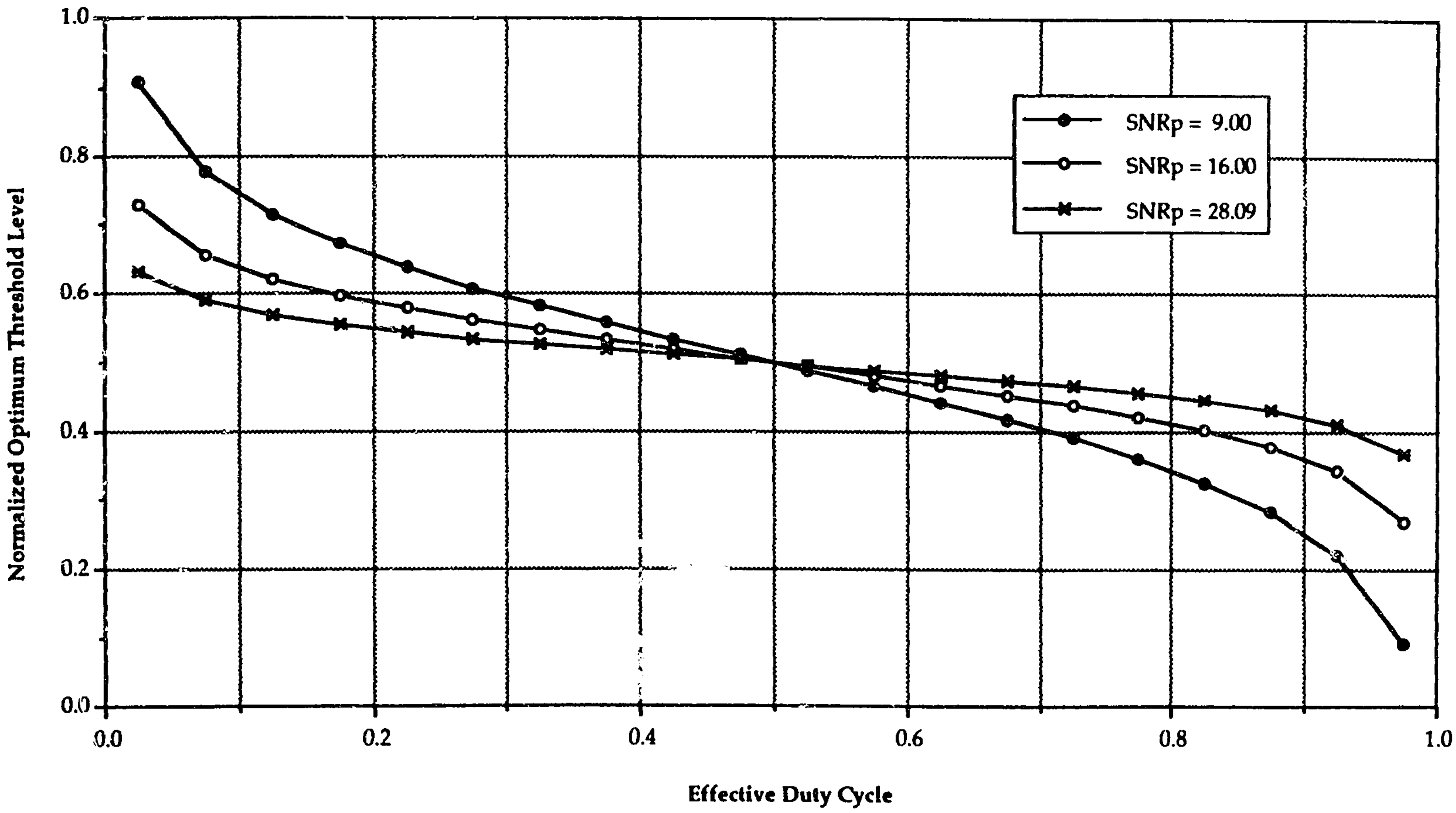


Figure 2.9 Normalized optimum threshold level versus the effective duty cycle of a code.

Table 2.4 Examples of the normalized optimum threshold levels for the 3B16P1G PIM and NRZ binary code.

Code		NRZ	3B16P1G PIM	
Sequence		PRBS	1111...	0000...
Effective Duty Cycle		0.5	0.1111	0.0455
Effective Error Multiplication Factor		1.0	3.0	2.444
Normalized Optimum Threshold Level	SNRp = 9.0	0.5	0.7310	0.8383
	SNRp = 16.0	0.5	0.6300	0.6903
	SNRp = 28.09	0.5	0.5740	0.6084

from the normalized threshold level of 0.5.

2.6 Power Gain

One of the advantages that the PIM code has over the NRZ binary code is its ability to increase the effective transmitted power while keeping the average transmitted power constant. This is due to the decrease in the duty cycle, which allows an increase in the peak power of the transmitted pulses. Thus, the 3B16P1G PIM code, with a duty cycle of 1/9, is able to couple 4.5 times more peak power into the channel than the NRZ binary code, with a duty cycle of 0.5, for the same average power.

In the case of the PIM code, this decrease in duty cycle corresponds to a proportionate decrease in its pulse width. The 3B16P1G PIM pulse width is one-third of that for an encoded NRZ binary pulse. From the analysis done in Section 1.4.2.1, this decrease in the pulse width entails a noise power penalty due to the increase required in the receiver bandwidth. This power penalty is equal to the ratio of the NRZ to the PIM pulse width, which in the case of the 3B16P1G PIM code is equal to three. This, of course, is only valid if the power spectral density curves of the two codes are similar in shape.

In order to compare the two power spectra of these two codes they first have to be derived. It is assumed that the pulses contained in each data stream are rectangular in shape. Then the average power spectral density (Sp_N) for a NRZ binary code which contains rectangular pulses of width "T_P" is given by

$$Sp_N = C_1 \cdot \mathrm{sinc}^2\left(\frac{\omega}{T_P}\right) \qquad 2.21$$

where C_1 is a constant. The average power spectrum for the 3B16P1G PIM code which contains rectangular pulses can be calculated by multiplying the average impulse power spectrum from Figure 2.6 by the power spectrum for a rectangular pulse shape. A rectangular pulse with a width T_P has a power spectral density "S_{P_R}" given by

$$S_{P_R} = C_2 \cdot \mathrm{sinc}^2\left(\frac{\omega}{T_P}\right) \qquad 2.22$$

where C_2 is a constant. The average power spectra of the two codes are graphed in Figure 2.10, where the frequency has been normalized to the pulse width T_P. The magnitude of the NRZ power spectral density has been normalized to its maximum value. Because both of the codes have the same average power, the area under the two power spectral density curves must be equal. Thus, the magnitude of the power spectral density of the 3B16P1G PIM code was adjusted until this condition was achieved. The two power spectra are similar in shape except at the low frequencies where the power spectrum of the 3B16P1G PIM code contains little or no power. Because the similarities between the two curves the noise power penalty of 3.0 should be valid.

The effective power gain of the 3B16P1G PIM code can be calculated by dividing the power gain due to the increase in the peak power by the noise power penalty due to the larger required bandwidth. This gives an effective power gain of about 1.5 which corresponds to the value obtained from Equation 1.21.

In summary, the eight-plus mapping and the approximate RDS coding rules were selected for the 3B16P1G PIM code in order to optimize its implementation into hardware. Also, theoretical values were calculated for the power spectrum, the effective error multiplication factor, the optimum threshold level, the optimum bandwidth, and the effective power gain. These values will be used to compare and confirm the experimental results that are to be obtained for the 3B16P1G PIM code.

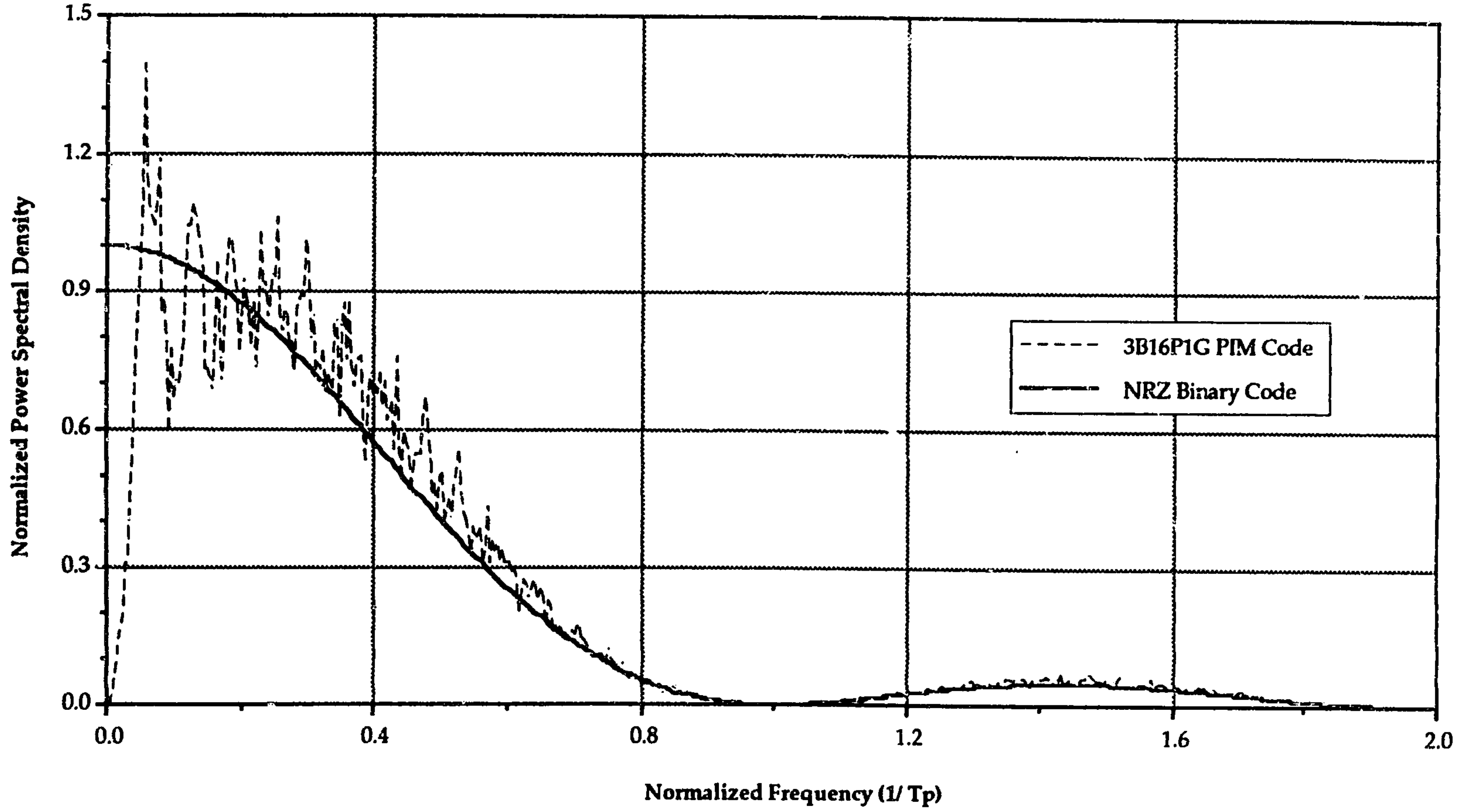


Figure 2.10 A comparison of the average power spectra of the 3B16P1G PIM code and the NRZ code.

CHAPTER 3

EXPERIMENTAL SETUP

The 3B16P1G PIM code using the eight-plus mapping and the approximate RDS coding rules was implemented into an experimental 90 Mbit/s PIM encoder-decoder system which was designed for operation over an electrical transmission link. The components required for the system include the PIM encoder, the channel, the receiver and the PIM decoder. The circuitry contained in the receiver can be divided into four stages: an amplifier, a noise generator, a lowpass filter and a DC level shifter.

The NRZ code is also to be tested in this system, so that its performance can be compared to that obtained for the PIM code. To standardize the characteristics of the signal transmitted through the channel and the sampling characteristics at the output of the receiver, a driver and detector circuit for the NRZ code were fabricated with chips from the same logic family that were used in the construction of the PIM encoder and decoder. To operate at high frequencies, all circuits were designed with matched 50Ω input impedance to prevent reflections between the different components that are contained in the system. The details on the design and construction of these circuits are described below.

3.1 PIM Encoder

The 3B16P1G PIM encoder was built using eighteen commercially available, high speed emitter-coupled-logic (ECL) integrated circuits (ICs)[49]. Due to the high speed operation of the circuit, it was built on a single, double sided 0.031" printed circuit board. Using a single circuit board eliminates the problems associated with time delays and impedance mismatching that occur when signals when signals are transferred between boards, while the double sided characteristic of the circuit board allows the creation of matched impedance transmission lines by controlling the width of a signal trace that is placed over a ground plane[50]. This allows an increase in the circuit's speed by reducing or eliminating reflected signals, assuming proper termination of the signal traces with pull-down resistors.

Since it was important to reduce the time delay between individual ICs, the chips were packed closely together in order to shorten the signal paths. Higher packing densities were achieved by using 0.031" instead of the usual 0.062" circuit board because of the reduction in the width of the signal trace by a factor of two for a 50Ω matched impedance line. Similarly, in order to reduce the board area and to minimize the effect of lead impedance in passive components, surface mount chip resistors and capacitors were extensively used. This allowed the encoder to be built on a 7" by 4.8" printed circuit board where BNC connectors for 50Ω coaxial cable were used to couple signals onto and off the board.

Normally, ECL logic requires two D.C. power supplies. A -5.2 V supply that provides power for the ICs and a -2 V supply that sinks the current from the pull-down resistors that are used to terminate the matched impedance lines. To allow the direct connection of the encoder board to commercial test equipment with 50Ω matched inputs or outputs that are terminated to ground, the power supply voltages for ECL circuitry of the encoder were shifted upward 2 V, thus permitting the pull-down resistors to be terminated to ground. Thus, the encoder still requires two D.C. power supplies but at 2 V and -3.2 V.

The PIM encoder transforms the 90 Mbit/s NRZ binary data sequence into a stream of pulses with varying intervals that occur at an average rate of 30 MSym/s. The width of the generated PIM pulses is comparable in width to the pulses contained in a 270 Mbit/s NRZ binary data stream. The block diagram of the encoder is shown in Figure 3.1 where the number of dashed lines contained inside the arrows indicate the number of parallel data paths. The encoder can be divided into four distinct stages: the clock stage, the input stage, the buffer stage and the output stage.

The 270 MHz clock signal that is supplied to the board is used to drive the output stage and a divide-by-three circuit; the latter acts as a 90 MHz clock for the input stage. In the input stage, the 90 MHz clock drives the logic that converts the 90 Mbit/s NRZ binary data into a 3-bit parallel data stream, which in turn is loaded into the first-in-first-out (FIFO) asynchronous buffer. The data is held in the buffer until required at the output stage. When the PIM interval located in the shift registers of the output stage has been transmitted, the next 3-bits of data stored in the buffer is converted into sixteen position PIM and loaded into the bank of shift registers. At this time the encoding 3-bit binary sequence is purged from the buffer. A short or long interval is encoded, depending on the current value of the RDS, which is dependent on the state of the buffer. The encoded PIM interval is clocked through the shift registers at 270 MHz to create the 30 MSym/s PIM data stream.

A more detailed description of the design and operation for each of the different stages of the encoder is given in the following sections. The components required for the construction of the encoder are listed in Table 3.1, and the definitions of the schematic symbols utilized in the circuit diagrams are presented in Table 3.2.

3.1.1 Clock Stage

The circuitry contained in the clock stage, shown in Figure 3.2, has two functions. First, it buffers the 270 MHz master clock to increase the fanout

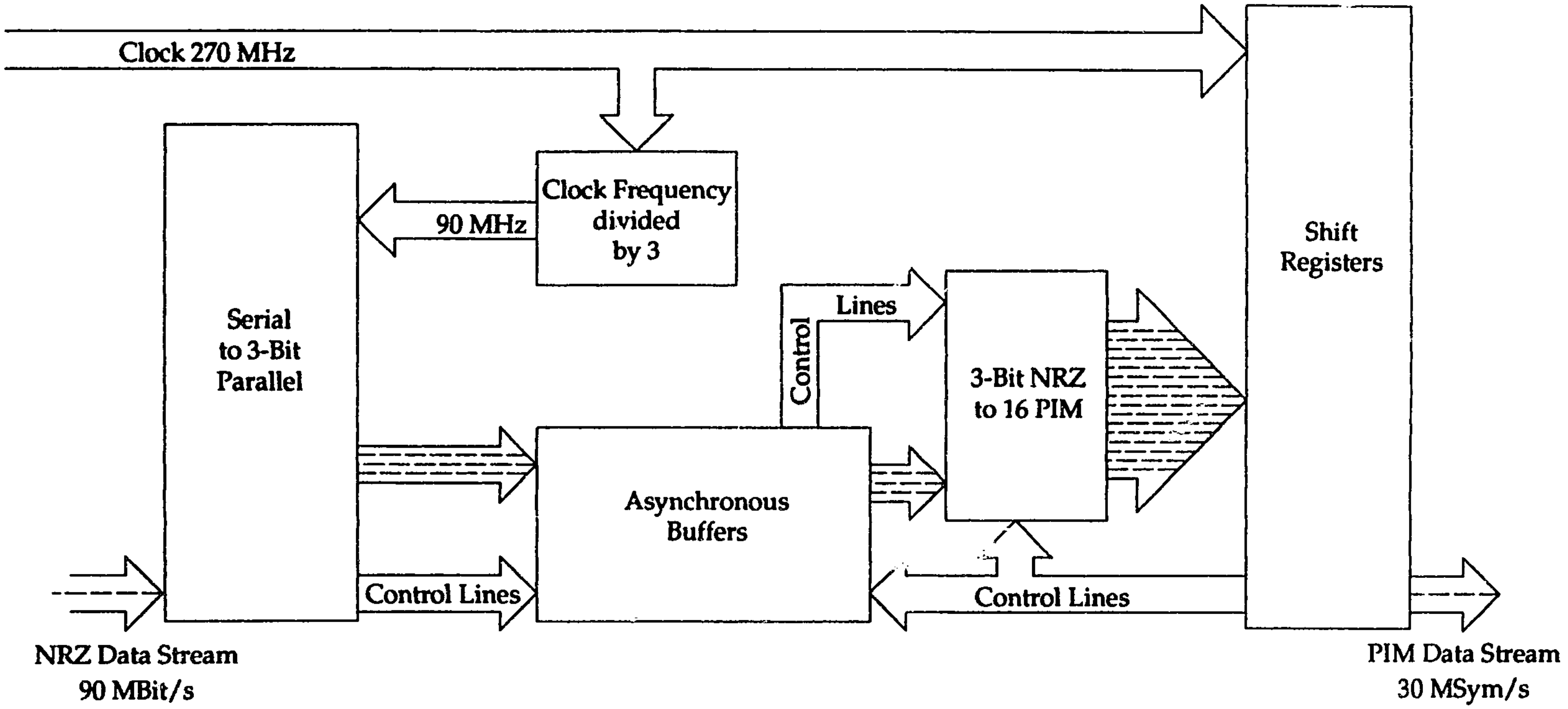


Figure 3.1 Block diagram of the 3B16P1G PIM encoder.

Table 3.1 List of the PIM encoder components.

Location No.	Part No.	Description
IC 1, 12, 13	MC10H209	Dual OR/NOR Gates
IC 2, 3, 4	MC10H141	Shift Registers
IC 5,	MC10H102	Quad NOR Gates
IC 6, 7, 8, 9	MC10H117	Dual OR-AND Gates
IC 10, 11	MC10H162	Binary to 1/8 Converter
IC 14, 15, 16	MC10H186	Hex-D Flip Flop
IC 17, 18	MC10H131	Dual D Flip Flop
C1	0.1μF	Capacitor
D1	1N4148	Diode
R1	68Ω	Resistor
R2	200Ω	Resistor
SW1	MS-242245	Switch

Table 3.2 Definition of the schematic symbols.

⊟	-3.2 Volts
─╫─	Ground
⊕	1.8 Volts (used instead of 2.0 Volts)
─◇─	51Ω termination to ground
─×─	100Ω termination to ground
─•─	Connected signal lines
─▶|─	FMMD914 surface mount diodes
	Switch
	Signal leaving or entering board
	Signal leaving or entering different circuit stages
All ICs	Are decoupled with 0.01 μF capacitors - between 2.0 V and ground - between -3.2 V and ground

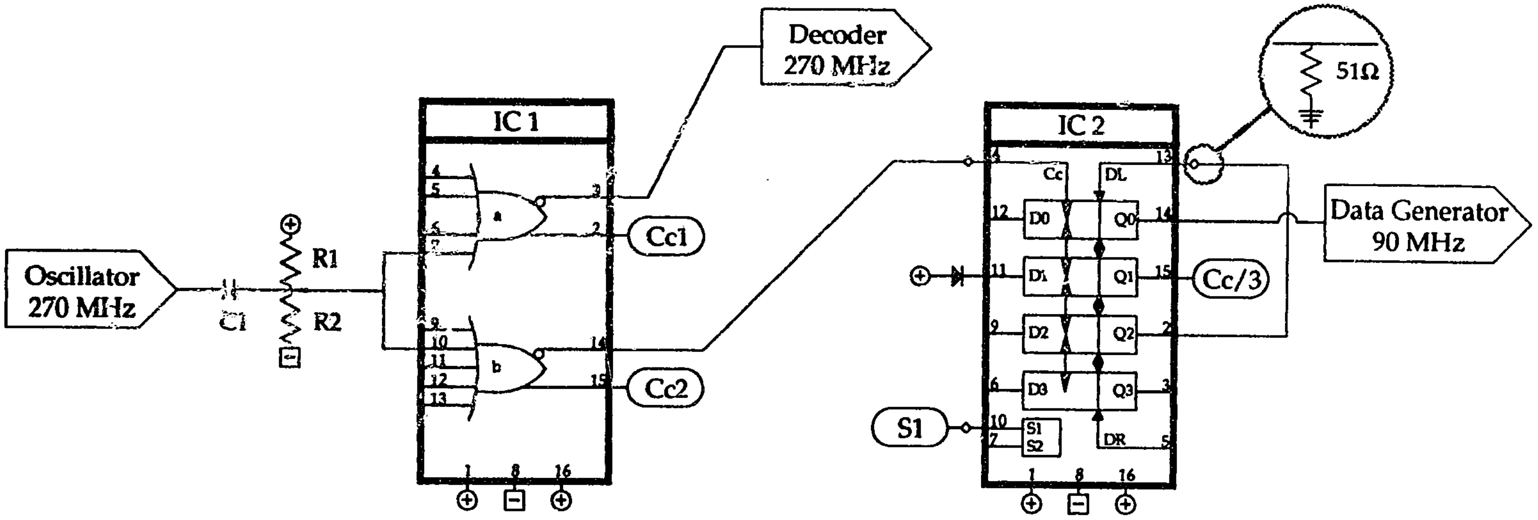


Figure 3.2 The encoder's clock stage circuitry.

capacity of the clock signal. Second, it generates a 90 MHz clock signal by dividing the buffered master clock by three. The source of the 270 MHz master clock signal is a HP 3200B VHF oscillator which is AC coupled to the encoder board. The two input resistors R1 and R2 are used both as a voltage divider to set the DC level of the clock signal to the threshold level for the ECL logic and to provide a 50Ω matched input impedance. The clock signal is buffered by the dual OR/NOR chip IC1 in order to increase its fanout factor to four. The 270 MHz clock signals Cc1 and Cc2 from the output of the OR gates of IC1 are used to drive the high speed output stage. The output of the NOR gate IC1a is sent off the board and is used as the 270 MHz master clock for the decoder.

