

---

**A NOVEL MAXIMAL-LENGTH SEQUENCE  
SYNCHRONISATION NETWORK.**

---

**Allan Mark DAVIDSON**  
*University of Cape Town*  
*Cape Town, South Africa*

DIGITAL COMMUNICATIONS RESEARCH GROUP  
University of Cape Town  
5 July 1995

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

---

**A NOVEL MAXIMAL-LENGTH SEQUENCE  
SYNCHRONISATION NETWORK.**

---

## ABSTRACT

Spread Spectrum has become a popular digital modulation scheme in recent years. The advantages the scheme offers, at the expense of bandwidth, make it attractive in a multitude of commercial applications. The most common method, and the one of interest in this thesis, of generating Spread Spectrum is multiplying the data waveform by a wideband, digitally generated waveform. This is referred to as Direct Sequence Spread Spectrum.

The characteristics of Spread Spectrum systems are determined by the spreading waveform. A common group of spreading waveforms, and the ones dealt with in this text, are the maximal-length sequences. These are a class of pseudorandom waveforms. Their properties include a two valued autocorrelation function with its maximum value at no code-phase offset. This allows for multiple access to a single resource and the suppression of multi-path interference as adjacent codes have little effect on each other.

This same property requires that the receiver must accurately align its replica of the spreading waveform to the transmitted waveform in order to despread the received waveform and demodulate the data. Common methods of synchronisation use a two pronged solution. Firstly the correct code phase is determined. This is referred to as code acquisition. Secondly the clocking frequency of the received waveform must be resolved in order to precisely align the two sequences. This is referred to as code tracking. Receivers therefore tend to be complex and expensive.

This thesis involved the investigation of two pseudo-noise synchronisation networks proposed by J.G. van de Groenendaal. These networks offered both code acquisition and tracking in a single robust loop.

The investigation, done in co-operation with J.G. van de Groenendaal, pursued two avenues. Firstly the loops were simulated. This method allows for the easy alteration of system parameters. Valuable insight into the loop dynamics can thus be gained. Secondly the loops were built on the bench. This allows for the practical confirmation of the results of the simulation.

Both synchronisation loops were based on variations of the maximal likelihood phase detector. This phase detector is formed by taking the product of the first derivative with respect to time of the receiver's replica of the transmitted waveform and the received waveform. The initial investigation involved calculating the phase information generated by this phase discriminator for a variety of code-phase and frequency offsets. It was found that there were two stable points in the baseband Spread Spectrum search grid, a grid where a cell consists of a certain code-phase and frequency offset. These stable points existed at no frequency offset, which means that the loops should track the input frequency, and a one or no code-phase offset, which means that the loops should acquire either code-phase.

A simple model where the novel synchronisation loop's conditions are represented by a 'ball' resting on the baseband Spread Spectrum search grid as expressed in terms of the integrated phase output of the maximal likelihood phase discriminator was developed. In this model the 'ball' will roll around the surface until one of the two stable points is entered. This describes quite accurately the paths the novel synchronisation loop does in fact take through the baseband Spread Spectrum search grid.

The first loop is based directly on the maximal likelihood phase detector. The differentiator is thus in the feedback path of the loop. This results in the loop being unstable and parameter sensitive. Moving

the differentiator into the input path, as in the second loop, resulted in a more stable loop. This loop therefore offered a complete, simple synchronisation solution.

The novel synchronisation loop with the differentiator in the input path was found to operate at signal-to-noise ratios of -2 dB. Improvement of this signal-to-noise ratio does not offer any advantages in a Spread Spectrum environment as the loop needs to work in a coherent system where the radio frequency carrier must be resolved before the receiver's pseudo-noise sequence can be synchronised. A radio frequency carrier cannot be easily resolved at signal-to-noise ratios lower than 0 dB. The loop was further adapted to operate in the data environment. Under conditions of data modulation the received waveform is randomly inverted by the data. This results in the loop being driven out of lock. The phase discriminator's slope, having locked on a certain polarity, cannot track an input of the opposite polarity. The loop was adapted by including detection circuitry that would monitor the state of the receiver with respect to the incoming data waveform and alter the polarity of the of the discriminator's slope where necessary.