The other output of NOR gate IC1b is used to clock the shift register IC2 which has been configured as a divide-by-three counter. This was accomplished by feeding back the Q2 output into the "DL" input of the shift register. The parallel inputs were left low (internal 50 KΩ terminating resistors hold the inputs low) except for D1, which was pulled high by a diode tied to the positive power supply. When the encoder is reset, the control signal S1 goes low and a parallel shift occurs on the next clock cycle which loads the sequence 0010 into the registers. Following reset, the S1 control signal goes high, enabling a left shift. Due to the feedback loop and the initial values loaded into the register this produces the binary data stream "0100100100....." which is used as the 90 MHz clock. Its non-ideal duty cycle of 0.33 does not affect the design of the encoder circuitry because all the clocked chips are edge triggered. One of the 90 MHz clock signals generated by IC2 is used to drive the HP 3762A data generator, while the other 90 MHz clock signal Cc/3 is used to drive the low speed input stage.

3.1.2 Input Stage

The circuitry contained in the input stage, shown in Figure 3.3, converts the serial NRZ binary data into a 3-bit parallel NRZ data stream and loads it into the asynchronous buffer. It also contains the circuitry required to reset the encoder. The 90 MHz clock Cc/3 provided from the clock stage of the encoder drives the shift register IC3 which has been set to shift right by pulling pin S2 high. The serial NRZ binary data is supplied to the "DR" input of shift register IC3, by the HP data generator driven by a 90 MHz clock signal from the clock stage of the encoder. The serial NRZ data is shifted right in shift register IC3 and after every third shift the 3-bit data sequence D0, D1, and D2 is loaded into the buffer. This is accomplished by using a divide-by-three counter to supply the timing information needed to distinguish the 3-bit data sequences, and a combination of NOR/OR gates which provide the control pulse required to load the 3-bit data sequence into the buffer. This divide-by-three counter uses the same circuit as the one utilized in the clock stage. Thus, this divide-by-three counter which is clocked at 90 MHz by Cc/3 will produce the data sequence "10010010..." with a repetition frequency of 30 MHz.

For the data to be loaded from the shift register into the buffer, the control line S- must be held high for one gate delay. But since the rise and fall times of the logic are roughly equivalent to the propagation delay of the gate, two gate delays are required to guarantee that the control pulse remains high long enough to trigger the buffer. The control pulse is generated by splitting the signal at the Q3 output of shift register IC4 which has been configured as a divide-by-three circuit. One part of the signal goes to NOR gate IC5a, while the other part of the signal is inverted and delayed by passing it through OR gate IC5d and NOR gate IC5b. The output of NOR gate IC5a will only pull the S- control line high when

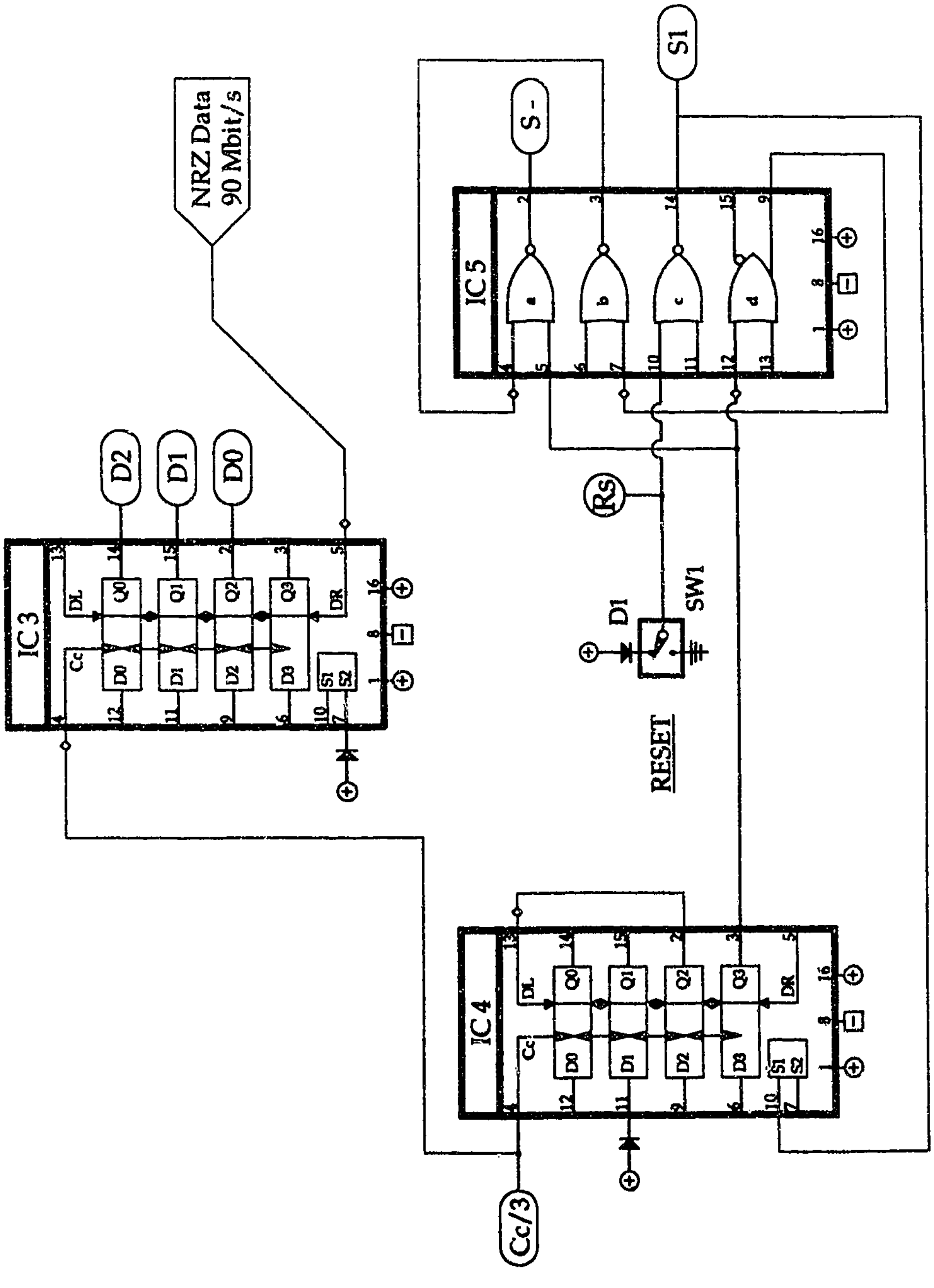


Figure 3.3 The encoder's input stage circuitry.

both its inputs are zero. This condition only occurs when the Q3 output of the shift register goes from high to low. When the Q3 output of the shift register is high, the inputs to NOR gate IC5a are "10" which pulls the S- control line low. When the Q3 output of the shift register goes low, the inputs to IC5a are "00" until the signal from the Q3 output propagates through the two gates. This produces a pulse which is triggered on the falling edge of the Q3 output of the shift register, that pulls the S- control line high for approximately one gate delay.

The reset circuit is used to set the initial conditions of the divide-by-three counters and the shift registers contained in the output stage. It consists of a two position switch "SW1" that either pulls the reset line low by connecting it to ground or high by connecting it through a diode to the positive power supply. In this case the reset signal for the divide-by-three counters has to be inverted. This is accomplished by using NOR gate IC5c to produce the S1 control signal, which is used to select either the parallel or left serial shift mode for the shift registers contained in IC2 and IC4. The reset signal to the shift registers contained in the output stage does not have to be inverted and can be taken directly from the output of switch SW1.

3.1.3 Buffer Stage

The buffer stage is the most complicated stage that is contained in the encoder. It must operate at high speed and be able to provide simultaneous read or write access to the output and input stages. In the worst case, the maximum time that is available to read the data contained in the buffer and to encode it into a PIM interval is equal to the time required to send the shortest PIM interval. The shortest PIM interval which contains only two PIM slots, consists of a PIM pulse followed by a guardtime and occurs for the NRZ binary sequence "000". Since the width of a single PIM slot is equal to the reciprocal of 270 Mbit/s, the maximum time that is available is 7.4 nsec. Since the time required to encode the NRZ data sequence into a PIM interval will take approximately two gate delays or approximately 2.4 nsec. This gives a worst case maximum access time for the buffer of approximately 5 nsec.

To meet these requirements for speed and simultaneous read and writes, certain design strategies were implemented. First, a first-in-first-out flow through buffer was selected to reduce the complexity of the addressing circuitry in order to decrease the access time. Second, asynchronous logic which permits simultaneous read and write access to the buffer along with an increase in the speed of operation was used. Finally, the data was stored in 3-bit sequences, to further increase the buffer speed by permitting parallel processing of data.

The buffer consists of an iterative network that contains a number of identical cells. The purpose of a buffer cell is to either store the data from the previous buffer cell if the next buffer cell is full or to shift the data to the next buffer cell if it is empty. To simplify the design, each buffer cell is divided into two sections: one that keeps track of the status of the buffer cell, empty or full, and the other that stores the data.

The status of a buffer cell is dependent on the status of the next and previous buffer cells. From the state table for the status of the buffer cell given in Table 3.3a, it can be seen that the current buffer cell will only change its status in two situations. One is when the current buffer cell is full but the next buffer cell is empty. Then the data is shifted from the current buffer cell into the next buffer cell and the current buffer cell status is changed from full to empty. The other is when the current buffer cell is empty but the previous buffer cell is full. Then the data is shifted from the previous buffer cell into the current cell and the status of the current buffer cell is changed from empty to full. In order to use the relationships contained in the state table for the design of the buffer cell, the

Table 3.3 State tables for the design of an asynchronous buffer cell.

State of the Buffer Cell	Next State of the Buffer Cell			
S	$S_E^+S_E^-$	$S_E^+S_F^-$	$S_F^+S_F^-$	$S_F^+S_E^-$
S_E	S_E	S_F	S_F	S_E
S_F	S_E	S_E	S_F	S_F

$$\text{Set}\begin{cases}S_E=0\\S_F=1\end{cases}$$

State of the Buffer Cell	Next State of the Buffer Cell				
S	S^+S^-	00	01	11	10
0		0	1	1	0
1		0	0	1	1

a) State Table for the Status of a Buffer Cell

"S" Status of the Buffer Cell
"+" Superscript to indicate the Preceding Buffer Cell
"-" Superscript to indicate the Previous Buffer Cell
"E" Subscript to indicate Empty
"F" Subscript to indicate Full

State of the Data Storage Cell	Next State of the Data Storage Cell			
Z	$S_E Z_0^-$	$S_E Z_1^-$	$S_F Z_1^-$	$S_F Z_0^-$
Z_0	Z_0	Z_1	Z_0	Z_0
Z_1	Z_0	Z_1	Z_1	Z_1

$$\text{Set}\begin{cases}Z_0=0\\Z_1=1\end{cases}$$

State of the Data Storage Cell	Next State of the Data Storage Cell				
Z	$S\,Z^-$	00	01	11	10
0		0	1	1	1
1		0	1	0	0

b) State Table for the Data Storage Cell

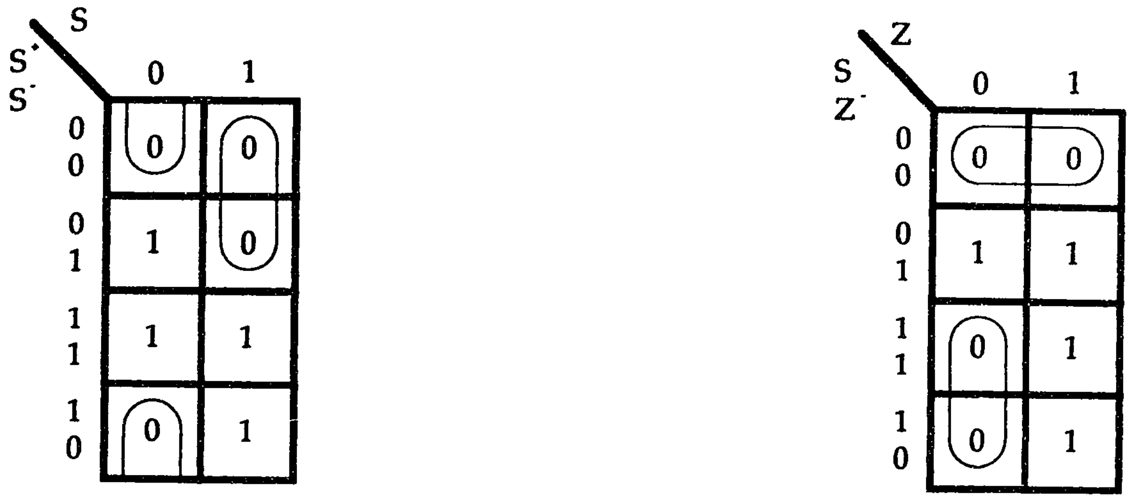
"Z" Data in the Data Storage Cell
"-" Superscript to indicate the Previous Buffer Cell
"0,1" Subscripts to indicate different Data States

states for the status of the buffer cell have to be assigned. In this case the logical "1" was selected to represent the status of a full buffer cell and the logical "0" was selected to represent the status of an empty buffer cell.

The design for the data storage section of the buffer cell requires that it has the capacity to store 3-bits of data. This design process can be simplified by again using the iterative network approach where a data storage cell will be designed that has the capacity to store 1-bit of data. Thus, the amount of data storage required in the buffer cell can be achieved by increasing the number of 1-bit data storage cells it contains.

The data storage cell operates by allowing the data to propagate along the buffer cells until it reaches a buffer cell that is full. At this time the data is volatile and can be overwritten by other data that is applied to the input of the buffer until the status of the buffer cell which contains the data is changed from empty to full. This locks the data into that cell and prevents it from being overwritten. The data contained in a data storage cell is dependent on the status of that buffer cell and on the data contained in the previous data storage cell. From the state table for the data storage cell given in Table 3.3b, it can be seen that if the status of the buffer cell shows that it is full, the data contained in the data storage cell cannot be changed; and if the status of the buffer cell shows that it is empty, the data contained in that data storage cell is overwritten by the data contained the previous data storage cell. In this case the selection of the data states in which to assign the values of a logical "1" and "0" are not critical since the values contained in the next data state of the data storage buffer are independent of this process.

In order to transform the information contained in the state tables on the status of a buffer cell and on the data storage cell into logic equations which can be implemented in hardware, Karnaugh maps, as shown in Figure 3.4, were used



$$\overline{S} = \overline{S}\cdot\overline{S^-} + S\cdot\overline{S^+}$$

$$S = (S+S^-)\cdot(\overline{S}+S^+)$$

a) Karnough Map for the Status of the Buffer

$$\overline{Z} = \overline{S}\cdot\overline{Z^-} + S\cdot\overline{Z}$$

$$Z = (S+Z^-)\cdot(\overline{S}+Z)$$

b) Karnough Map for the Data Storage Cell

"S" Status of Buffer Cell
"Z" Data in Data Storage Cell
"+" Superscript to indicate the Next Buffer Cell
"-" Superscript to indicate the Previous Buffer Cell

Figure 3.4 Karnaugh maps for the design of an asynchronous buffer cell.

to provide a minimum solution. Since the MECL logic family has an IC which contains a 2-level OR-AND network, the buffer cell was designed as a minimum 2-level OR-AND network to take full advantage of the configuration of the IC in order to reduce the chip count and the associated gate delays. This requires the Karnaugh maps to be reduced to a minimum logic expression in the form of the product of sums. Because Karnaugh maps are normally reduced to logic expressions in the form [illegible] the sum of the products, a two-stage process is used. First, the Karnaugh [illegible] reduced to give the minimum inverse logic expression in the form of the sum of products by grouping the zeros of the next state. Then this inverse logic expression is inverted to give the minimum logic expression in the form of a product of sums.

This gives the minimum logic expression for the status of the buffer cell as

$$S = (S+S^{-})\cdot(\overline{S}+S^{+}) \qquad 3.1$$

and similarly the minimum logic expression for the data storage cell as

$$Z = (S+\overline{Z^{-}})\cdot(\overline{S}+Z) \qquad 3.2$$

where "S" refers to the status of the current buffer cell, "Z" refers to the data contained in the buffer storage cell and the superscripts "-" and "+" indicate the previous or the next buffer cell respectively. The circuitry required to realize either Equation 3.1 or 3.2 is a network of two OR gates feeding an AND/NAND gate.

To implement the buffer cell into hardware where each buffer cell will contain one buffer status cell and three data storage cells, would require four OR/AND networks. Since each MC10H117 ECL IC contains two suitable OR/AND networks, the buffer cell can be implemented by using two of these chips.

The number of buffer cells required will depend on the maximum range of values for the RDS. In the case of the approximate coding rules, the RDS will vary over a range of 15 PIM slots or 5 binary bits. Thus, the buffer will require two buffer cells, each with a capacity to store 3 binary bits. The buffer circuit, shown in Figure 3.5, will consist of four MC10H117 ICs. The circuitry that contains the status for the two buffer cells is implemented using the two OR/AND networks IC6ab and IC8ab. The remaining OR/AND networks contained in IC6 and IC8 and the dual OR/AND networks contained in IC7 and IC9 are used to implement the data storage cells required to transfer or store the data bits D0, D1 and D2 respectively.

The data from the input stage is applied to the buffer at the pins labelled D0, D1 and D2 on IC6 and IC7 where it propagates along the buffer, overwriting the data contained in the empty buffer cells. The data is locked into the buffer by a control pulse which is generated in the input stage that pulls the S- control line high on IC6. This S- pulse propagates through the buffer to the last empty buffer cell and sets its status to full. The last buffer cell, which consists of IC8 and IC9, passes the data labelled as Q0, Q1 and Q2 to the output stage. Once the data has been read, the output stage pulls the S+ control line low for approximately one gate delay. This sets the status of the last buffer cell to empty and the data contained in the previous buffer cell is shifted forward. This action is repeated for each full buffer cell, thus updating the data contained in the buffer.

In order for the output stage to encode the correct PIM interval, it has to know whether or not the RDS is positive or negative. This is determined by the status of the first buffer cell, where the status of the first buffer cell "S" and its complementary output "$\overline{\mathrm{S}}$" are relayed to the output stage by the lines labelled $\overline{\mathrm{E1H}}$ and $\overline{\mathrm{E1L}}$ respectively. If the first buffer cell is full, the RDS is positive and the $\overline{\mathrm{E1L}}$ line is pulled low to indicate to the output stage that a short interval is to be encoded. But if the first buffer is empty, the RDS is negative and the $\overline{\mathrm{E1H}}$ line

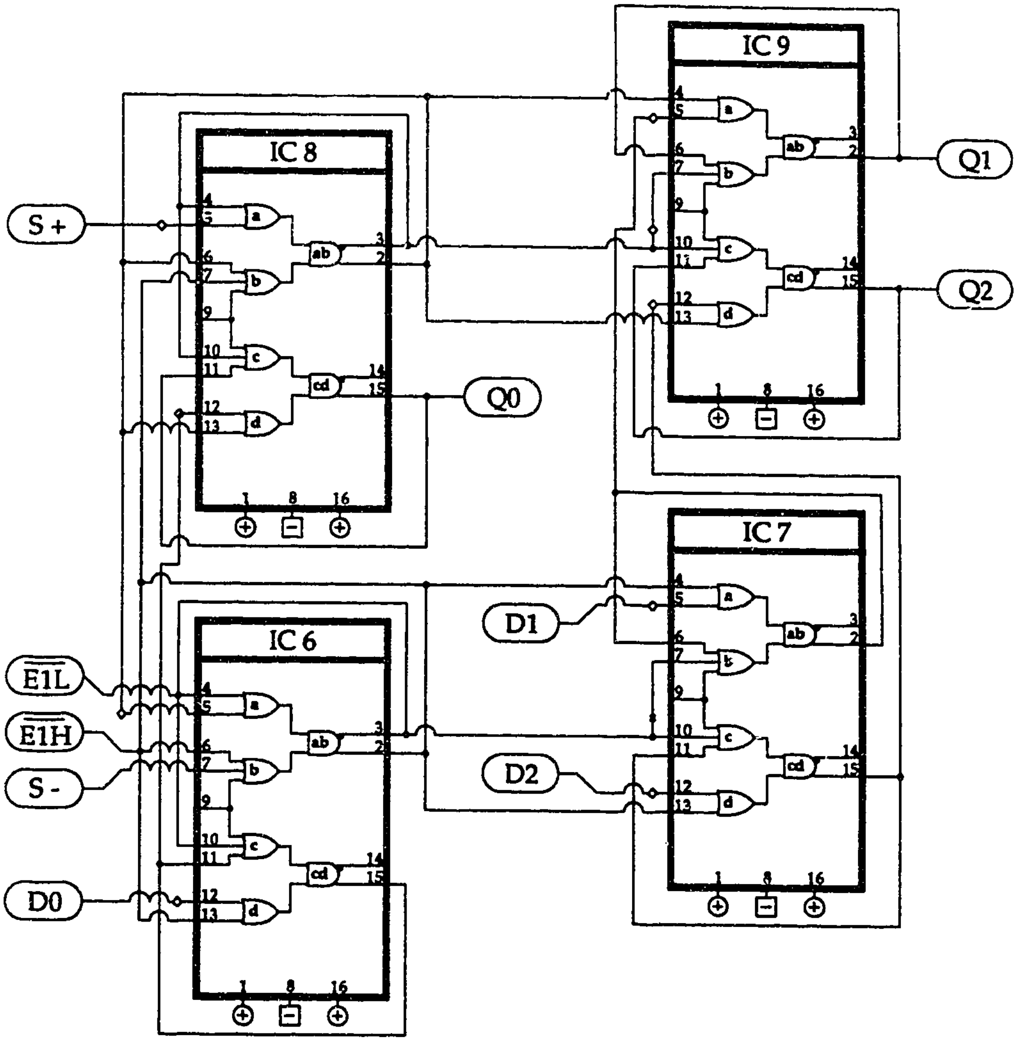


Figure 3.5 The encoder's buffer stage circuitry.

is pulled low to indicate to the output stage that a long interval is to be encoded.

3.1.4 Output Stage

The circuitry contained in the output stage, shown in Figure 3.6, is used to convert the NRZ binary data stored in the buffer into PIM intervals that make up the PIM data stream. It can be divided into three different sections, the PIM interval encoder, the high speed shift register and the buffer control circuitry.

The design for the PIM interval encoder was simplified because of the use of the eight-plus coding rules. This is because the difference in length between the short and the long PIM interval for an encoded 3-bit NRZ data sequence is eight PIM slots. This allows the use of two 3-bit binary to 1-of-8 converter ICs, where their outputs are offset by eight slots to produce either a long or short PIM interval depending on which converter IC was selected. The data Q0, Q1 and Q2 that are stored in the last buffer cell are applied to the inputs of the two 3-bit binary to 1-of-8 converters, IC10 and IC11. The converter IC that is selected will depend on whether the $\overline{\text{E1H}}$ line (which is connected to the enable of IC10) or the $\overline{\text{E1L}}$ line (which is connected to the enable of IC11) is pulled low. The states of the $\overline{\text{E1H}}$ and $\overline{\text{E1L}}$ lines are set in the buffer stage and are determined by the current value of the RDS. The outputs of the two converters are shifted by eight PIM slots with respect to each other and wire-ored (wire-or: a feature of the ECL logic family such that a logic OR gate can be simulated by attaching the outputs of several ECL logic gates to a single ECL logic input) to the high speed shift register constructed from D flip-flops.

The shift register is composed of twenty D flip-flops that are contained in two different types of ICs. One type, the MC10H186 IC, contains six D flip-flops with a common reset. These ICs (14, 15 and 16) contain the majority of the D flip-flops that are required in the shift register, and were used to reduce the

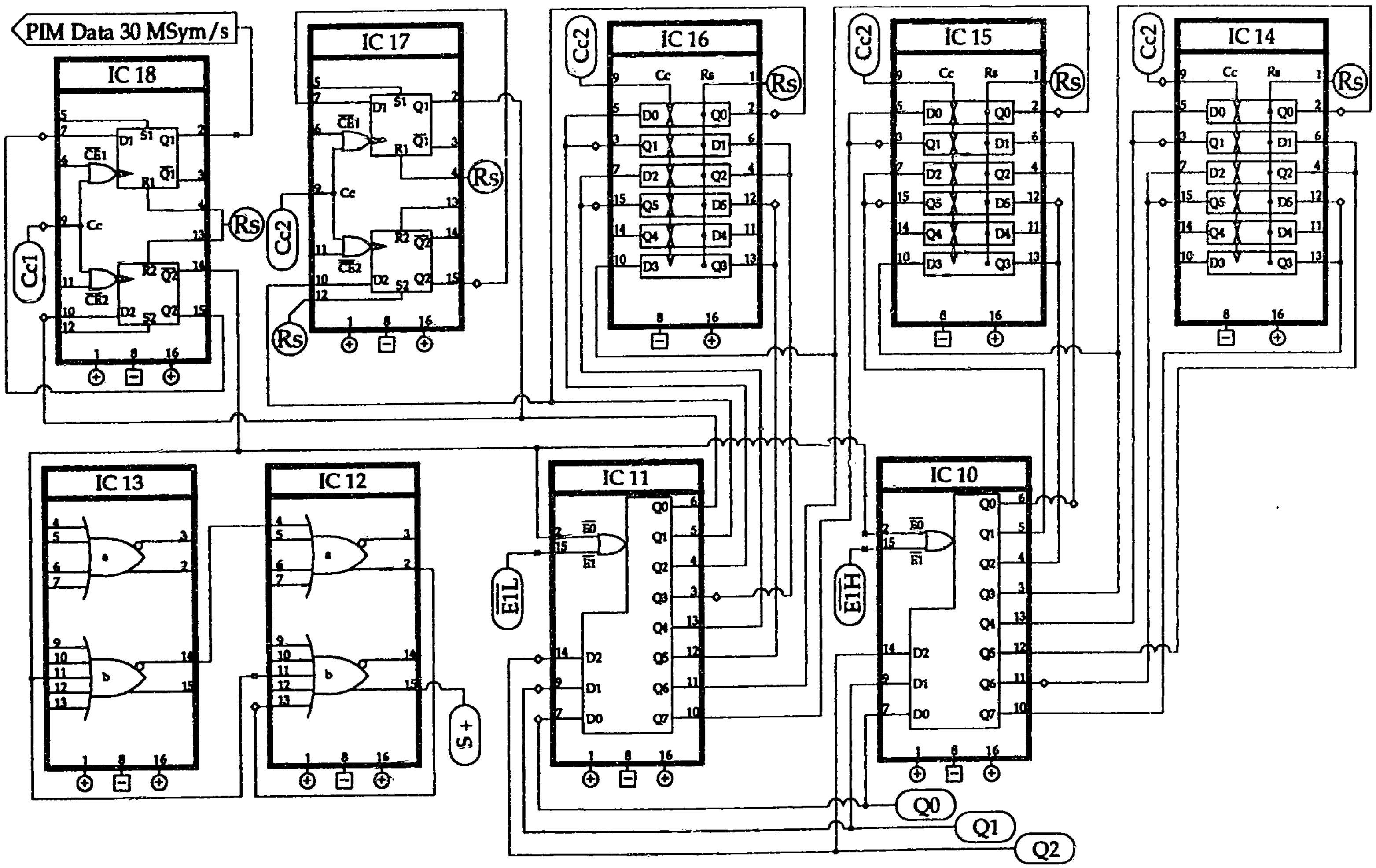


Figure 3.6 The encoder's output stage circuitry.

chip count of the encoder. The other type, the MC10H131 IC, contains dual flip-flops with complementary inverted outputs and set and reset circuitry. These extra features of IC17 and IC18 are used in the generation of the buffer control pulse and to set the initial conditions of the shift register on the reset of the encoder. The D flip-flops that are contained in the shift register are clocked at 270 MHz by the the clock signals Cc1 and Cc2 which are generated in the clock stage. These two buffered clock signals were used to reduce the fanout and to aid in the layout of the circuit.

The last element of the output stage is the buffer control circuitry. It is used to initiate the shift of a new 3-bit data sequence into the last buffer cell after the current 3-bit data sequence has been encoded into a PIM interval by the output stage. This shift in the buffer is triggered by a pulse that pulls the S+ control line low for one gate delay. As in the case of the buffer control circuitry contained in the input stage, two gate delays are required to guarantee that the control pulse remains low long enough to trigger the buffer. The pulse that pulls the S+ control line low is generated by splitting the signal at the complementary $\overline{Q2}$ output of the second D flip-flop contained in IC18. One part of the signal goes to OR gate IC12b, while the other part of the signal is inverted and delayed by passing it through NOR gate IC13b and OR gate IC12a.

The output of OR gate IC12b will only pull the S+ control line low when both its inputs are zero. This condition only occurs when a PIM pulse is shifted into the D flip-flop which causes the complementary $\overline{Q2}$ output to go from high to low. When the complementary $\overline{Q2}$ output is high, the inputs to OR gate IC12b is "10" which pulls the S+ control line high. When the complementary $\overline{Q2}$ output goes low, the inputs to OR gate IC12b are "00" until the signal from the complementary $\overline{Q2}$ output propagates through the other two gates. This produces a control pulse which is triggered on the falling edge of the complementary output $\overline{Q2}$, that pulls the S+ control line low for approximately one gate delay.

On reset of the encoder, the D flip-flops contained in the shift registers are set to zero, except for the second flip-flop of IC17. It is set high and represents the pulse located at the end of the PIM interval. This establishes the required initial conditions for the proper operation of the shift register. After reset, the pulse is shifted along the shift register until it is latched into the the second D flip-flop of IC18, which causes its complementary $\overline{Q2}$ output to go low. This control signal from the complementary $\overline{Q2}$ output serves two functions. First, it drives the circuitry that pulls the S+ control line low to update the data stored in the last buffer cell. Second, it enables the 3-bit binary to 1-of-8 converter that is selected by the buffer. Once the 3-bit binary to 1-of-8 converter is enabled, the 3-bit binary sequence that is applied to its input will cause one of its eight output lines, wire-ored to the input of the shift register, to be pulled high. On the next clock cycle, this pulse is latched into the shift register, creating a new PIM interval, while the previous PIM pulse which enabled the 3-bit binary to 1-of-8 converter is shifted out of the second flip-flop of IC18, disabling it. The cycle repeats itself creating a continuous PIM data stream that is coupled off the encoder board from the Q1 output of IC18. To allow the PIM encoder to drive an AC coupled circuit, the Q1 output is terminated with a 100Ω resistor to create a DC current path for the ECL logic gate.

3.2 PIM Decoder

The 3B16P1G PIM decoder transforms the 30 MSym/s PIM data stream back into a 90 Mbit/s NRZ binary data sequence. The PIM decoder was easier to implement into hardware than the PIM encoder. This is because the value of the RDS is not required to decode the received PIM intervals and since the eight-plus mapping rules were used, only the first 3-bits of the 4-bit PIM interval length is required for the decoding process. Thus, a 3-bit binary counter, clocked at the frequency determined by the width of the PIM slot, could be used to determine the length of the PIM interval and its encoding 3-bit NRZ binary sequence. In the case of a short interval, the counter will count from one to eight slots in the PIM interval and the 3-bit binary value of the counter will represent the encoded 3-bit binary sequence. Similarly, in a long interval the 3-bit counter will count from nine to sixteen slots in the PIM interval. After it has counted the first eight slots which are only relevant to the RDS, the counter will shift from "111" to "000" when the ninth slot is counted. By counting the remaining slots contained in the long interval the 3-bit binary value of the counter will represent the encoded 3-bit binary sequence. If the value of the RDS is required, such as in the situation where a simple form of error detection is to be implemented, a 4-bit counter could be used.

The PIM decoder was built using eighteen commercially available high speed ECL ICs. They were mounted on a 6.6" by 4.4", double sided 0.031" printed circuit board which was slightly smaller than the circuit board required by encoder. Because of the similarities between the encoder and decoder, the same hardware design and construction principles were used in both circuits. These are described in greater detail in Section 3.1.

The block diagram of the PIM decoder is shown in Figure 3.7, where the number of dashed lines contained inside the arrows indicate the number of parallel data paths. The PIM decoder can be divided into four distinct stages: the clock stage, the input stage, the buffer stage and the output stage. The 270 MHz master clock that is supplied from the PIM encoder board to the clock stage of the decoder is buffered and is then used to drive the input stage and a divide-by-three counter which provides a 90 MHz clock to the output stage.

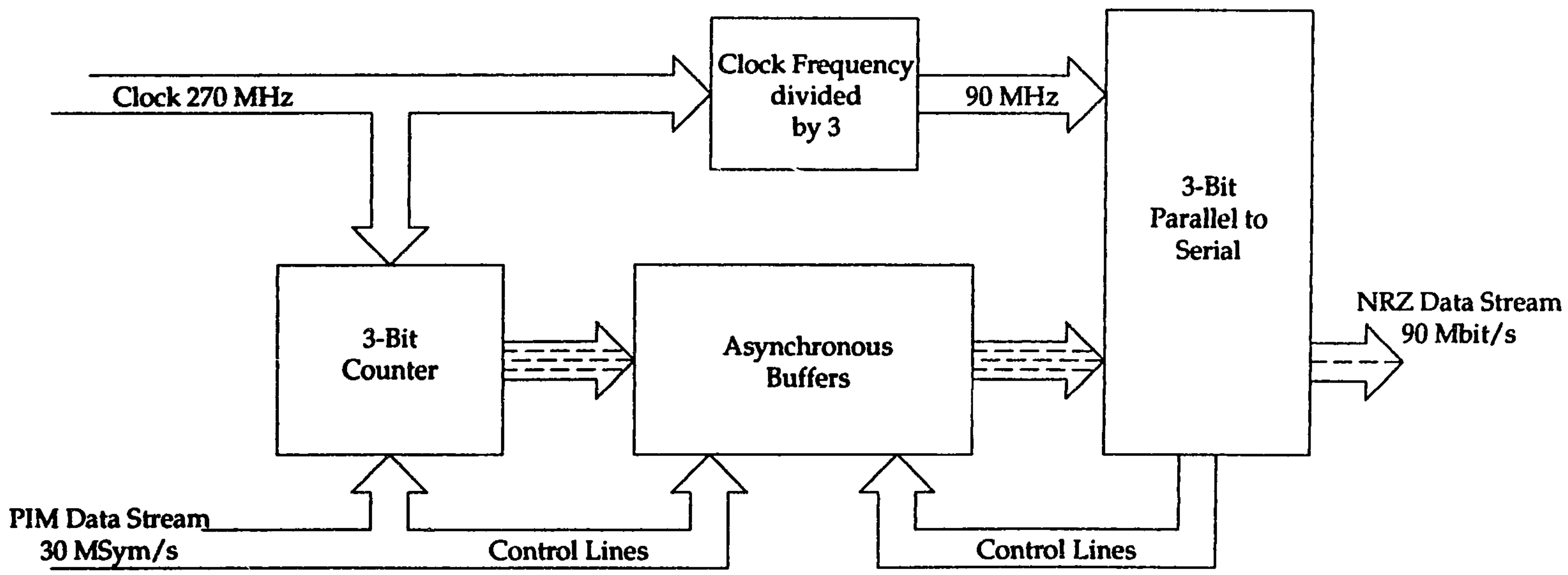


Figure 3.7 Block diagram of the 3B16P1G PIM decoder.

The input stage contains a high-speed, 3-bit binary counter that is clocked at 270 MHz. This is the required counting frequency since the PIM slot is equal to one-third of the width of a pulse from a 90 Mbit/s NRZ data stream. The PIM data is supplied to the decoder at an average rate of 30 MSym/s. When a PIM data pulse is detected at the decoder which represents the end of the PIM interval, it loads the current 3-bit binary value from the counter into the asynchronous buffer. The binary counter is then reset to "000" while the data is held in the buffer until required in the output stage. The output stage which is driven by the 90 MHz clock generated in the clock stage, takes the 3-bit parallel data stored in the buffer and loads it into shift registers. The data is shifted out of the PIM decoder as a continuous 90 Mbit/s NRZ data stream. A more detailed description of the design and the operation of the different stages contained in the PIM decoder is given in the following sections. A list of the components used in the construction of the PIM decoder is given in Table 3.4.

3.2.1 Clock Stage

The decoder clock circuitry, shown in Figure 3.8, is almost identical to the one that is contained in the encoder. It serves the same functions: it buffers the incoming 270 MHz master clock signal to increase the fanout, and it generates a 90 MHz clock to drive the output stage. The 270 MHz master clock signal from the encoder's clock stage is buffered in the decoder clock stage by the dual OR/NOR gates that are contained in IC1. The 270 MHz buffered clock signals Cc and its inverse $\overline{Cc}$ from the outputs of OR/NOR gate IC1a are to be used to drive the input stage. The 270 MHz output from NOR gate IC1b is used to clock shift register IC2 which has been configured as a divide-by-three counter and generates the 90 MHz clock signal Cc/3 that is required to drive the output stage. For a more thorough description of the divide-by-three counter, see Section 3.1.1.

Table 3.4 List of the PIM decoder components.

Location No.	Part No.	Description
IC 1, 7, 8	MC10H209	Dual OR/NOR Gates
IC 2, 15	MC10H141	Shift Registers
IC 3, 4, 5	MC10H131	Dual D Flip Flop
IC 6, 17	MC10H186	Hex-D Flip Flop
IC 9, 10 ,11, 12, 13, 14	MC10H117	Dual OR-AND Gates
IC 16	MC10H104	Quad AND Gates
IC 18	MC10H102	Quad NOR Gates
VC1	4.5/20 pF	Variable Capacitor
D1	1N4148	Diode
SW1	MS-242245	Switch

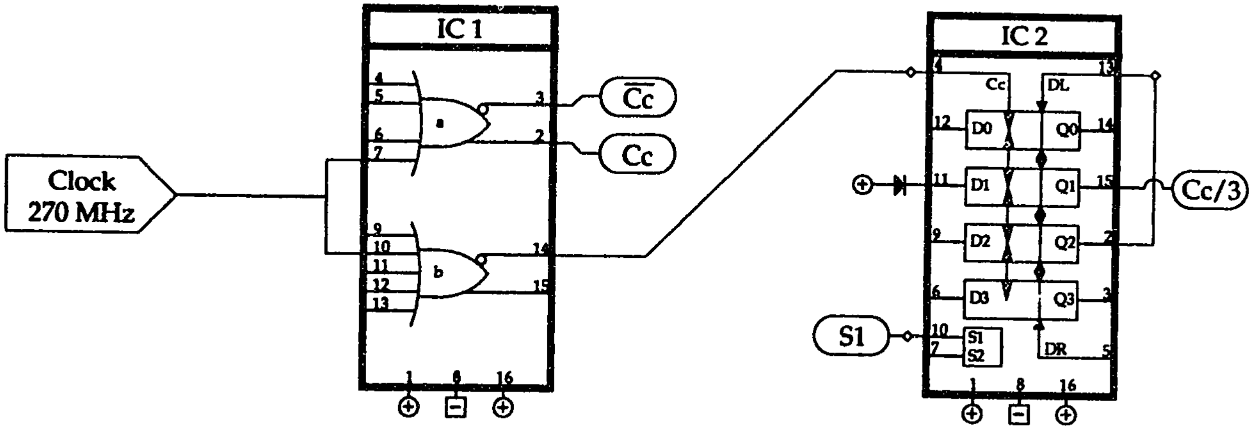


Figure 3.8 The decoder's clock stage circuitry.

3.2.2 Input Stage

The input stage of the decoder, shown in Figure 3.9, converts the received PIM intervals back into the encoded 3-bit NRZ data sequences which are then loaded into the asynchronous buffer. The PIM data stream is coupled onto the decoder board where it is latched into the input buffer which is comprised of the dual D flip-flops contained in IC3. The input buffer which is clocked at 270 MHz by $\overline{Cc}$ from the clock stage, is used as a threshold level detector, a pulse regenerator and to increase the fanout of the received signal. When the pulse that represents the end of the current PIM interval is latched into both of the D flip-flops of IC3, their Q1 and Q2 outputs go high, and three different operations are initiated. The Q1 output prompts the current value of the counter to be loaded into an intermediate buffer which is constructed from three of the six D flip-flops contained in IC6. It also triggers the generation of a pulse that pulls the S- control line low to transfer the previous value of the counter that is stored in the intermediate buffer into the asynchronous buffer. Concurrently, the Q2 output triggers the reset of the binary counter to its initial value of "000".

The counter is constructed from two dual D flip-flops IC4 and IC5 which are driven by the 270 MHz clock signal Cc. The clock that is used to drive the counter is 180° out of phase with the clock $\overline{Cc}$ that is used to latch data into the input buffer. This phase shift between these two stages is required to allow sufficient setup time for the output of the counter to stabilize before it is latched into the intermediate buffer. The divide-by-two circuit is the basic building block of the counter and is constructed from a D flip-flop with its $\overline{Q}$ output fed back into the "D" input.

The first binary bit of the counter consists of a D flip-flop, configured as a divide-by-two circuit, which is clocked at the counting frequency of 270 MHz.

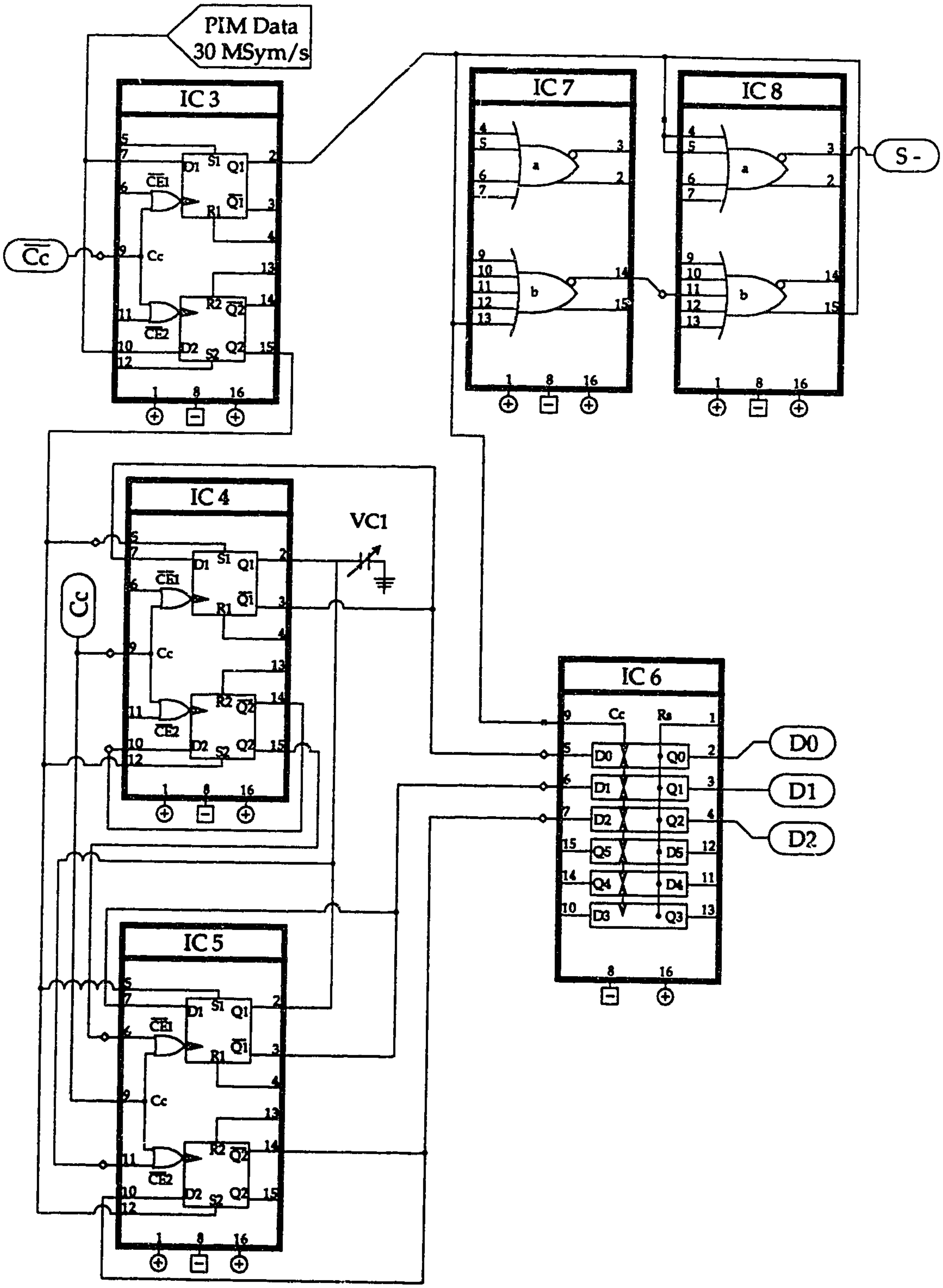


Figure 3.9 The decoder's input stage circuitry.

Both of the D flip-flops of IC4 are configured to provide the first binary bit of the counter, where the $\overline{Q1}$ output is used to supply the first binary bit to the input of the intermediate buffer and the Q1 and Q2 outputs are used to control the dual D flip-flops of IC5 which will provide the second and third binary bit of the counter.

A divide-by-four circuit is required to generate the second bit of the counter. It is constructed from the first D flip-flop of IC5 which is configured as a divide-by-two circuit, that is only enabled when the output of the first binary bit is high. This is accomplished by connecting the Q1 output from IC4 to the $\overline{CE1}$ clock enable of IC5. The $\overline{Q1}$ output of IC5 which provides the second binary bit of the counter is connected to the input of the intermediate buffer.

The third binary bit of the counter is implemented with a divide-by-eight circuit that is configured from the second flip-flop of IC5. To produce a divide-by-eight circuit, clock enable $\overline{CE2}$ must only enable the divide-by-two circuit when the output of the first and second binary bit of the counter is high. This is accomplished by wire-oring the complementary outputs of the first bit (Q1 of IC4) and second bit (Q1 of IC5) of the counter to clock enable $\overline{CE2}$ of IC5. In order to adjust the delay of this enable signal to compensate for the non-ideal characteristics of the ICs, the variable capacitor VC1 was added to the circuit. The $\overline{Q2}$ output of IC5 provides the third binary bit to the input of the intermediate buffer.