During the prototyping of the loop on the bench certain implementations were investigated. These included the signed edge detector, a wideband low noise implementation of a square wave differentiator, and the synchronous oscillator, a form of injection locked oscillator. The loop was shown to achieve synchronisation.

The novel synchronisation loop with the differentiator in the input path is thus capable of synchronising two maximal-length sequences in both code-phase and frequency.

---

# CONTENTS

<b>ABSTRACT</b>	i
<b>LIST OF FIGURES</b>	v
<b>LIST OF TABLES</b>	ix
<b>PREFACE</b>	x
<b>1 INTRODUCTION.</b>	1
<b>2 DESCRIPTION OF COMMON SYNCHRONISATION NETWORKS.</b>	3
2.1 The Generation of Spread Spectrum Systems.	3
2.2 The Synchronisation of Pseudorandom Sequences.	5
2.3 Code Acquisition in Synchronisation Systems.	7
2.4 Code Tracking in Synchronisation Systems.	10
2.5 A System Level Overview of the Novel Maximal-length Sequence Synchronisation Network.	18
<b>3 OVERVIEW OF THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION NETWORK.</b>	19
3.1 Phase Information Generated in the Maximum Likelihood Phase Discriminator.	20
3.2 Characterisation of the Baseband Spread Spectrum Search Grid by Integrated Phase.	23
3.3 Methods of Detecting Lock in the Novel Maximal-length sequence Synchronisation Loop.	25
<b>4 RESULTS OF THE SIMULATION OF THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION NETWORKS.</b>	26
4.1 Simulation of Loop with the Differentiator in the Feedback Path.	26
4.2 Simulation of the Loop with the Differentiator in the Receiver Frontend.	32
4.3 Noise Performance of the Synchronisation Network with the Differentiator in the Receiver Frontend.	40
4.4 Adaptation of the Novel Synchronisation Loop to a Data Environment.	50
4.5 Comparison of the Novel Synchronisation Network with the Differentiator in the Input Path with Other Common Synchronisation Circuits.	67
<b>5 DESCRIPTION OF THE PRACTICAL IMPLEMENTATIONS OF THE VARIOUS COMPONENTS IN THE SYNCHRONISATION LOOP.</b>	69
5.1 Signed Edge Detection as an Alternative to Differentiation.	69

5.2	Implementation of the Adjustable Clock.	77
<b>6</b>	<b>DESCRIPTION OF THE PROTOTYPING OF THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION NETWORKS.</b>	<b>96</b>
6.1	Results from the Novel Synchronisation Network with the Differentiator in the Feedback Path.	96
6.2	Prototyping of the Novel Synchronisation Network with the Differentiator in the Input Path.	98
<b>7</b>	<b>CONCLUSION.</b>	<b>107</b>
<b>8</b>	<b>RECOMMENDATIONS.</b>	<b>109</b>
<b>1</b>	<b>TESLA SOURCE CODE.</b>	<b>110</b>
<b>2</b>	<b>CIRCUIT DIAGRAM OF PROTOTYPE LOOP.</b>	<b>114</b>
	<b>REFERENCES</b>	<b>116</b>