The 3-binary bits from the counter are loaded into the intermediate buffer when a PIM pulse is latched into the decoder's input buffer. Once the data is loaded into the intermediate buffer, the counter is reset to "000". Since the output of the counter is supplied by the complementary outputs of the flip-flops, it is reset to zero by pulling the S1 and S2 set lines, of IC4 and IC5 high.

The data that is stored in the intermediate buffer is loaded into the asynchronous buffer when the S- control line is held high for one gate delay by a control pulse that is generated from the two dual NOR/OR ICs, IC7 and IC8. For a more detailed description of the circuitry required to generate this control pulse, see Section 3.1.2.

3.2.3 Buffer Stage

The buffer stage of the decoder, shown in Figure 3.10, is very similar in design to the one in the encoder. It uses the same arrangement of the two MC10H117 ICs to create a buffer cell which is capable of storing or transferring three binary bits. For a more complete description of the design and operation of the buffer cells, see Section 3.1.3. The main difference between the two buffers is that the decoder's buffer contains three instead of two buffer cells, which allows it to store up to a maximum of nine binary bits or the equivalent of 27 PIM slots. This excess buffer capacity decreases the time required for the buffer in the decoder to synchronize or resynchronize to the incoming PIM data stream. As an example, at reset the status and the contents of the buffer cells are undetermined except for the status of the last buffer cell, which is set high (full). The buffer synchronizes to the PIM data stream by either slipping 3-bit sequences if it is full, or duplicating the 3-bit sequences held in the intermediate buffer if it is empty. If the buffer only has the capacity to store 15 PIM slots, this process will eventually adjust the zero RDS point to correspond to the center storage location of the buffer. The time that this takes will depend on the particular PIM data sequences that are received. By increasing the buffer capacity, the location of the zero RDS point is no longer as critical. This increases the possible number of received PIM sequences which can initially the buffer, thus decreasing the time required to resynchronize.

The data from the input stage is applied to the buffer at the pins labelled

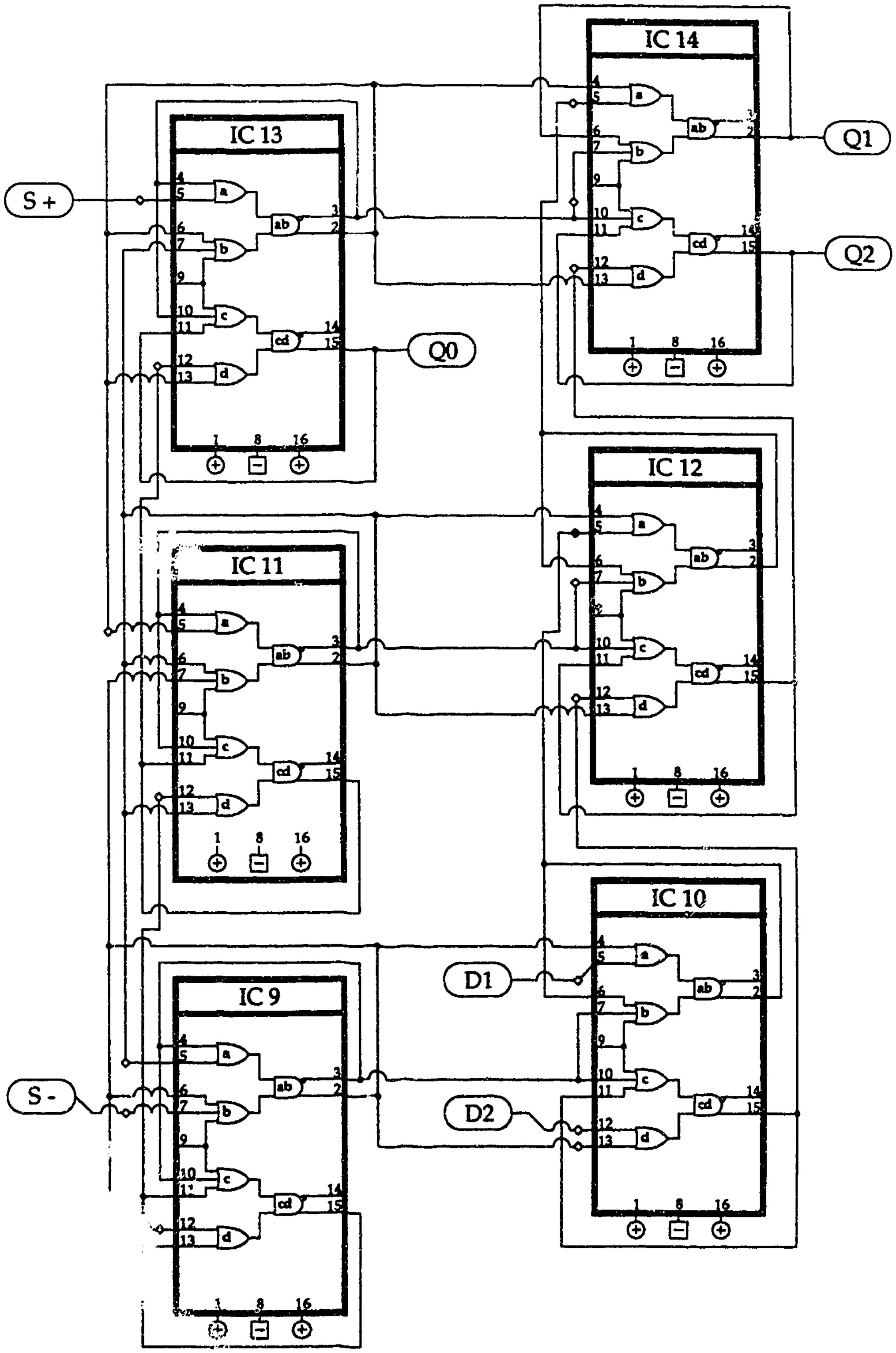


Figure 3.10 The decoder's buffer stage circuitry.

D0, D1 and D2 on IC9 and IC10 where it propagates along the buffer, overwriting the data contained in the empty buffer cells. The data is locked into the buffer by a control pulse, generated in the input stage, that pulls the S- control line high on IC9. This S- pulse propagates through the buffer to the last empty buffer cell and sets its status to full. The last buffer cell, which consists of IC13 and IC14, passes the data labelled as Q0, Q1 and Q2 to the output stage. Once the data has been read, the output stage pulls the S+ control line low for approximately one gate delay. This sets the status of the last buffer cell to empty, which causes the data contained in the previous buffer cell to be shifted forward. This action is repeated for each full buffer cell, thus updating the data contained in the buffer.

3.2.4 Output Stage Circuitry

The output stage transforms the 3-bit parallel NRZ data contained in the asynchronous buffer into a 90 Mbit/s serial NRZ data stream. It also contains the decoder's reset circuitry. As shown in Figure 3.11, the 90 MHz clock that is provided by the input stage, is used to drive the NRZ serial shift register which was constructed from three D flip-flops that are contained in IC17, and a divide-by-three counter which was constructed from shift register IC15. A more detailed description on the implementation of the divide-by-three counter can be found in Section 3.1.2. It creates a 30 MHz repetitive data stream "100100..." where a pulse is generated for every 3-bit binary sequence. This pulse is used to trigger both the transfer of the 3-bit parallel data stored in the buffer cell to the NRZ serial shift register, and the control pulse required to update the buffer.

The data Q1, Q2 and Q3 which is contained in the last buffer cell is applied to the inputs of three AND gates that are contained in IC16. The outputs of AND gates 1C16b, 1C16c and 1C16d are wire-ored to the inputs of the NRZ serial shift register IC17. The AND gates are used to isolate the data that is contained in the

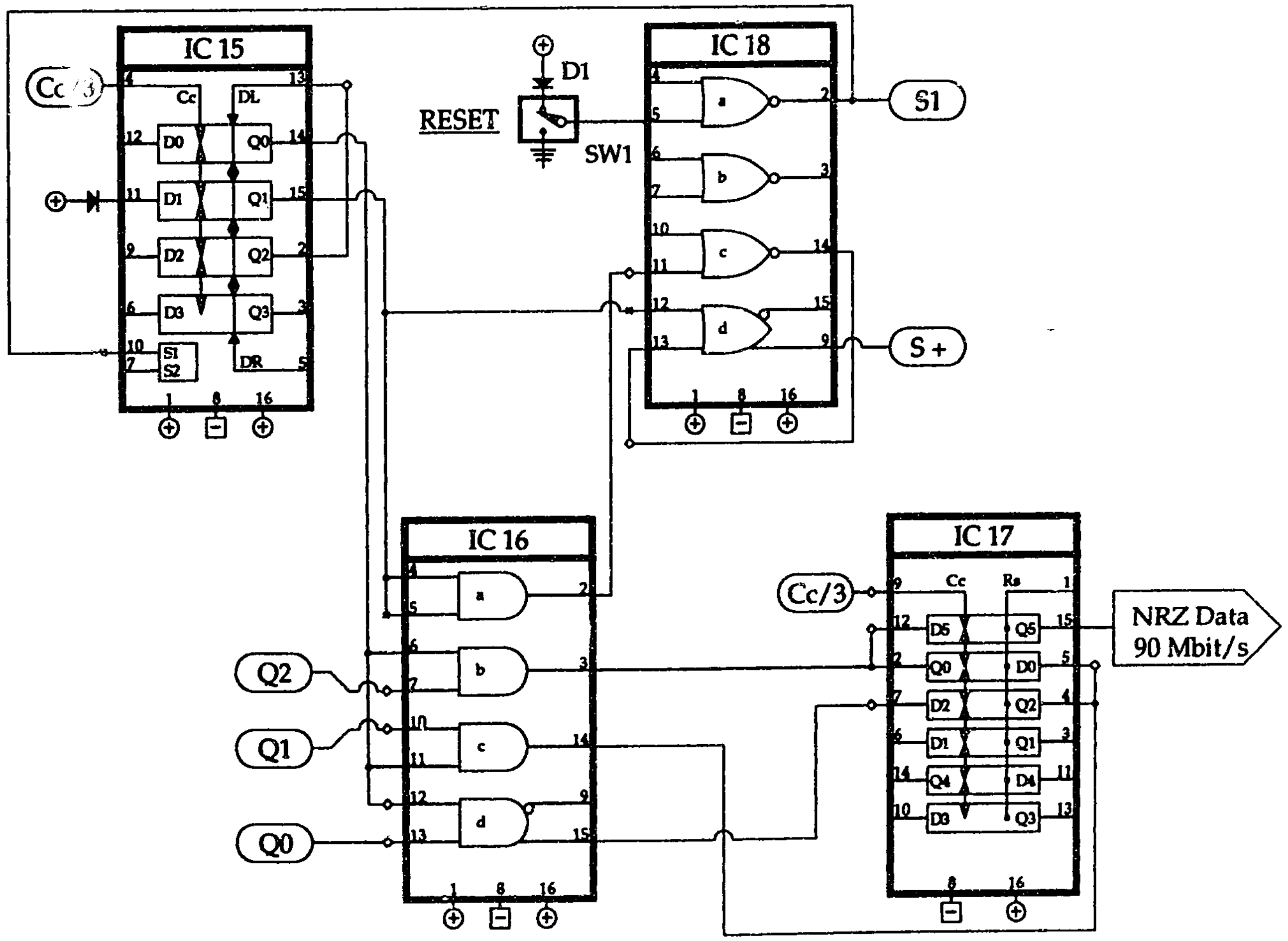


Figure 3.11 The decoder's output stage circuitry.

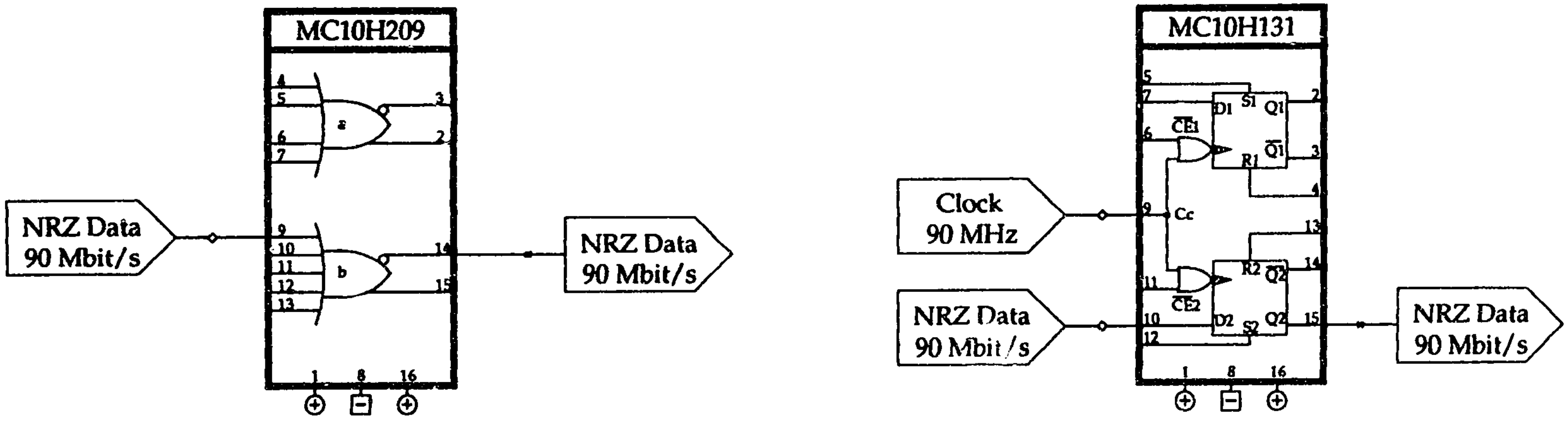
buffer stage from the NRZ serial shift register. When the Q0 output of the divide-by-three counter goes high, the three AND gates are enabled and the data which is stored in the last of the cell buffer is applied to the inputs of the NRZ serial shift register. On the next clock cycle, the data is loaded into the NRZ serial shift register and the pulse contained in the divide-by-three counter is shifted forward. Thus, the Q0 output of the divide-by-three counter goes low which disables the AND gates. Concurrently, the Q1 output of the divide-by-three counter goes high which initiates the shift of a new 3-bit data sequence into the last buffer cell. This shift occurs when the control pulse generated using AND gate IC16a, NOR gate IC18c and OR gate IC18d pulls the S+ control line to the buffer low for one gate delay. A more detailed explanation of the circuitry used for the generation of the S+ control pulse is given in Section 3.1.4.

This process, where the 3-bit binary data contained in the buffer is loaded into the NRZ serial shift register, occurs every three clock cycles to create a continuous 90 MHz NRZ data stream. The NRZ binary data stream is coupled off the decoder board from the Q5 output of the NRZ serial shift register IC17.

3.3 NRZ driver and detector

In order to compare the performance of the 3B16P1G PIM code and the NRZ code for the same communication system, it is important that the characteristics of the transmitted signal and the sampling characteristics of the detector or decoder are similar. Because of this, a driver and a detector circuit were constructed for the NRZ code using the same chips from the ECL logic family that are contained in the PIM encoder and decoder.

The NRZ driver circuit, shown in Figure 3.12a, uses only a single OR gate from a MC10H209 MECL chip. The OR gate is used to buffer the NRZ signal from the data generator and it provides an output signal with similar amplitude



a) The NRZ driver circuit.

b) The NRZ detector circuit.

Figure 3.12 The NRZ driver and detector circuitry.

levels and rise and fall times as those produced by the PIM encoder. To match the input impedance of the OR gate to the data generator, the input is terminated to ground with a 50Ω resistor. Since the receiver is AC coupled, the output of the OR gate is terminated to ground with a 100Ω resistor to create a DC current path.

The NRZ detector circuit, shown in Figure 3.12b, uses a single D flip-flop on a MC10H131 chip. The D flip-flop acts as a threshold detector by sampling the incoming 90 MHz NRZ data stream, once per bit. The clock and data inputs of the D flip-flop are terminated to ground with 50Ω resistors to provide the proper matching impedance, while the output of the D flip-flop is connected to the error detector. Because the same ECL chip is used for the NRZ detector and the input buffer of the PIM decoder, similar sampling characteristics should be achieved.

3.4 Channel

The channel provides a direct electrical link between the output of the PIM encoder or the NRZ driver and the receiver In order to simulate a long transmission path which is attenuation limited, the channel is constructed from a 30 dB fixed attenuator connected together to a 0 to 12 dB variable step attenuator. Because of the quality of these components the bandwidth of the channel spans from DC to greater than 1 GHz. This bandwidth is significantly greater than the effective bandwidth of the receiver, thus it will have little effect on limiting the bandwidth of the system.

3.5 Receiver

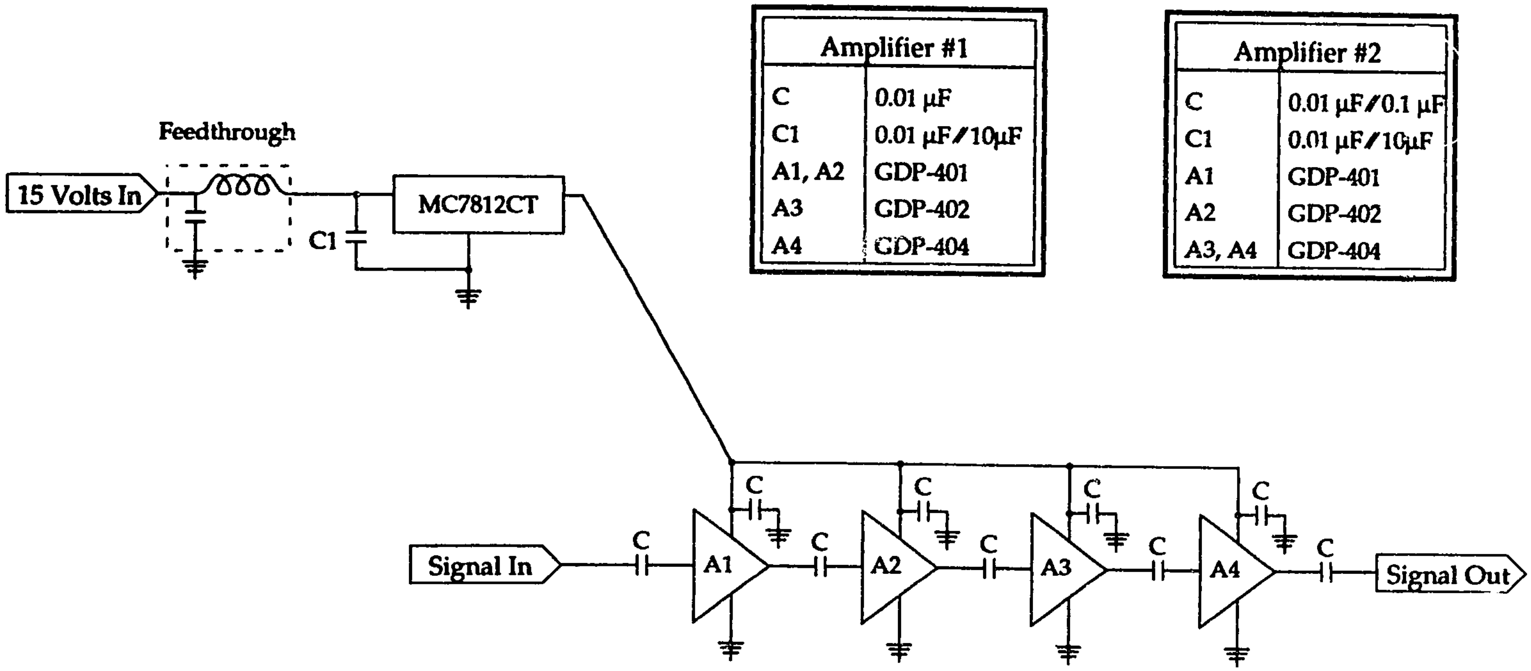
The receiver circuitry can be divided into four sections: an amplifier, a noise generator, a lowpass filter and a DC level shifter. Initially, the signal which was attenuated as it passed through the channel has noise added to it as it enters the receiver. The signal and the additive noise are then amplified and passed through a lowpass filter. The DC level of the signal is then modified by the level shifter before being transmitted to the input of the encoder or the NRZ detector. This design allows the SNR, bandwidth and the effective threshold level of the received signal to be varied, and thus the effect of these parameters on the performance of the 3B16P1G PIM and the NRZ binary code can be examined and compared.

3.5.1 Amplifier

Two amplifiers are required in the receiver. One is used to amplify the received signal and additive noise, and the other is used as a noise generator. The design and construction of the amplifiers are identical except for the values of the by-pass capacitors, which are used to AC couple the signal and the amplifier modules used. The amplifier circuit, shown in Figure 3.13, contains four Avantek fixed gain amplifier modules, cascaded together. All four modules have the same bandwidth, with an upper cut-off frequency which is greater than 400 MHz, and a lower cut-off frequency which is determined by the value of the coupling capacitors. The difference between the various amplifier modules are their maximum power output, small signal power gain and noise figure. To increase the stability of the amplifier circuit, it was enclosed in a die-cast metal case; the external d.c. voltage supply was filtered using a filtercon where it entered the case and was used to drive an onboard 12-volt voltage regulator (MC7812CT).

The measured characteristics of the amplifiers are:

1) Amplifier #1, used as a noise generator, has a passband of 390 KHz to 445 MHz and a gain of 45.3 dB.
2) Amplifier #2, for the received signal, has a passband of 28 KHz to 427 MHz and a gain of 47.9 dB.



Amplifier #1	
C	0.01 µF
C1	0.01 µF/10µF
A1, A2	GDP-401
A3	GDP-402
A4	GDP-404

Amplifier #2	
C	0.01 µF/0.1 µF
C1	0.01 µF/10µF
A1	GDP-401
A2	GDP-402
A3, A4	GDP-404

Figure 3.13 The amplifier circuit.

In amplifier #2, the decrease in the lower cut-off frequency was achieved by increasing the value of the coupling capacitors. This decreases the effect that AC coupling has on some of the NRZ code sequences.

3.5.2 Noise Generator Stage

The noise generator stage is divided into three sections: a noise generator, the variable attenuators and an adder circuit. The noise generator consists of amplifier #1 with a 50Ω resistor at its input. The noise produced at the output of the amplifier is a combination of the thermal noise of the resistor multiplied by the gain of the amplifier plus the noise generated by each amplifier module multiplied by the corresponding gain. Two variable attenuators are used to modify the magnitude of the noise that is added to the signal at the input of the receiver.

The circuit that is used to add the noise to the signal is shown in Figure 3.14. It is a symmetric three-port network which contains the three branch resistors R_X, R_Y and R_Z. For the configuration shown, the input impedance "Zin$_X$" of any port where the other two ports have the load impedances ZL_Y and ZL_Z, can be given as

$$Z_{in_X} = R_X + \frac{(R_Y+Z_{L_Y})(R_Z+Z_{L_Z})}{(R_Y+Z_{L_Y}) + (R_Z+Z_{L_Z})} \qquad 3.3$$

In the case of a matched network where the input impedance is equal to the load impedances, it can be shown that due to the symmetry of the circuit, R_X is equal to R_Y and R_Z. In this case, Equation 3.3 can be simplified and rewritten as

$$Z=3R \qquad 3.4$$

where "R" represents the value of the branch resistors. An adder circuit with an input impedance of 50Ω would require three 16.7Ω branch resistors. Since only 16Ω chip resistors were available, they were used in the construction of the adder

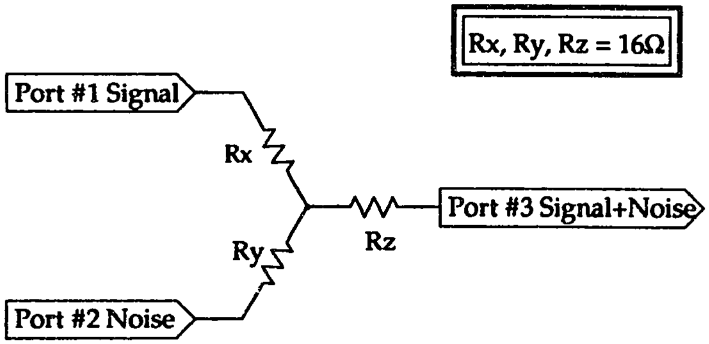


Figure 3.14 The three port adder circuit

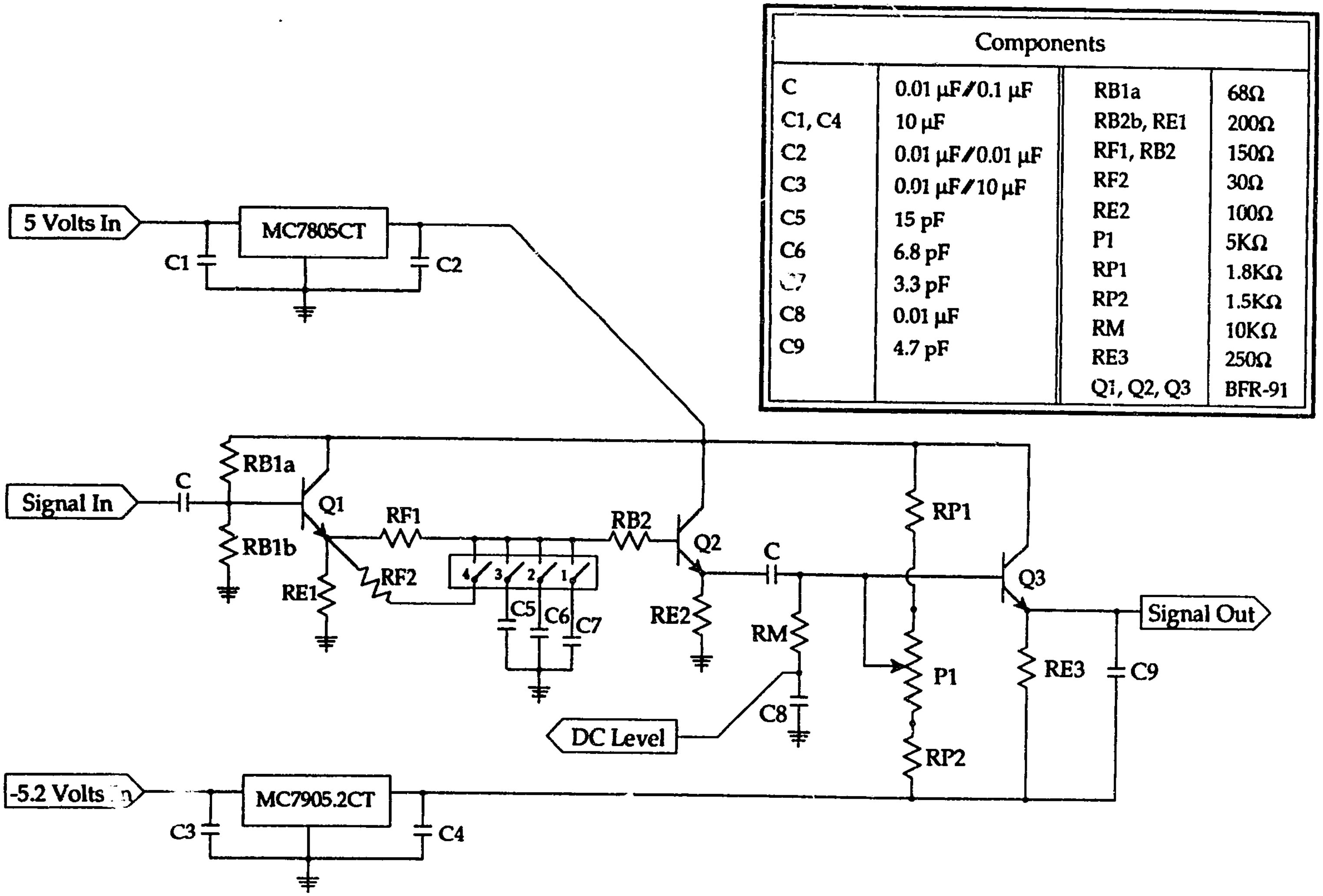
circuit which gives it an input impedance of 49Ω. Also, the signal will be attenuated as it passes through the adder circuit because the matching resistor networks act as a voltage divider. Thus, the signal amplitude will be attenuated to 51% of its original magnitude or by 5.85 dB

3.5.3 Lowpass Filter and the DC Level Shifter

The lowpass filter and level shifter circuitry was built to fit onto a single circuit board in order to simplify the circuit design and layout. The lowpass filter was designed to vary the bandwidth of the receiver over the range of frequencies which would be essential to the transmission of both the NRZ and PIM codes. Likewise, the level shifter was designed to vary the output DC level of the receiver which in turn would determine the percentage of the NRZ or the PIM waveform that would cross the fixed input threshold level of the NRZ detector or the PIM decoder.

The lowpass filter and DC level shifter circuit, shown in Figure 3.15, was built with BFR-91 transistors which have an unity gain bandwidth of 5 GHz, and with chip resistors and capacitors, because of the high frequencies associated with the PIM waveform. To simplify the setup and to provide onboard voltage regulation to the circuit, two voltage regulators MC7905.2CT and MC7805CT were used to supply the +5 and -5.2 voltages that were required. The lowpass filter and DC level shifter circuit can be subdivided into four sections: the input, the lowpass filter, the DC level shifter and the output.

The input section is AC coupled with an input impedance of 50Ω. The AC coupling was achieved with the two capacitors, 0.01μF and 0.1μF which were connected in parallel, to give a lower cut-off frequency of 28.9 KHz for an input impedance of 50Ω. The base resistors R_{B1a} and R_{B1b} serve two functions. As a voltage divider, they determine the DC operating point for the input transistor



Components			
C	0.01 µF/0.1 µF	RB1a	68Ω
C1, C4	10 µF	RB2b, RE1	200Ω
C2	0.01 µF/0.01 µF	RF1, RB2	150Ω
C3	0.01 µF/10 µF	RF2	30Ω
C5	15 pF	RE2	100Ω
C6	6.8 pF	P1	5KΩ
C7	3.3 pF	RP1	1.8KΩ
C8	0.01 µF	RP2	1.5KΩ
C9	4.7 pF	RM	10KΩ
		RE3	250Ω
		Q1, Q2, Q3	BFR-91

Figure 3.15 The lowpass filter and level shifter circuit.

Q1, while at the same time, acting in parallel, they serve to provide an impedance match for the input signal. The values for the base resistors were selected so that a 50Ω input impedance and a DC operating point of approximately 3.75 volts was obtained. The input section is buffered from the lowpass filter by a common collector amplifier consisting of the transistor Q_1 with an emitter resistor R_{E1}.

A first order Butterworth lowpass filter was constructed from a capacitor and resistor placed in series. The frequency response for the lowpass filter is

$$V_{out} = \frac{V_{in}}{\sqrt{1+j\omega/\omega_o}} \qquad 3.5$$

where ω_o is the 3 dB bandwidth of the filter and is equal to the inverse of the RC time constant. In order to vary the bandwidth of the filter, either the value of the capacitor or the resistor had to be modified. In this case a bank of four dip switches were used to add capacitors or a resistor in parallel to the existing components that are contained in the lowpass filter.

In order to span the range of bandwidths which are required for both the NRZ and PIM code, the lowpass filter was designed to operate over two different frequency bands. The fourth dip switch was used to select the frequency band by modifying the value of the resistance of the circuit which causes a large change in the RC time constant of the lowpass filter. The first three dip switches are used to scan the selected frequency band by making smaller changes to the RC time constant of the lowpass filter by modifying the value of the capacitance of the circuit. Up to three capacitors could be added in parallel to the lowpass filter for a possible of 2^3 or eight different values. The output of the lowpass filter is buffered from the level shifter by a common collector amplifier consisting of transistor Q_2 with emitter resistor R_{E2}. Due to instabilities that occurred in the common collector amplifier, base resistor R_{B2} was added to transistor Q_2 to increase its time constant and stability.

The DC level shifter is used to vary the level of the DC voltage applied to the input of the output stage. First, the input of the output stage was DC isolated from the output of the lowpass filter by the AC coupling of the signal between the two stages. Second, the DC level at the input of the output stage is set by varying the DC voltage at the base of transistor Q_3 with the aid of a voltage divider. The voltage divider is connected across the 5 and -5.2 voltage rails, and consists of a twenty-turn potentiometer P_1 with two limiting resistors R_{P1} and R_{P2}. These limiting resistors serve two functions: they decrease the voltage drop across the potentiometer which increases its sensitivity, and they set the minimum output load resistance for the common collector amplifier used to buffer the output of the lowpass filter.

For the components selected, the level shifter will be able to shift the DC level at the input of the output stage between 3.2 volts to -3.4 volts with a sensitivity of 0.307 volts per turn of the potentiometer. In order to measure the DC level at the input of the output stage, a voltmeter was connected to the base of transistor Q_3 via resistor R_M that is in series with capacitor C8. This combination of components acts as a lowpass filter with a 3 dB bandwidth of 1.6 KHz. Because its bandwidth is much smaller than the lower cut-off frequency of the circuit of approximately 28 KHz, it will effectively isolate the meter from the circuit.

The output stage is used to buffer the DC level shifted signal which is to be applied to the input of the NRZ detector or the PIM decoder. The buffer is a common collector amplifier consisting of transistor Q_3 with emitter resistor R_{E3}. For certain DC levels, part of the signal waveform will go negative. This occurs when the peak-to-peak amplitude of the signal is greater than the magnitude of the threshold level, and the signal has been shifted downwards with respect to the threshold level so that the lower end of the signal waveform crosses the zero voltage axis. The ECL logic has a signal amplitude of approximately 1 volt peak to peak and a threshold level that is approximately 0.63 volts. In the case where the signal barely exceeds the ECL threshold level, the output buffer will have to drive the 50Ω input resistance of the PIM decoder or the NRZ detector negative by at least 0.37 volts.

To allow the output buffer to operate in the negative voltage region, the common collector amplifier which contains transistor Q_3, is connected between the 5 volt and -5.2 volt power rails. However, because the 50Ω load of the output buffer is tied to ground, the buffer will be required to sink current if the signal goes negative. Since the buffer's output transistor, Q_3, can only source current at the emitter, emitter resistor R_{E3} must form the current sink. A value for emitter resistor R_{E3} can be calculated if we assume that when transistor Q_3 reaches cut-off, all the current will flow through emitter resistor R_{E3} and the input impedance of 50Ω. This forms a voltage divider where one of the resistances is known. By setting the value of the maximum expected negative voltage for the signal, in this case -1.3 volts which is approximately 3.5 times the expected maximum negative voltage of 0.37, the value of emitter resistor R_{E3} was calculated to be 250Ω. As designed, the level shifter can vary the DC level at the input of the NRZ detector or the PIM decoder from a maximum of 2.5 volts to a minimum of -1.3 volts.

As in the case of the amplifiers, the lowpass filter and level shifter circuit were mounted in a die-cast metal case to provide shielding.

CHAPTER 4

EXPERIMENTAL RESULTS

The components of the experimental communication system which were described in the previous chapter were assembled in order to test the 3B16P1G PIM code. This chapter describes the setup of the experimental system along with the experimental results obtained for the 3B16P1G PIM code. These results are examined and in the appropriate cases compared to the theoretically derived values.

4.1 Experimental Setup

The layout of the experimental system is shown in Figure 4.1. It can be configured to use either the PIM encoder and decoder, or the NRZ driver and detector circuits. In either case, the master clock for the system which is used to drive the circuitry contained in the PIM encoder, is generated by a HP 3200B oscillator running at 270 MHz. The 270 MHz clock signal is buffered on the PIM encoder board and supplies the master clock for the PIM decoder as well as an onboard divide-by-three counter, which is used to create a 90 MHz clock. This 90 MHz clock signal drives the HP 3762A data generator which provides the 90 Mbit/s NRZ binary data stream to either the PIM encoder or the NRZ driver circuit. The encoded or buffered data is passed through the channel to the receiver.

The channel is an electrical link which contains a fixed 40 dB attenuator and a 0 to 12 dB variable step attenuator. The data signal is attenuated as it passes through the channel on its way to the receiver. At the input of the receiver, noise is added to the signal with the use of a 3-port adder circuit. The noise is generated by a 47 dB amplifier with its input connected to a 50Ω resistor.

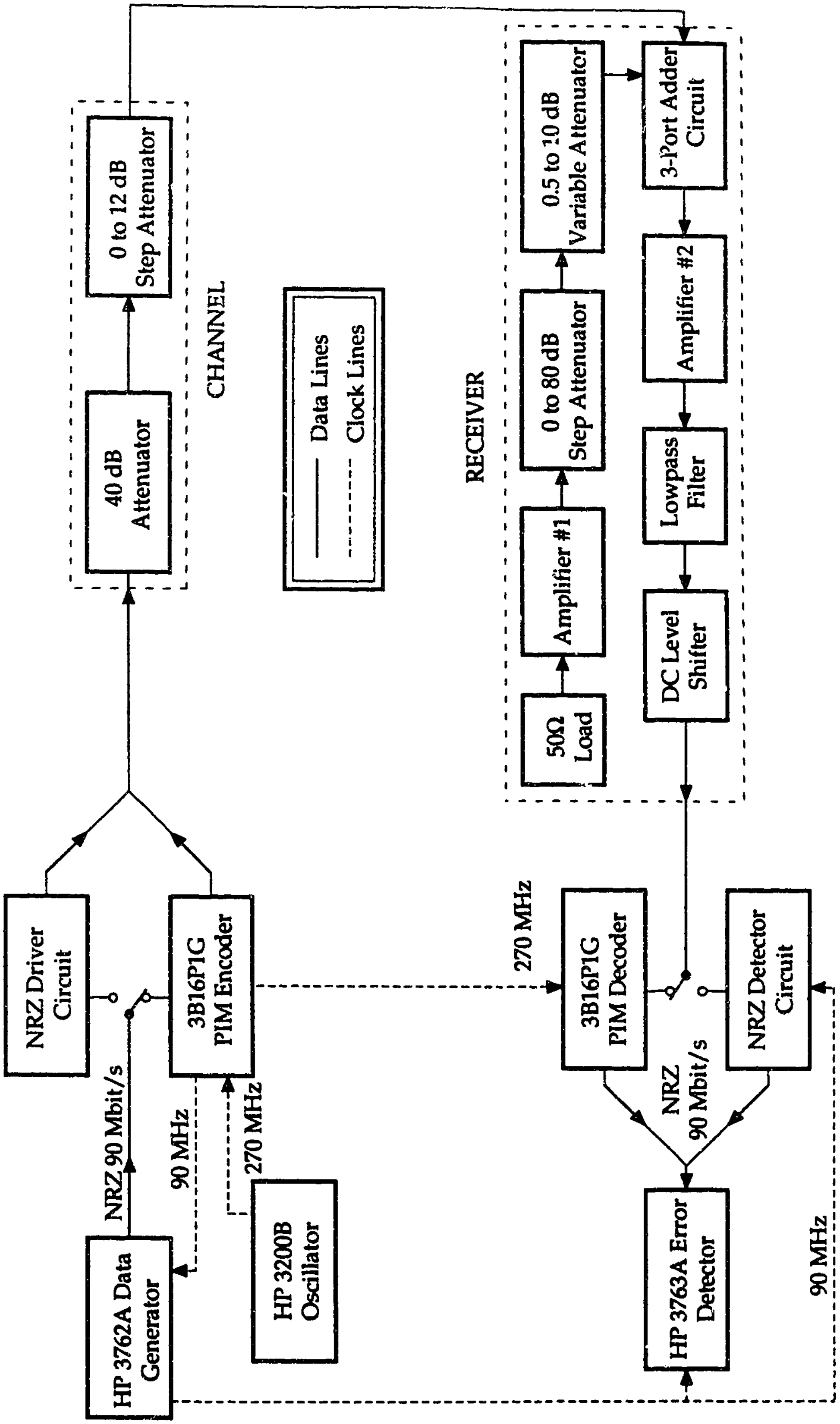


Figure 4.1 The experimental setup.

The amount of noise added to the signal is controlled by two attenuators; a 0 to 80 dB step attenuator with a 10 dB step size, and a 0.5 to 10 dB variable attenuator. This arrangement allows the SNR of the received signal to be altered by varying the amount of noise added to the signal, where the signal level is held constant. The signal with the added noise is then amplified by 45 dB before being passed through a lowpass filter which determines the bandwidth of the system. The DC level of the band-limited signal is then adjusted by the DC level shifter in order to set the threshold level of the signal at the output of the receiver. As mentioned previously, the magnitude of the signal at the output of the receiver remains fixed. The value selected was approximately 0.7 volts which is equal to the voltage swing that occurs at the output of an ECL logic gate. This indicates a combined channel and receiver gain of one which can be achieved by setting the attenuation of the channel equal to the gain of the receiver.

The signal from the output of the receiver enters either the PIM decoder or the NRZ detector, where it is sampled once per clock cycle. The decoded or buffered NRZ data stream is then sent to the HP 3763A error detector, where the BER which is used to gauge the performance of the system, is measured.

In the experimental system, the NRZ driver and detector circuits are used to provide an interface to the channel and receiver that is similar to that of the PIM encoder and decoder. Because of the similar output and sampling characteristics, a more accurate comparison can be made between the performance of the two codes.

4.2 Coder and Decoder Operation

In order to test the operation of the PIM encoder and decoder they were placed in the experimental system, where the channel and receiver circuitry were by-passed by connecting the output of the PIM encoder directly to the input of the PIM decoder. Error-free operation was obtained for an encoded 90 Mbit/s NRZ 2^{23}-1 PRBS. An example of the output waveforms from the PIM encoder and decoder for a 90 Mbit/s data sequence is given in Figure 4.2. The top trace shows the repetitive NRZ binary sequence "1010..." with a pulse width of 11.1 nsec. which is to be encoded by the 3B16P1G PIM encoder. In this case there are only two possible 3-bit binary words, "101" and "010". When encoded into 3B16P1G PIM, each 3-bit binary word has two possible pulse intervals, six or fourteen pulse widths for the binary sequence "101" and three or eleven pulse widths for the binary sequence "010". At this data rate, the PIM pulse is approx. 3.7 nsec. wide which will produce a sequence of PIM intervals of 22.2 or 51.8 nsec. and 11.1 or 40.7 nsec in length. This encoded PIM data stream which appears at the output of the encoder is shown by the middle trace in Figure 4.2. This PIM data stream is sent to the decoder which converts it back into the repetitive NRZ binary sequence "1010..." as shown by the bottom trace in Figure 4.2.

4.3 System Bandwidth

The bandwidth of the system is the result of the combined bandwidths for the channel and the receiver. Because the bandwidth of the receiver is much smaller than that of the channel (as will be shown later), it is the receiver that determines the bandwidth of the system. Thus, the bandwidth of the system can be varied by modifying the bandwidth of the receiver.

The receiver acts as a bandpass filter where its lower 3 dB frequency limit "f_L" is due to the AC coupling of the signal. Its value is fixed and depends on the bypass capacitors and the design of the circuitry that was used in the construction of the receiver. The upper 3 dB frequency limit "f_H" of the bandwidth of the receiver is dependent on the characteristics of its lowpass

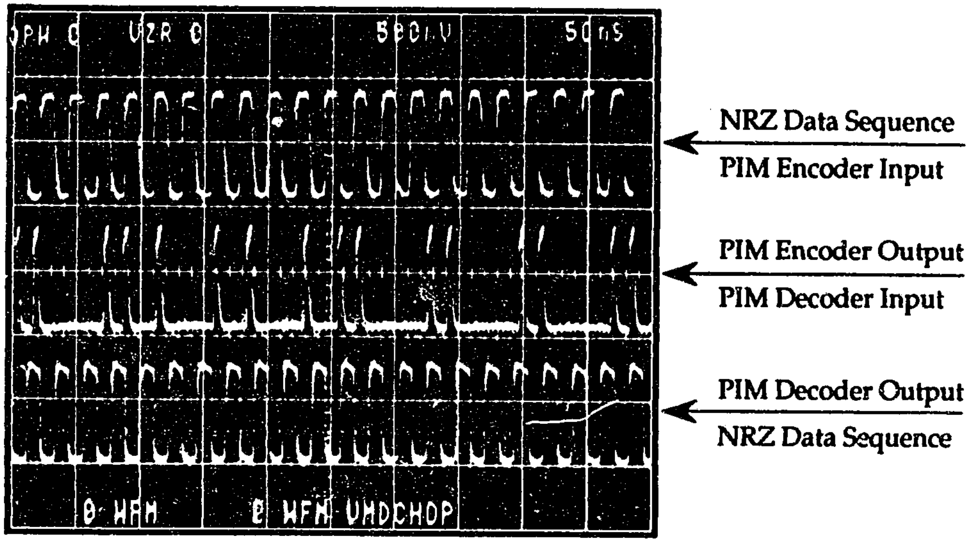


Figure 4.2 The experimental output waveforms of the 3B16P1G PIM encoder and decoder for the encoded NRZ data sequence "1010...". (Vert. = 0.5V/Div., Horiz. = 50 nsec./Div.)

filter. The lowpass filter has been designed so that its bandwidth can be varied by modifying its resistive and/or capacitive values by adding components in parallel to those of the filter by the use of four dip switches (see Figure 3.15). This gives the lowpass filter a possible "2^4" or sixteen different bandwidths.

In order to find the bandwidth of the system for each one of the dip switch settings, it was important to obtain the frequency response of the system (channel and receiver) for at least one setting of the lowpass filter. This was done for the lowpass filter dip switch setting of "L000" (no components were added in parallel) by measuring the response of the system at a series of discrete frequencies. This data was normalized with respect to the maximum gain of the system and is displayed in Figure 4.3. The 3 dB bandpass bandwidth of the system at this dip switch setting, is shown to be 37.5 kHz for "f_L", and 115 MHz for "f_H".

The other fifteen possible bandwidths of the system can now be easily obtained by transmitting AWGN noise with a power spectral density "N_O" through the system and measuring the noise power "N" at the output of the receiver. The relationship between the noise power "N" and the bandwidth of the system is given by

$$N=K \cdot N_O \cdot (f_H - f_L) \qquad 4.1$$

where K is a constant which is defined by the shape of the bandpass frequency response of the system, and "f_H" and "f_L" are its 3 dB upper and lower frequency limits. Since in this case, the upper frequency limit, "f_H", is much larger than the lower frequency limit, "f_L", the bandwidth of the system can be represented by the 3 dB upper frequency limit, "f_H". Thus, Equation 4.1 can be simplified to

$$N=K \cdot N_O \cdot f_H \qquad 4.2$$

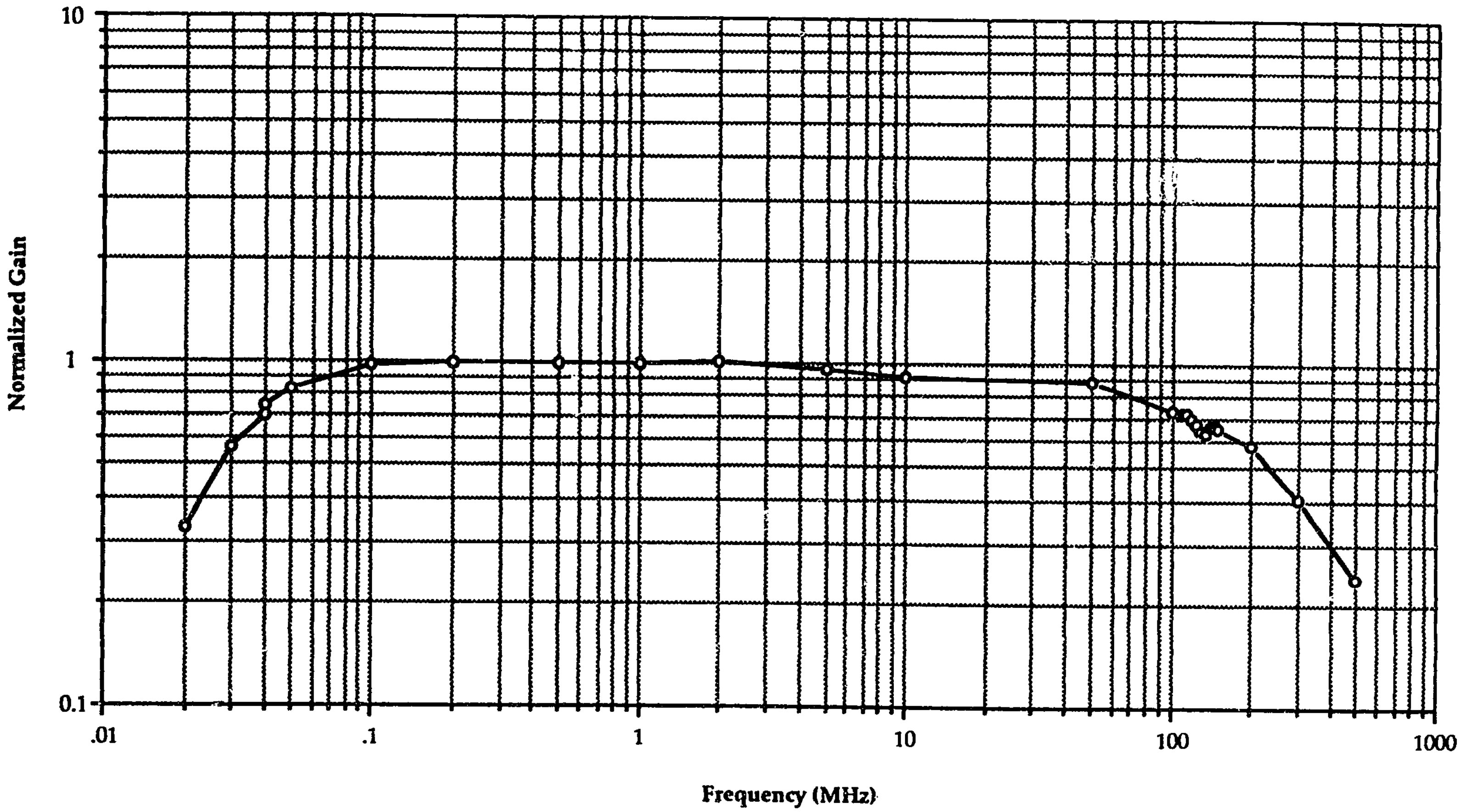


Figure 4.3 The normalized frequency response of the system for the lowpass filter setting of "L000".

By holding the noise power spectral density constant, the bandwidth of the system, "f_H", can be found by comparing the noise power measured at the output of the receiver for an unknown bandwidth to that of the known bandwidth. Using the value "$f_H = 115$ MHz" for the filter switch setting, "L000", Equation 4.2 can be rearranged to give

$$f_{H_{xxxx}} = f_{H_{L000}} \frac{N_{xxxx}}{N_{L000}} = 115 \cdot \frac{N_{xxxx}}{N_{L000}} \text{ MHz} \tag{4.3}$$

Thus, the noise power, "N_{XXXX}", was measured for each switch setting and the bandwidths of the system, "$f_{H_{xxxx}}$", were calculated. The values for the bandwidth of the system at the different switch settings are given in Table 4.1.

4.4 Power Spectrum

The average power spectrum of the 3B16P1G code, shown in Figure 4.4, was experimentally obtained for a 90 Mbit/s encoded NRZ 2^{23}-1 PRBS by averaging one thousand samples taken from the spectrum analyzer with a digital oscilloscope. In order to compare the experimental and the theoretical average power spectra, the theoretical average power spectrum of the 3B16P1G PIM code, which was obtained in Chapter 2 for a rectangular pulse shape, is shown in Figure 4.5 over the frequency range of interest.

In comparing the continuous portion of the experimental and theoretical power spectra, it can be seen that they are similar in shape and contain only a small amount of the total signal power in the DC and low frequency components. This confirms that the PIM code can be AC coupled with minimal baseline wander as compared to the NRZ code.

The only difference between the continuous portion of the two power spectra is that the magnitude of the higher frequency components from the experimental spectrum are attenuated as compared to the theoretical values.

Table 4.1 The bandwidth of the system.

Dip Switch Setting	System Bandwidth	% of Maximum Bandwidth
H000	231.42	100.00
H001	222.90	96.32
H010	194.51	84.05
H011	175.58	75.87
H100	142.92	61.76
H101	133.46	57.67
H110	123.52	53.37
H111	115.95	50.10
L000	115.00	49.69
L001	80.45	34.76
L010	62.94	27.20
L011	52.06	22.49
L100	41.41	17.89
L101	36.16	15.62
L110	32.65	14.11
L111	29.39	12.70

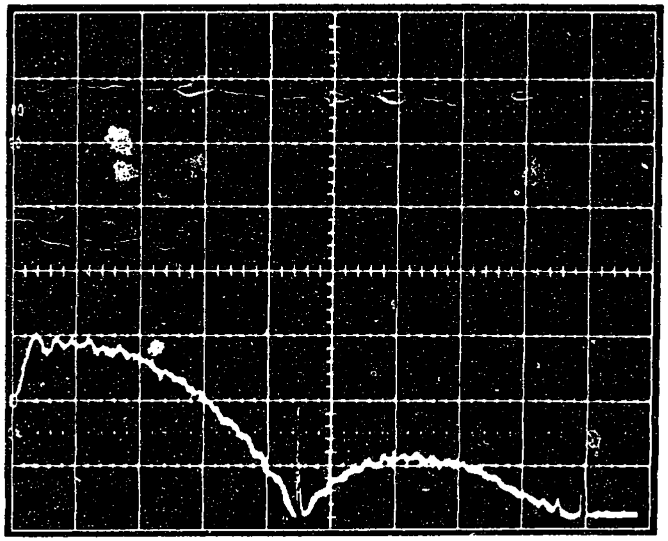


Figure 4.4 The experimental power spectrum of the 3B16P1G PIM code for an encoded NRZ 2^{23} -1 PRBS.
(Vert. = 10dB/Div., Horiz. = 60 MHz/Div.)

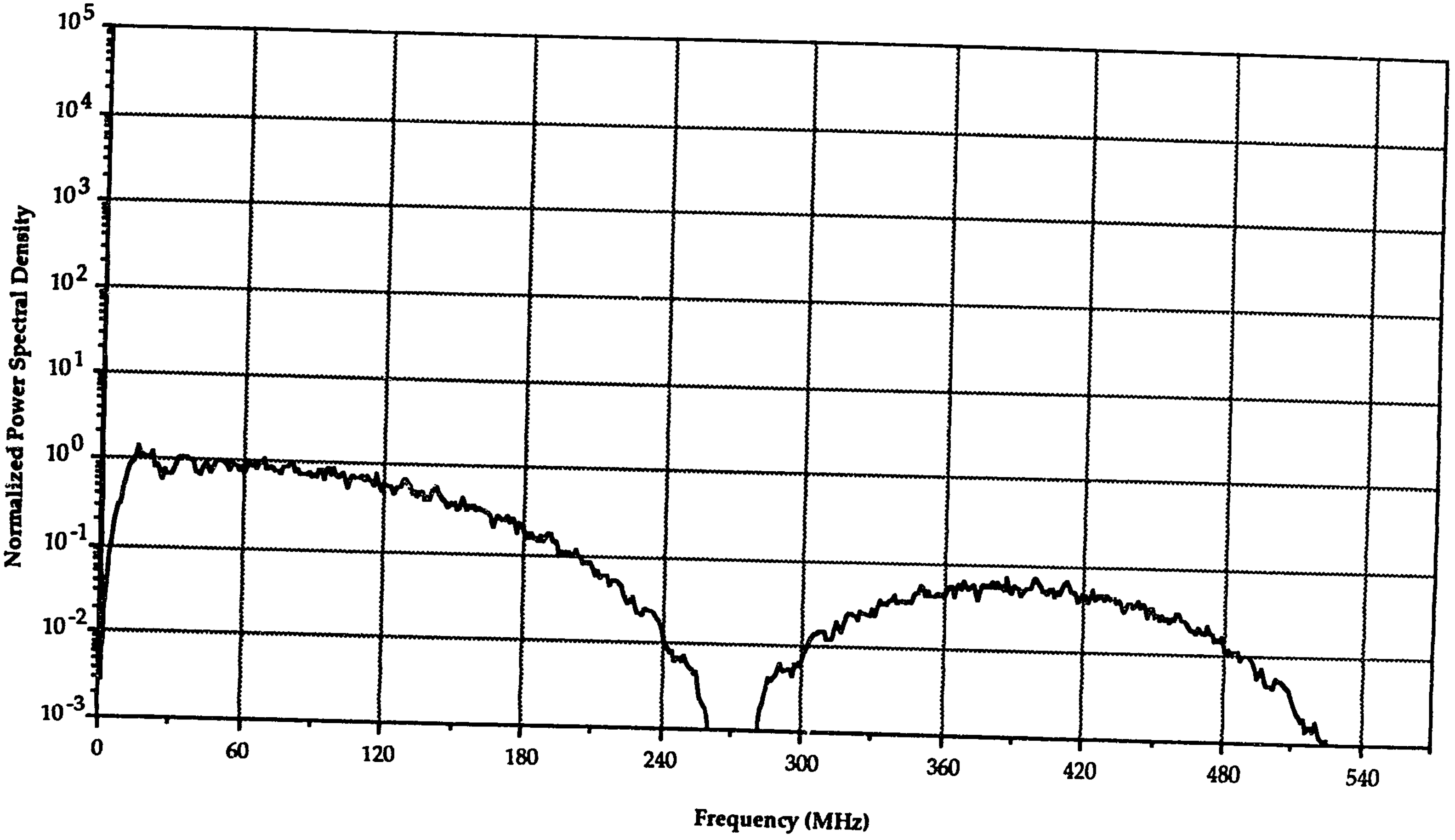


Figure 4.5 The theoretical average power spectrum of the 3B16P1G PIM code.

This can be accounted for by the non-ideal rectangular pulse shape of the experimental PIM data stream. The finite rise and fall times of the pulse are indicative of the attenuation or lack of high frequency components.

Further examination of the two spectra shows that only the experimentally obtained power spectrum has discrete frequency components at multiples of the maximum clock frequency of 270 MHz. These discrete frequency components could be caused either by the non-ideal pulse shape of the PIM data stream or by feedthrough from the clock circuitry; thus, it is unlikely that they could be used for clock recovery. Clock recovery, at the transmission rate (270 MHz), could be obtained by using non-linear processing of the PIM data stream, although accurate clock recovery may not be required. Since only the time between adjacent pulses is needed rather than the absolute time of each pulse, and the interval need only be measured to the nearest whole number, it may be sufficient to use a free running clock that is resynchronized at each pulse.

4.5 Bit Error Rate Measurements

The bit error rate (BER) is used as a measurement of the performance of a code in a given system. In this section, measurements of the BER were taken as a function of the threshold level, the systems bandwidth and the SNR for both the NRZ and the PIM codes. In the case of the threshold level and the systems bandwidth, the BER measurements were used to find their optimum values. Similarly, the BER measurements which were taken as a function of the SNR, were used to calculate the experimental error multiplication factor and the power gain of the PIM code.

4.5.1 Threshold Level

The threshold level is defined as the signal level that is required to trigger a change of state of a logic gate. It can be varied by changing the DC level of the signal at the output of the receiver. The effect of varying the threshold level on the BER was examined for two different ranges of the SNR. In one case, no external noise was added to the signal, which gave SNRs of greater than 35 dB. For the other case, sufficient noise was added to the signal to produce a noise floor for the BER curves at a BER of approximately 10^{-7}; this occurs for SNRs of less than 18 dB. In order to compare the data which was obtained at each of the two SNRs for a specific code sequence, the threshold level was normalized to the maximum eye opening of the data stream with the largest SNR.

The BER was measured as a function of the threshold levels for the NRZ binary sequences 2^{23}-1 PRBS (2^{23}-1 NRZ), 2^{10}-1 PRBS (2^{10}-1 NRZ) and "1010..." ("1010" NRZ), and the PIM encoded 2^{23}-1 PRBS (2^{23}-1 PIM) for the two ranges of the SNR. The results for the NRZ binary sequences were plotted versus the normalized threshold and are shown in Figure 4.6. At large SNRs, the BER for all the NRZ binary sequences, except for the 2^{23}-1 NRZ, decreases quickly with no evidence of a noise floor. The rate of the roll-off of these BER curves varies between the different NRZ data sequences. It appears that the rate of the roll-off decreases as the number of consecutive ones or zeros which are contained in the NRZ data sequence, increases. This can be accounted for by an increase in the baseline wander which occurs due to the AC coupling of the signal, as the number of consecutive zeros or ones are increased in the data stream. This is demonstrated in Figure 4.6 where the 2^{23}-1 NRZ, which contains the greatest number of consecutive zeros or ones, has the largest baseline wander but the slowest roll-off; while the "1010" NRZ, which does not contain any consecutive zeros or ones, has minimal baseline wander but the fastest roll-off of the BER curves.

As the SNR is decreased, the rate of roll-off of the BER curve will become smaller until a noise floor is evident. In the case of the 2^{23}-1 NRZ, the 2^{10}-1 NRZ

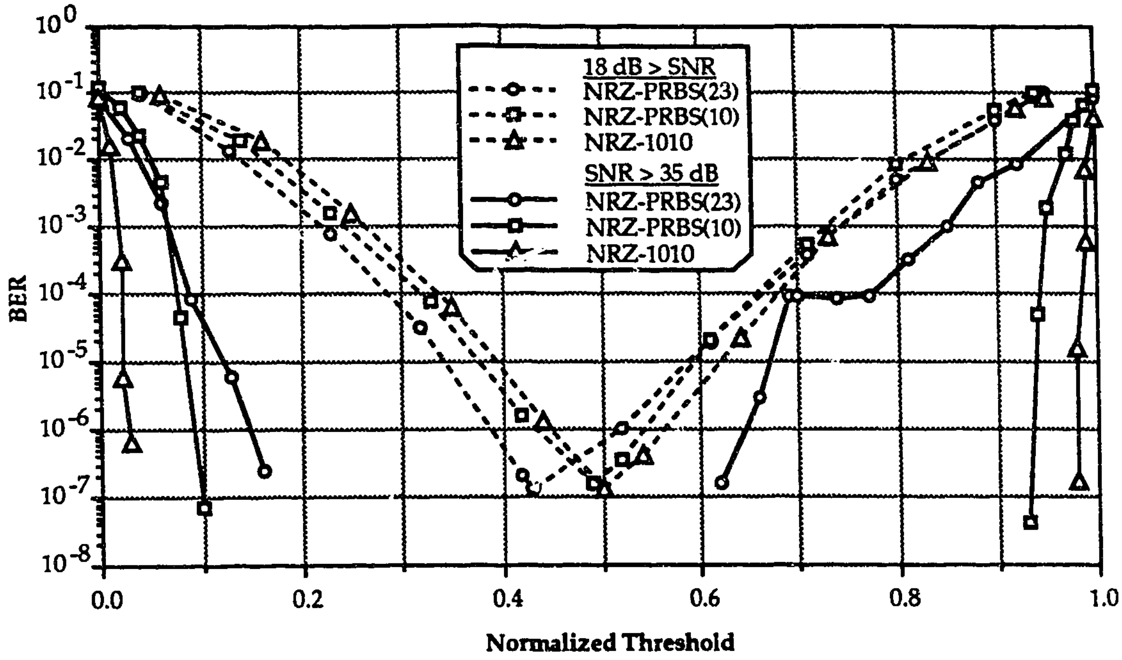


Figure 4.6 The BER versus the normalized threshold level for the NRZ binary code.

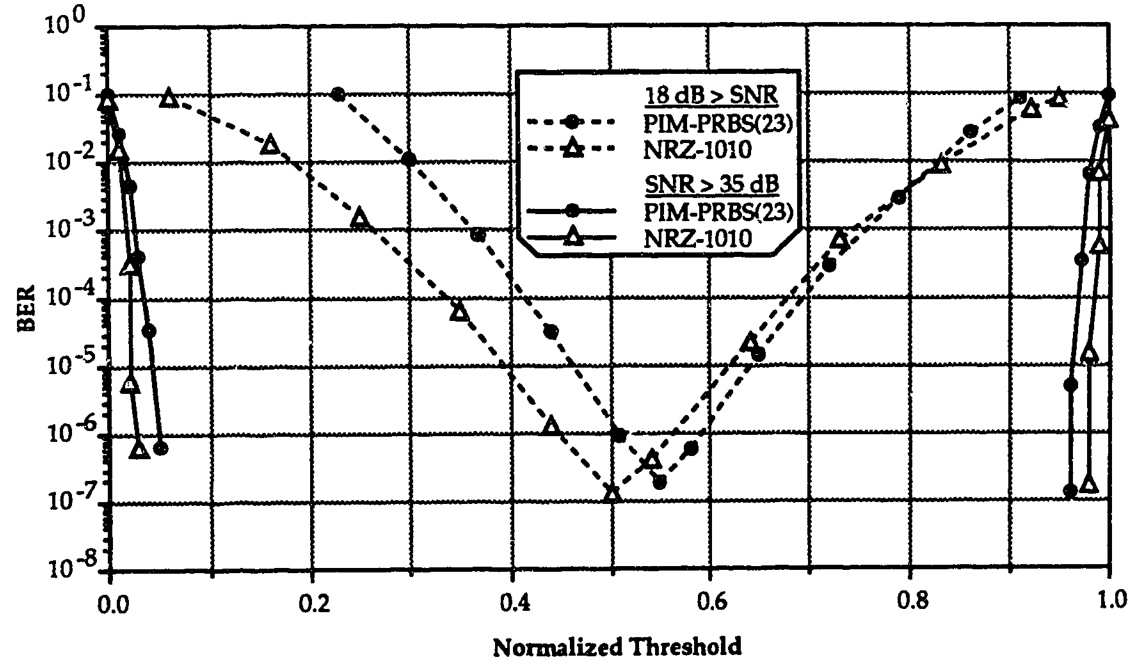


Figure 4.7 A comparision of the normalized threshold levels for the NRZ binary and the 3B16P1G PIM code.

and the "1010" NRZ, noise was added to the signal to decrease the SNR (SNR<18 dB) until the BER curves exhibit a noise floor at a BER of approximately 10^{-7}. The optimum threshold level occurs at this point of minimum BER. Theoretically, for a NRZ binary sequence which [illegible] a duty cycle of 0.5, the optimum threshold level would be located in the [illegible] of the eye opening, equi-distant from both logic levels, thus giving a normalized threshold level of 0.5. All three of the tested NRZ binary sequences have duty cycles of 0.5, but only two of the experimental BER curves, for the 2^{10}-1 NRZ and the "1010" NRZ, give normalized optimum threshold levels close or equal to the theoretical value of 0.5.

In the case of the 2^{23}-1 NRZ, an optimum normalized threshold level of 0.43 was measured. This is lower than the theoretical value and may be caused by the distortion of the signal in the receiver due to the magnitude of the baseline wander that occurs. This assumption is supported by the non-ideal shape of the BER curve for the 2^{23}-1 NRZ at the large SNR. Because of this distortion, the results for the 2^{23}-1 NRZ will not be used for direct comparison or in the calculation of data to be used for comparison. This indicates the difficulties that can arise in transmitting NRZ binary sequences which contain significant numbers of consecutive zeros or ones.

In Figure 4.7, the results for the 2^{23}-1 PIM are compared with those for the "1010" NRZ where the BER has been plotted versus the normalized threshold level. As expected at large SNRs (SNR>35 dB) where no noise was added to the signal, neither the NRZ nor the PIM data sequence shows any indication of a noise floor. The rate of the roll-off of the BER curves are similar, with the 2^{23}-1 PIM having a slightly slower roll-off than the "1010" NRZ.

Noise is added to the signal to decrease the SNR (SNR<18 dB) until the BER curve for the 2^{23}-1 PIM exhibits a noise floor at a BER of approximately 10^{-7}.

From Figure 4.7, it can be seen that the normalized optimum threshold level for the 2^{23}-1 PIM is approximately 0.55 as compared to 0.5 for the "1010" NRZ. In the theoretical analysis of the threshold level for the PIM code (see Section 2.5.1), it was shown that the lower duty cycle of the PIM code would shift the normalized optimum threshold level above 0.5; this prediction is confirmed by the experimental data. The calculation of the normalized optimum threshold level requires the error multiplication factors for the specific PIM code sequence. In the case of the 2^{23}-1 PIM, the error multiplication factors are dependent on the design of the buffer along with other factors, and were not derived.

4.5.2 Bandwidth

An important factor that affects the performance of the system is its bandwidth. A change in the bandwidth of the system will modify the SNR and the pulse shape of signal, causing either an increase or decrease in the BER. This can be better understood by examining the effect that varying the bandwidth has on the noise power, signal power and the pulse shape.

For AWGN noise with a constant power spectral density "N_O", the noise power is proportional to the bandwidth. Thus, as the bandwidth increases, so does the noise power. In the case of the signal, its power spectral density is not constant and varies with frequency. For example, by examining the average power spectrum of the PIM code that is shown in Figure 2.10, it can be seen that a large proportion of the signal power is located in the lower frequencies and that the power spectral density of the signal decreases rapidly with frequency. Thus, for large bandwidths, varying the bandwidth will cause only small fluctuations in the signal power, while at smaller bandwidths, a similar variation in bandwidth will cause major fluctuations in the signal power. Because the signal and noise power vary at different rates as the bandwidth is changed, the SNR will also vary with the bandwidth.

The pulse shape is dependent on the magnitude and phase of the frequency spectrum of the transmitted data stream. As the bandwidth is varied so is its effect on the spectrum and correspondingly the pulse shape. The pulse shape can have major impact on the performance of the system. As an example, a triangular shape pulse which has a large peak power as compared to its average power, would have a larger SNR if sampled at the maximum value than for a rectangular pulse shape containing the same average power. This larger SNR would decrease the BER, which would improve the performance of the system. However, the performance of the system using the triangular pulse would be more sensitive to the location of the decision sampling point. Thus, a change in the bandwidth which would modify the pulse shape could have a major impact on the performance of the system.

The bandwidth effects the SNR and pulse shape of the transmitted data, which in turn affects the performance of the system. Therefore, there exists an optimum bandwidth for each transmitted data sequence which is defined as the bandwidth which produces the smallest BER. The trade-off here is that although the bandwidth must be restricted to reduce the noise power, it must not be too small, otherwise the signal will be adversely effected.

In order to find the optimum bandwidth of the system for the 2^{23}-1 NRZ, the 2^{10}-1 NRZ, the "1010" NRZ, and the 2^{23}-1 PIM, the BER was measured as a function of bandwidth for each of the data sequences. In all cases, the signal and the noise power at the input of the receiver were kept constant while the bandwidth was varied. The resulting BER measurements are shown in Figure 4.8.

The 2^{23}-1 NRZ and the 2^{10}-1 NRZ have an optimum bandwidth of 41.4 MHz which is to be expected because of their similar power spectra. The "1010"

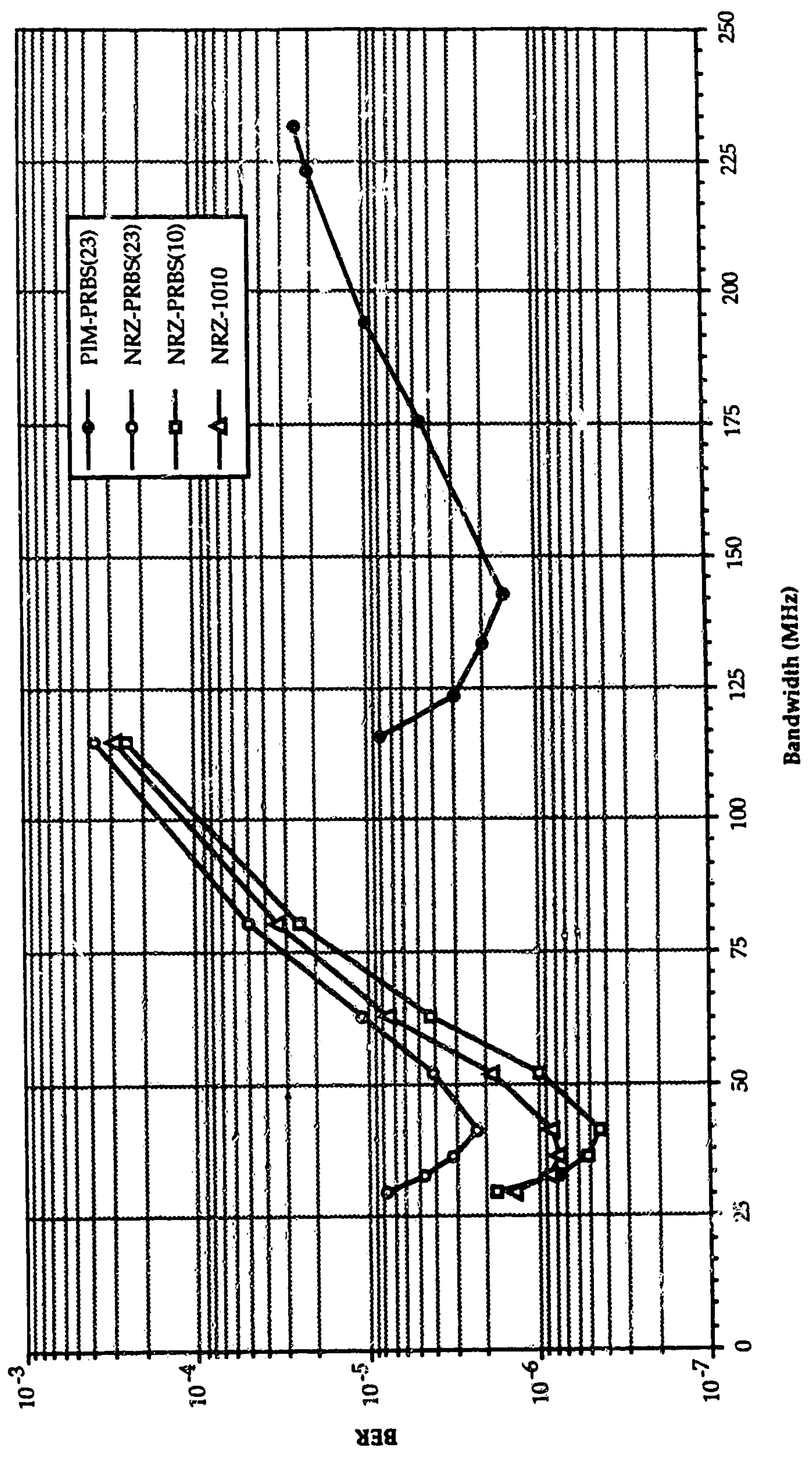


Figure 4.8 The BER versus the bandwidth of the system.

NRZ has an optimum bandwidth of 36.2 MHz which is slightly smaller than the optimum bandwidth of the NRZ PRBSs. This can be accounted for by the differences in their power spectra; the power spectrum of the "1010" NRZ contains only discrete spectral components which are spaced at multiples of 45 MHz, while the power spectra of the NRZ PRBSs are more representative of a continuous spectrum. The optimum bandwidth of approximately 40 MHz for the 90 Mbit/s NRZ data sequences is reasonable, since the magnitude of the power spectral density at this frequency is one-half of its maximum value.

The 2^{23}-1 PIM has an optimum bandwidth of 143 MHz, which is larger than predicted. In Section 2.6, it was estimated that the 3B16P1G PIM code would require three times the bandwidth of the encoding NRZ random data stream. Thus, the 2^{23}-1 PIM was expected to have an optimum bandwidth of 124 MHz which is three times 41.4 MHz, the optimum bandwidth of the encoded 2^{23}-1 NRZ. There are two factors which may explain why the experimental optimum bandwidth of the 2^{23}-1 PIM is larger than its predicted value. One is that the slight differences in the shape of the theoretical average power spectra of the NRZ and PIM codes, shown in Figure 2.10, are significant. The other is that the pulses transmitted for the NRZ and the PIM codes are not identical. This could cause degradation in the bandwidth performance of the PIM code by either effecting the way that the logic gates respond to the narrower pulses or the increased difficulty in synchronizing the pulses to the decision sampling point. Either factor could cause a change in the ratio of the optimum bandwidth of the 2^{23}-1 PIM to that of the NRZ PRBS.

Using the experimental values obtained for the optimum bandwidth, it can be calculated that the 2^{23}-1 PIM requires 3.45 times instead of 3 times more bandwidth than the NRZ PRBSs. This equates to a 3.45 noise power penalty for the 2^{23}-1 PIM in an optimized bandwidth system because of the increase in noise power at the output of the receiver.

4.5.3 SNR

The BER is directly related to the SNR. In the experimental system, BER measurements were taken as a function of the SNR for various NRZ and PIM data sequences. Each BER measurement was arrived at by taking the average of ten readings from the error detector, which calculated the BER on the basis of 100 bit errors. The SNR was calculated by using the average DC signal power divided by the measured noise power. When no noise was added to the signal, the average DC signal power was calculated from the maximum amplitude of the signal that was measured at the output of the receiver. Similarly, when no signal was transmitted, the noise power was measured at the output of the receiver with a HP 436A Power Meter. Each BER measurement was taken at its optimum threshold level and at the maximum bandwidth of the system (231 MHz).

The BER measurements were made for the PIM code using the 2^{23}-1 PIM, the "1111" PIM and the "0000..." PIM; and for the NRZ code using the 2^{23}-1 NRZ, the 2^{10}-1 NRZ and the "1010" NRZ. These results are shown in Figure 4.9. It can be seen that the three experimental BER curves for the NRZ data sequences overlie each other until a SNR of approximately 25 is reached. At this point the BER curve of the 2^{23}-1 NRZ diverges from that of the other two NRZ data sequences. Previously, in Section 4.4.1 where the effect that the threshold level has on the BER was examined, it was noted that the performance of the 2^{23}-1 NRZ appeared to have been affected by the limitations of the system. Because of these anomalies, the 2^{10}-1 NRZ will be used to compare to the PIM data sequences.

This experimental BER data which was obtained for the NRZ and PIM data sequences, was curve fitted and the resulting equations were used to aid in

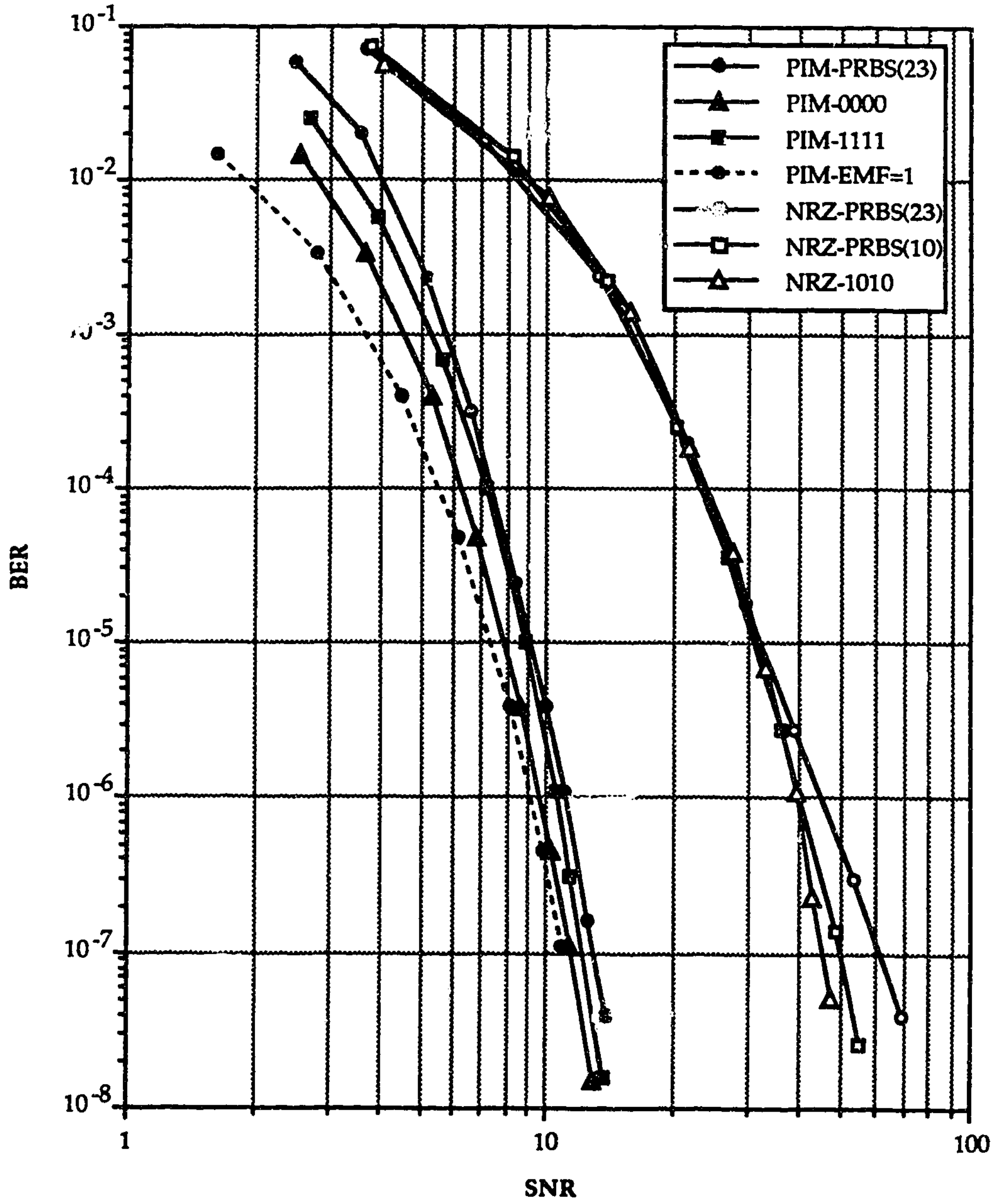


Figure 4.9 The BER versus the SNR for the 3B16G1P PIM and the NRZ binary code.

the calculation of the experimental error multiplication factor and peak power gain of the various PIM sequences.

4.5.3.1 Error Multiplication Factor

The three experimental BER curves for the PIM data sequences lie to the left of the BER curves for the NRZ data sequences. This shift to a lower SNR is due to the higher peak powers that the PIM code can generate as compared to the NRZ code for a fixed average power. The 3B16P1G PIM code has a theoretical peak power gain of 4.5 over that of the NRZ code with a duty cycle of 0.5. Assuming that a PIM data sequence existed that had an error multiplication factor of one, then its BER curve would be a mirror image of the BER curve for the NRZ data sequences but shifted to the left by the value of its peak power gain. In this case, an error multiplication factor of one would mean that for each error that occurred in the PIM data stream, only one error would be detected in the decoded binary data sequence.

The BER curve for this ideal (but nonexistent) PIM data sequence with an error multiplication factor of one (EMF=1 PIM), was created by dividing the 2^{10}-1 NRZ SNR values by the theoretical peak power gain of 4.5. The BER curve for this EMF=1 PIM has been included in Figure 4.9. An increase in the error multiplication factor will cause an equivalent increase in the BER which will shift the BER curve upwards. Thus, the vertical displacement of the experimental BER curves from the BER curve for the PIM EMF=1 is equivalent to the error multiplication factor for that PIM data sequence.

The order in which the experimental BER curves for the PIM data sequences are displaced from the BER curve for the PIM EMF=1 are as predicted from the theoretical analysis of the error multiplication factor that is contained in Section 2.4. There it is shown that the "0000" PIM has the smallest error multiplication factor followed by the "1111" PIM which has a slightly larger error multiplication factor. The 2^{23}-1 PIM was expected to have the largest error multiplication factor because it is the only one of the three PIM data sequences in which the encoded data is position dependent. This causes an increase in the error rate and correspondingly in the error multiplication factor because, when an error occurs, the decoded data stream is shifted out of position by three NRZ binary bits and the remaining decoded data is in error until the decoded data stream is shifted back into its original position.

A closer examination will be made of the error multiplication factors for the 2^{23}-1 PIM and the "0000" PIM. The experimental error multiplication factors of these two PIM data sequences are shown in Figure 4.10. They were calculated from the vertical displacement of their BER curves from the BER curve for the EMF=1 PIM. The slope of the error multiplication factor curves is more indicative of systematic errors probably introduced by inaccuracies in the curve fitting of the data. This is because the error multiplication factor only depends on the encoded bit sequence and the hardware implementation of the decoder; thus, it should be independent of the SNR.

By taking the average, one gets an experimental error multiplication factor for the "0000" PIM of 2.5 which is close to the theoretical value of 2.4. Similarly, the 2^{23}-1 PIM has an average experimental error multiplication factor of 12.5. An indication of the magnitude of the increase in the error rate due to encoding of position dependent data can be calculated by dividing the experimental error multiplication factor of the 2^{23}-1 PIM by the experimental error multiplication factor of the "0000" PIM. Thus, for this hardware configuration of the decoder, the encoding of position dependent data will cause roughly a 5.0 times increase in the error rate.

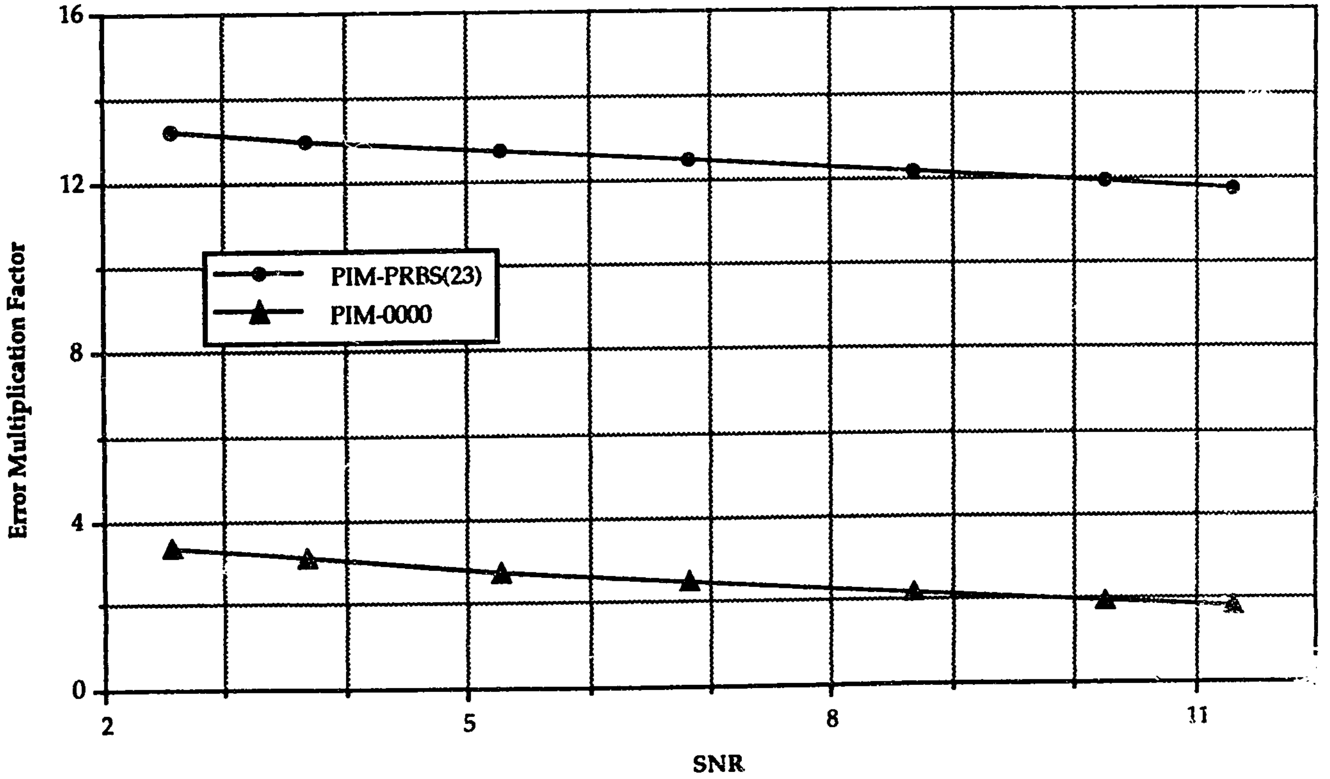


Figure 4.10 The experimental error multiplication factors for the 3B16P1G PIM code.

4.5.3.2 Power Gain

The theoretical peak power gain of the 3B16P1G PIM code over the NRZ code was calculated to be 4.5 for the case of the EMF=1 PIM. As the error multiplication factor is increased, it shifts the BER curve upwards. Because of the slope of the BER curve, this upward shift also translates into an effective shift to the right, toward the BER curves of the NRZ data sequences. Thus, an increase in the error multiplication factor will decreases the peak power gain by a factor that is related to the slope of the BER curve at that point.

The peak power gain for the 2^{23}-1 PIM and the "0000" PIM were calculated by dividing the SNR of the 2^{10}-1 NRZ by the SNR of the corresponding PIM data sequences at constant values of the BER. The resulting curves for the peak power gain are shown in Figure 4.11, where they are compared to the peak power gain for the EMF=1 PIM which is constant at the maximum theoretical value of 4.5. From Figure 4.11, it can be seen that the maximum peak power gain of the 2^{23}-1 PIM and the "0000" PIM occurs at the lowest measured BER and the peak power gain decreases as the BER is increased. This decrease is due to the changing slope of the BER curves where the gentler the slope, the greater the effect on the peak power gain due to the error multiplication factor.

At a BER of 10^{-7}, the peak power gain for the "0000" PIM which has an error multiplication factor of 2.5, is 4.2 or 93% of the theoretical peak power gain. Similarly, at the same BER, the peak power gain for the 2^{23}-1 PIM which has an error multiplication factor of 12.5, is 3.8 or 85% of the theoretical peak power gain. These reductions in the peak power gain from the theoretical maximum value are due to the error multiplication that occurs in the PIM code.

For the case where the bandwidth of the system has been optimized, the 3B16P1G PIM code will incur a noise power penalty because of the larger

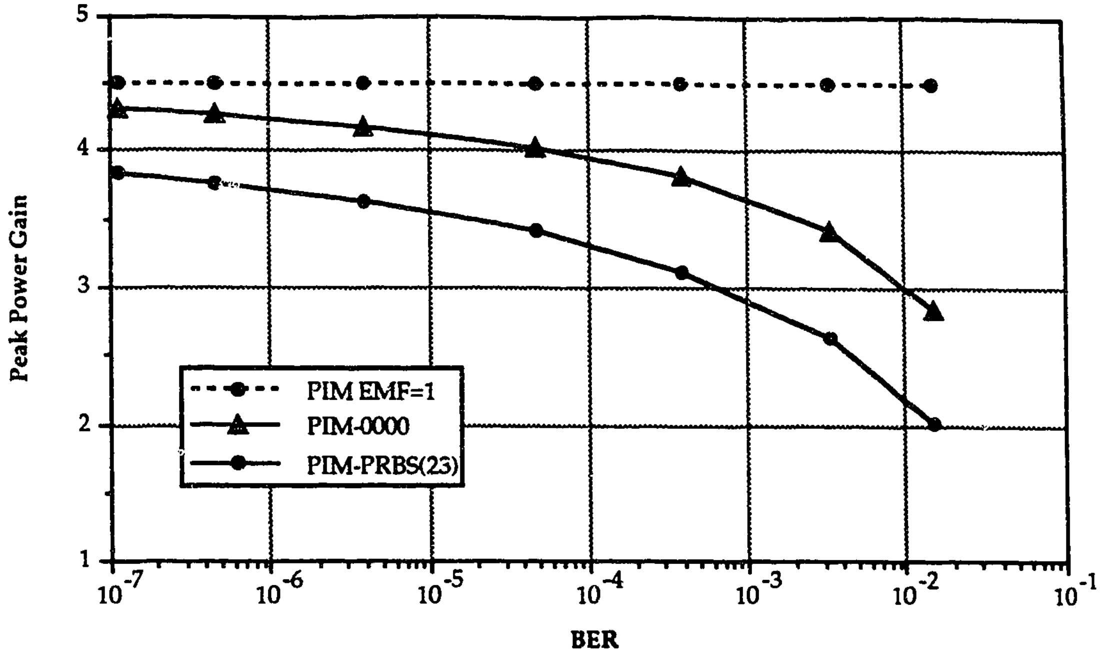


Figure 4.11 The peak power gain versus the BER for the 3B16P1G PIM code.

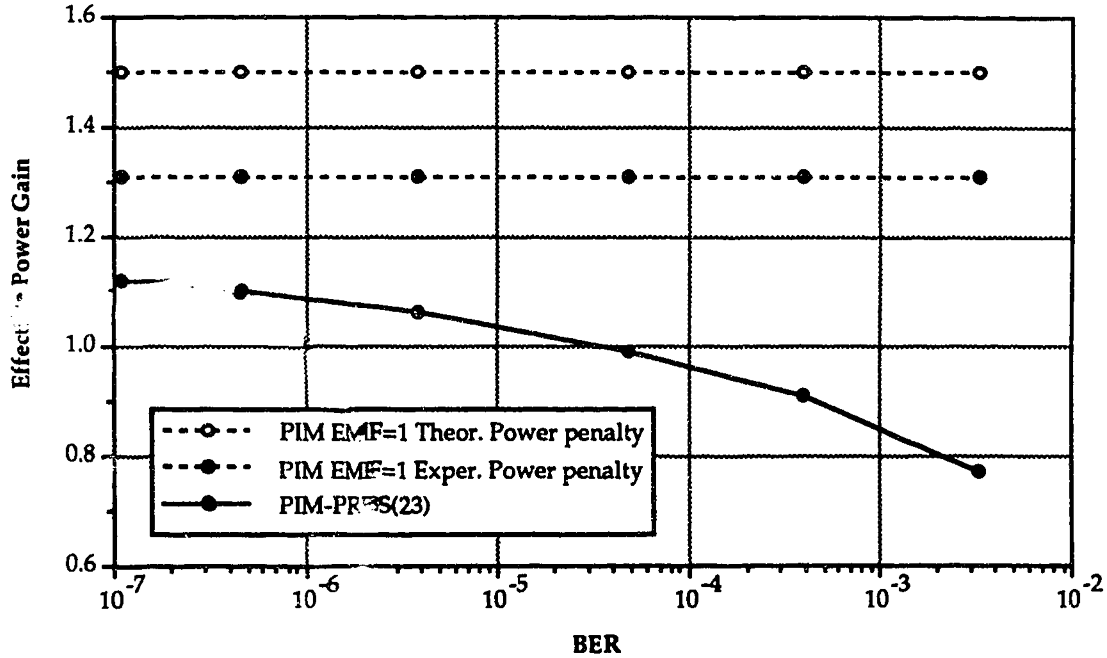


Figure 4.12 The effective power gain versus the BER for the 3B16P1G PIM code.

bandwidth that it requires as compared to the NRZ code. In this situation, the effective power gain of a PIM data sequence is equal to the peak power gain divided by the noise power penalty, where the theoretical noise power penalty of the 3B16P1G PIM code was predicted to be 3.0. Thus, the maximum effective power gain of the EMF=1 PIM was calculated to be 1.5 by dividing its peak power gain of 4.5 by the theoretical noise power penalty of 3.0. In Section 4.2, however, the experimental noise power penalty of the 3B16P1G PIM code was found to be 3.45. Using this experimental value for the noise power penalty, the maximum effective power gain for the EMF=1 PIM reduces to 1.31.

Similarly, the effective power gain of the 2^{23}-1 PIM was calculated by dividing its peak power gain by the experimental noise power penalty of 3.45. This is shown in Figure 4.12, together with the effective power gain of the EMF=1 PIM for both the theoretical and experimental values of the noise power penalty. These curves for the effective power gain exhibit the same characteristics as the curves for the peak power gain, except that they have been decreased by the value of the noise power penalty. Thus, the magnitude of the effective power gain of the 2^{23}-1 PIM also decreases as the BER increases and, at a BER of 10^{-7}, it has an effective power gain of 1.12, or 85% of the maximum experimental effective power gain of 1.31. This value of the effective power gain of the 2^{23}-1 PIM is only 75% of the theoretically predicted value of 1.5 because of the effect that the error multiplication factor has on the BER and because of the increase in the noise power penalty due to the larger than expected bandwidth.

The performance of the 2^{23}-1 PIM in the experimental system will give the best estimate of the expected performance of an actual encoded PIM data stream. Thus, it is expected that, in a fixed non-optimized bandwidth system which is operating at a BER of less than 10^{-7}, the 3B16P1G PIM code will have an effective power gain over that of the NRZ code of at least 3.8. In an optimized bandwidth system which is operating over the same range of BERs, the 3B16P1G PIM code will have an effective power gain over that of the NRZ code of at least 1.12.

CHAPTER 5

SUMMARY AND CONCLUSIONS

In recent years there has been increased interest in the use of LEDs as sources for short haul and low bit-rate SMF systems. In order to increase the viability of using a LED in these systems, a new family of binary codes has been developed, which would allow the LED to be overdriven, thus increasing the peak power coupled into the fiber. This family of codes is based on an asynchronous pulse interval modulation (PIM) scheme in which the encoded data is represented by the length of the interval between adjacent pulses. Because of problems associated with the variable width of the PIM intervals, the "nB2$\cdot$2^{n}PmG" family of PIM codes was developed. Here each sequence of "n" binary bits can be encoded into one of its two possible PIM intervals with a guardtime "m" that is located after each PIM pulse. The PIM interval selected is the one that limits the RDS or the variation of the PIM data stream with respect to the encoded NRZ data stream.

In this thesis, the development of the "nB2$\cdot$2^{n}PmG" family of PIM codes is discussed, together with an analysis of their potential power gain over the NRZ code. It was found that, for the non-optimized bandwidth system, the potential power gain of the PIM code was equivalent to one-half of the inverse of its duty cycle. In practice, the bandwidth of the system is optimized to the encoded data stream. In this case, the PIM code was found to have a potential power gain that is equal to the the number of binary bits encoded, divided by two.

After further analysis, the 3B16P1G PIM code was selected from the "nB2$\cdot$2^{n}PmG" family of codes to be implemented into hardware in order to test the feasibility of this family of codes. An experimental communication system was constructed to operate at 90 Mbit/s over an attenuation-limited electrical link. Because of the high speed of operation, special asynchronous buffers, which are located in the PIM encoder and decoder, were designed. These buffers are used to store the binary data before encoding and after decoding in order to compensate for the variable frame width of the PIM interval. The FIFO buffer that was designed allows simultaneous read and write operations with an access time of less than 4 nsec.

The power spectral distribution of the 3B16P1G PIM code was generated both experimentally for the encoded NRZ 2^{23}-1 PRBS and by a computer simulation for an encoded random binary sequence. There was good correlation between the two average power spectral distributions; in both cases, the minima of the continuous component of these spectra occurred at DC and at multiples of the maximum clock rate of 270 MHz. This together with the smaller values of the power spectra that are contained at the lower frequencies, will reduce the effect that AC coupling of the signal has on the performance of the system by decreasing the baseline wander. The experimental spectral distribution contains discrete components which should, theoretically, not be present. This may be due to the non-symmetry of the received pulses, or due to clock feedthrough from the encoder circuit. In either case, the discrete components of the spectra cannot be considered to be a dependable source for the recovery of the clock signal.

For the experimental system, BER measurements were taken for both the 3B16P1G PIM code and the NRZ code as a function of the threshold level, the system bandwidth and the SNR. Whenever possible, a 2^{23}-1 PRBS was used (or encoded with the PIM code) for testing, since of all the possible data sequences that could be generated, it is the one that best represents an actual data stream. It should be mentioned that there were unacceptable degradations in the performance of the NRZ 2^{23}-1 PRBS in the system, so, for the purpose of comparison, the experimental data collected for the NRZ 2^{10}-1 PRBS were used instead. This degradation was due to the bit sequence dependence of the NRZ code. For the case of the PIM code, there were insignificant differences between the performance of the encoded NRZ 2^{23}-1 PRBS and the NRZ 2^{10}-1 PRBS. For the repetitive PIM encoded NRZ sequences, which may not have the same error multiplication factors, the performance of the PIM code will depend on the data sequence encoded. In an actual system, where the encoded data are usually quite diverse, the PIM code can be considered bit sequence independent.

Theoretical analysis of the threshold level of the PIM code shows that the optimum threshold level is a function of the duty cycle, the error multiplication factor and the peak SNR. From the theoretical results, it was expected that the optimum threshold level would be located above the mid-point of the signal's amplitude. This was verified by the experimental threshold level measurements. It should be noted however, that since the optimum threshold level is a function of the error multiplication factor and the peak SNR, that if these factors vary over time they could cause a decrease in the performance of the system. The effect that these factors have on the performance of the system will decrease as the peak SNR is increased, or correspondingly as the BER of the system is decreased.

The effective error multiplication factor is a combination of the error multiplication factors that are attributed to the threshold violation errors and the erasure errors, and represents the average increase in the error-rate that occurs when an erroneous PIM interval is decoded. The theoretical value of the effective error multiplication factor calculated for the PIM encoded NRZ "0000..." sequence was experimentally confirmed. In the case of the PIM encoded NRZ 2^{23}-1 PRBS, the effective error multiplication factor was experimentally determined to be approximately 12.5. If the 3B16P1G PIM code were implemented into an actual system, it is expected that it would have an effective error multiplication factor close to this value.

The effective error multiplication factor will decrease the performance of the system by increasing the BER for a constant SNR. This causes a decrease in the power gain for the PIM encoded sequence, equal to the change in the signal power required to reduce the BER to the value that would occur for a PIM encoded sequence with an error multiplication factor of one. Since the slope of the BER curve increases as the BER decreases, a smaller change in the signal power is required at lower BERs to compensate for the increased error rate. So, the effective error multiplication factor has a smaller effect on the power gain and on the degradation of the performance of the system as the BER is decreased.

In a system where the bandwidth has not been optimized, the experimentally obtained power gain of the 3B16P1G PIM code with respect to the NRZ code was 3.85 at a BER of 10^{-7}. This is only 85% of the theoretically calculated peak power gain. The difference between the two values is due to the power penalty that is associated with the the error multiplication factor of the PIM code. In the more likely case, where the bandwidth of the system has been optimized to the transmitted code, the experimental power gain for the 3B16P1G PIM code drops from 3.85 to 1.12 at a BER of 10^{-7}. This is only 75% of the theoretically calculated power gain. This decrease in the performance of the optimized bandwidth system as compared to the non-optimized system, is due to the larger than predicted optimum bandwidth of the 3B16P1G PIM code. Initially it was expected that the optimum system bandwidth for the 3B16P1G PIM code would be three times that of the NRZ code, but experimental measurements show that it actually requires 3.45 times the bandwidth. This will cause an increase in the noise power penalty which is responsible for the decrease the expected power gain of the 3B16P1G PIM code. As the BER decreases below 10^{-7} the power gain will increase up to a maximum of 4.5 for the non-optimized bandwidth system, and 1.3 for the optimized bandwidth system as the effect that the effective error multiplication factor has on the power gain is decreased.

To summarize, the results show that the "nB2·2^nPmG" family of PIM codes can be implemented at high speed. This, along with their bit sequence independence, excellent resynchronization characteristics and their potential power gain over the NRZ code, makes them a suitable candidate for short haul LED optical single-mode fiber systems. Though the performance of the PIM code is degraded by a variable optimum threshold level and a power penalty due to the error multiplication factor of the code, there should only be a small degradation in the actual performance of a typical system which is operated at a BER of 10^{-9}, because the effect that these factors have on the performance of the system decreases with the BER.

This thesis has investigated the "nB2·2^nPmG" family of PIM codes and has examined their feasibility for the use in a LED single-mode fiber optical system. However, further theoretical and experimental research could be carried out in the following areas:

1) Initial assumptions were made that the LED could be over driven without limit or with little ill effect. It is likely that the maximum power gain that can be achieved for the "nB2·2^nPmG" family of PIM codes will ultimately be limited by the characteristics of the source. Thus, the effects that overdriving the LED has on its characteristics, such as output power and reliability should be investigated.
2) Though this code is intended to be implemented in an optical fiber system, its performance has only been examined over an electrical link. There are differences, such as the characteristics

of the channel and the type of detector and amplifier used in the receiver, that exist between electrical and optical systems which could affect the performance of the code. By implementing the 3B16P1G PIM code into a single-mode optical fiber system, its performance can be investigated with respect to different system parameters.

3) Clock recovery is an important aspect of a system and has not been covered in this thesis. The power spectrum of the PIM code contains no discrete spectral lines that are located at multiples of the clock frequency. Thus, the clock would have to be recovered by non-linear processing of the received signal or generated by the use of a local clock, such as a multivibrator which operates at the clock frequency and is resynchronized whenever a PIM pulse is detected. Both of these methods for obtaining a clock signal for the PIM decoder could be implemented into the experimental system in order to investigate their effect on the performance of the system.

4) In the case of the "nB2·2^nPmG" family of PIM codes, once an error occurs, usually the rest of the received data is erroneous until the buffer overflows or underflows. The length of time that this takes to occur is dependent on the design and size of the buffer in the decoder and on the data sequence encoded. A more detailed analysis could be undertaken in order to find the effect that the receiver buffer has on the error rate or the effective error multiplication factor of the PIM code and thus, the optimum size for the buffer.

5) The "nB2·2^nPmG" family of PIM codes has a form of error

> detection. When an error occurs it incorrectly modifies the RDS. Eventually the RDS will exceeds its limits, which indicates that an error has occurred. Because of the variable lag in detecting the error it may not be possible to determine the erroneous PIM interval. Further study could be conducted on the error detection mechanism of the "nB2·2^nPmG" family of PIM codes and the implementation of the error detection capability of the PIM codes could be investigated.

It is hoped that the work performed in this thesis will lead to a better understanding of the "nB2·2^nPmG" family of PIM codes and encourage further research and development in this area.

BIBLIOGRAPHY

1) J. Palais, *Fiber Optic Communications*, second edition, Prentice Hall, New Jersey, 1988.

2) C.D. Chaffee, *The Rewiring of America The Fiber Optics Revolution*, Academic Press Inc., Florida, 1988.

3) T.E. Bell, "Fiber Optics", *IEEE Spectrum*, Vol.25, No.11, November 1988, pp.97-102.

4) T. Li, "Advances in Lightwave Systems Research", *AT&T Technical Journal*, Vol.66, No.1, January-Feburary 1987, pp.5-18.

5) R. Olshansky, D. Fye, J. Manning, M. Stern, E. Meland, W. Powazinik, L. Ulbricht and R. Lauer, "High-Power InGaAsP Edge-Emitting LEDs for Single-Mode Optical Communication Systems", *Electronic Letters*, Vol.21, No.17, August 15th 1985, pp.730-731.

6) T. Uji and J. Hayashi, "High Power Single-Mode Optical Fibre Coupling to InGaAsP 1.3 μm Mesa-Structure Surface Emitting LEDs", *Electronic Letters*, Vol.21, No.10, May 9th 1985, pp.418-419

7) G. Arnold, H. Gottsmann, O. Krumpholz, E. Schlosser and E.A. Schurr, "1.3 μm Edge-Emitting Diodes Launching 250 μm into Single-Mode Fibre at 100 ma", *Electronic Letters*, Vol.21, No.21, October 10th 1985, pp.993-994.

8) P.W. Shumate, J.L. Gimlett, M. Stern, M.B. Romeiser and N.K. Cheung, "Transmission of 140 Mbit/s Signals over Single-Mode Fibre using Surface- and Edge-Emitting 1.3 μm LEDs", *Electronic Letters*, Vol.21, No.12, June 6th 1985, pp.522-524.

9) R.H. Saul, W.C. King, N.A. Olsson, C.L. Zipfel, B.H. Chin, A.K. Chin, I. Camlibel and G. Minneci, "180 Mbit/s, 35 km Transmission over Single-Mode Fibre using 1.3 μm Edge-Emitting LEDs", *Electronic Letters*, Vol.21, No.17, August 15th 1985, pp.773-775.

10) L.W. Ulbricht, M.J. Teare, R, Olshansky and R.B. Lauer, "Loss-Limited Transmission at 140 Mbit/s over 30 km of Single-Mode Fibre using a 1.3 μm LED", *Electronic Letters*, Vol.21, No.19, September 12th 1985, pp.860-861.

11) J.L. Gimlett, M. Stern, R.S. Voldhanel, N.K. Cheung, G.K. Chang, H.P. Leblanc, P.W. Shumate and A. Suzuki, "Transmission Experiments At 560 Mbit/s and 140 Mbit/s using Single-Mode Fibre and 1300 nm LEDs", *Electronic Letters*, Vol.21, No.25/26, December 5th 1985, pp.1198-1200.

12) M. Stern, D.J. Millicker, A.E. Elrefaie and K. Runge, "Transmission at 1 Gbit/s over 7.4 km of Single-Mode Fibre using an Edge-Emitting LED and PINFET Receiver", *Electronic Letters*, Vol.21, No.23, April 9th 1987, pp.386-387.

13) T. Ohtsuka, M. Fujimoto, K. Yamaguchi, A. Taniguchi, H. Naitou and Y. Nabeshima, "Gigabit Single-Mode Fiber Transmission using 1.3-µm Edge-Emitting LEDs for Broad-Band Subscriber Loops", *Journal of Lightwave Technology*, Vol. LT-5, No.10, October 1987, pp.1534-1541.

14) L.L. Kanters, G.D. Khoe and L. Bosselaar, "LED/DFSM Fiber Systems for Maximum Transmission Capacity", *14th ECOC'88*, 1988, pp.183-186.

15) M.H. Reeves, A.R. Hunwicks, W. Zhao, S.G. Methley, L. Bickers and S. Hornung, "LED Spectral Slicing for Single-Mode Local Loop Applications", *Electronic Letters*, Vol.24, No.7, March 31st 1988, pp.389-390.

16) D.M. Fye, "Low-Current 1.3-µm Edge-Emitting LED for Single-Mode Fiber Subscriber Loop Applications", *Journal of Lightwave Technology*, Vol. LT-4, No.10, October 1986, pp.1546-1551.

17) L.W. Ulbricht, M.J. Teare, D.M. Fye and R.B. Lauer, "Subscriber Loop Transmission using Low Operating Current Light Emitting Diodes Subjected to Temperature Extremes", *Journal of Lightwave Technology*, Vol. LT-5, No.9, September 1987, pp.1258-1262.

18) H.Y. Tam and M. Beale, "Effects of Temperature on LED-Based Single-Mode WDM Systems for Subscriber-Loop Applications", *2nd IEE National Conference on Telecommunications*, April 2-5 1989, pp.94-97.

19) P.W. Shumate, "Optical fibers reach into homes", *IEEE Spectrum*, Vol.26, No.2, February 1989, pp.43-47.

20) D. Kalish and L.G. Cohen, "Single-Mode Fiber: From Research and Development to Manufacturing", *AT&T Technical Journal*, Vol.66, No.1, January-Feburary 1987, pp.19-32.

21) W.B. Jones Jr., *Introduction to Optical Fiber Communication Systems*, Holt, Rinehart and Winston Inc., New York, 1988.

22) L. Hafsljaer and A.S. Sudbo, "Attenuation and Bit-Rate Limitations in LED/Single-Mode Fiber Transmission Systems", *Journal of Lightwave Technology*, Vol. 6, No.12, December 1988, pp.1793-1796.

23) A. Frenkel and R.C. Menendez, "Analysis of the Applications of an LED/Single-Mode Fiber Transmission System in the Distribution Portion of the Loop", *ICC'87*, Seattle, 1987, pp.1542-1549.

24) J.L. Gimlett and N.K. Cheung, "Dispersion Penalty Analysis for LED/Single-Mode Fiber Transmission Systems", *Journal of Lightwave Technology*, Vol. LT-4, No.9, September 1986, pp.1381-1392.

25) V.A. Bhagavatula, "Recent Designs of Single-Mode Fiber for Telecommunications", *11th ECOC'85*, VENICE, 1985, pp.47-53.

26) M. Stern, J.L. Gimlett, L. Curtis, N.K. Cheung, M.B. Romeiser, W.C. Young and P.W. Shumate, "Bidirectional LED Transmission on Single-Mode Fibre in the 1300 and 1500 nm Wavelength Regions", *Electronic Letters*, Vol.21, No.20, September 26th 1985, pp.928-929.

27) F.P. Kapron, "Fiber-Optic System Tradeoffs", *IEEE Spectrum*, Vol.22, No.3, March 1985, pp.68-75.

28) T. Nakagami and T. Sakurai, "Optical and Optoelectronic Devices for Optical Fiber Transmission Systems", *IEEE Communications Magazine*, Vol.26, No.1, January 1988, pp.28-33.

29) G. Arnold, F. J. Berlec, H. Gottsmann, and W. Pfister, "Emissions and Coupling Properties of 1.3 µm V-Grove Edge-Emitting LEDs", *Journal of Optical Communications*, Vol.6, No.4, 1985, pp.127-131.

30) R. Plastow, "LEDs for Fiber Optic Communication Systems", *Photonics Spectra*, September 1985, pp.109-116.

31) P.W. Shumate, "Temperature Effects on Dispersion and Loss for High-Bit-Rate LED-Based Lightwave Systems", *Electronic Letters*, Vol.22, No.1, January 2nd 1986, pp.1-2.

32) W. Tomasi, *Advanced Electronic Communications Systems*, Prentice-Hall Inc., New Jersey, 1987.

33) W. van Etten, "Coupling of LED Light into a Single-Mode Fiber", *Journal of Optical Communications*, Vol.9, No.3, 1988, pp.100-101.

34) R.W. Dison and N.K. Dutta, "Lightwave Device Technology", *AT&T Technical Journal*, Vol.66, No.1, January-Feburary 1987, pp.73-83.

35) M.Fukuda, "Laser and LED Reliability Update", *Optical Fiber Communication Conference,* 1988 Technical Digest Series, Vol.1, Section-WJ1, January 25-28 1988, New Orleans.

36) H. B. Killen, *Digital Communications with Fiber Optics and Satellite Applications,* Prentice-Hall Inc., New Jersey, 1988.

37) C.E. Shannon, "Communications in the Presence of Noise", *Proceedings of the IRE,* Vol.37, January 1949, pp.10-21.

38) R. Petrovic, "Low Redundancy Optical Fiber Line Code", *Journal of Optical Communications,* Vol.9, No.3, 1988, pp.108-111.

39) J.J. O'Reilly, B. Tech, C. Eng and W. Yichao, "Line Code Design for Digital Pulse-Position Modulation", *IEE Proceedings* , Vol.132, Pt. F, October 1985, pp.441-446.

40) I. Garrett, "Pulse-Position Modulation for Transmission over Optical Fibers with Direct or Heterodyne Detection", *IEEE Transactions on Communications,* Vol. COM-31, No.4, April 1985, pp.518-527.

41) I. Garrett, "Toward the Fundamental Limits of Optical-Fiber Communications", *Journal of Lightwave Technology,* Vol. LT-1, No.1, March 1983, pp.131-138.

42) I. Garrett, "Digital Pulse-Position over Dispersive Optical Fibre Channels", *IEEE GLOBECOM'83,* Vol.2, No.1, November 1983, pp.733-737.

43) N.M. Calvert, M.J.N. Sibley and R.T. Unwin, "Experimental Optical Fibre Digital Pulse-Position Modulation System", *Electronic Letters,* Vol.24, No.2, January 21st 1988, pp.129-131.

44) Y. Ueno and M. Kajitani, "A Semiconductor Laser Communication System Using Differential Pulse Position Modulation", *NEC Research & Development,* Vol.29, April 1973, pp.31-37.

45) K.S. Visvanatha, P.A. Goud and C.G. Englefield, "Jitter Reduction in Self-Sustaining Monostable Clock Recovery Circuit", *Electronic Letters,* Vol.22, No.16, July 31st 1986, pp.855-856.

46) Y.A. Shreider, *The Monte Carlo Method: The Method of Statistical Trials,* Pergamon Press Ltd., London, 1966.

47) F.G. Stremler, *Introduction to Communication Systems,* second edition, Addison-Wesley Oybkushing Co., Philippines, 1982.

48) P.O. Borjesson and C.W. Sundberg, "Simple Approximations of the Error Function $Q(x)$ for Communications Applications", *IEEE Trans. Commun.*, Com-27, March 1979, pp. 639-643.

49) *MECL Integrated Circuits,* Motorola Inc., U.S.A., 1985.

50) W.R. Blood Jr., *MECL Systems Design Handbook,* Fourth Edition, Motorola Inc., U.S.A., 1983.