---

# LIST OF FIGURES

## Chapter 1

- 1.1 A Block Diagram of the Conceptual Layout of this Text. 2

## Chapter 2

- 2.1 Block Diagram of a Method of Generating Maximal-length Codes. 4  
2.2 Graph Showing the Two-valued Autocorrelation Function of a Maximal-length Code. 5  
2.3 Graph of the Power Spectrum of a 7 bit Maximal-length Sequence. 6  
2.4 Graph Showing the Baseband Search Area of a Spread Spectrum Receiver. 7  
2.5 Block Diagram of a Maximal-length Sequence Synchronisation Detection Circuit. 8  
2.6 Circuit Diagram of the Rapid Acquisition by Sequential Estimation Method. 9  
2.7 Block Diagram of the Optimum Maximal-length Sequence Tracking Loop. 11  
2.8 Graph of the Waveforms Generated by the Optimal Tracking Loop when the Sequences are more than a Chip Apart. 12  
2.9 Graph of the Waveforms Generated by the Optimal Tracking Loop when the Sequences are Within a Chip. 13  
2.10 Graph of the Discriminator Slope Generated by the Maximum Likelihood Phase Detector. 14  
2.11 Block Diagram of the Full-Time Early-Late Tracking Loop. 15  
2.12 Graphs of the Phase Discriminator Slopes of the Full-Time Early-Late Tracking Loop. 15  
2.13 Block Diagram of the Full-Time Early-Late Noncoherent Tracking Loop. 16  
2.14 Graphs of the Phase Discriminator Slopes of the Full-Time Early-Late Noncoherent Tracking Loop. 16  
2.15 Block Diagram of the Tau-Dither Early-Late Noncoherent Tracking Loop. 17

## Chapter 3

- 3.1 Block Diagram of the Novel Maximal-length Sequence Synchronisation Network with the Differentiator in the Feedback Path. 19  
3.2 Block Diagram of the Novel Maximal-length Sequence Synchronisation Network with the Differentiator in the Receiver Frontend. 20  
3.3 Block Diagram of the Simulation Used to Investigate the Maximum Likelihood Phase Discriminator. 21  
3.4 Graph of the Output of the Maximum Likelihood Phase Discriminator and the Resultant Phase Measurement. 22  
3.5 Graph of the Output of the Maximum Likelihood Phase Discriminator and the Resultant Phase Measurement. 22  
3.6 Graph of the Output of the Maximum Likelihood Phase Discriminator and the Resultant Phase Measurement. 23  
3.7 Graph of the Baseband Spread Spectrum Search Grid Characterised by Integrated Phase. 24  
3.8 Graph of the Baseband Spread Spectrum Search Grid Characterised by Integrated Phase. 24

**Chapter 4**

4.1	Block Diagram of the Novel Maximal-length sequence Synchronisation Network with the Differentiator in the Feedback Path as Implemented in TESLA.	27
4.2	Graph of the Two Maximal-length Sequences During the Simulation of the Synchronisation Loop.	28
4.3	Graph of the Two Maximal-length Sequences During the Simulation of the Synchronisation Loop for the Final 0.4 milliseconds.	29
4.4	Graph of the Output of the Maximum Likelihood Phase Discriminator.	29
4.5	Graph of the Output of the Maximum Likelihood Phase Discriminator.	30
4.6	Graph of the Phase-locked Loop's Loop Filter Output.	30
4.7	Graph of the Phase-locked Loop's Loop Filter Output from 3.5 to 4 milliseconds.	31
4.8	Block Diagram of the Novel Maximal-length sequence Synchronisation Network with the Differentiator in the Receiver Frontend as Implemented in TESLA.	32
4.9	Graph of the Two Maximal-length Sequences Achieving Lock.	34
4.10	Graph of the Output of the Maximum Likelihood Phase Discriminator.	34
4.11	Graph of the V.C.O. Driving Signal.	35
4.12	Graph of the Output of the Maximal Likelihood Phase Detector for the Negative Locking Case.	36
4.13	Graph of the V.C.O. Driving Signal for the Negative Going Case.	36
4.14	Graph of the Two Maximal-length Sequences Achieving Lock.	37
4.15	Graph of the Locking Times for a 15 Bit Maximal-Length Sequence .	38
4.16	Graph of the Locking Times for a 7 Bit Maximal-Length Sequence .	39
4.17	Block Diagram of the TESLA Simulation Used to Calculate the Integrated Phase in a Noisy Environment.	40
4.18	Graph of the Baseband Spread Spectrum Search Grid Characterised by Integrated Phase in a Noisy Environment.	41
4.19	Block Diagram of the Circuit Used in the Simulation of the Novel Maximal-length Sequence Synchronisation Network Under the Influence of Noise.	43
4.20	Graph of the Transmitted and Received Maximal-length Sequences.	44
4.21	Graph of the Spectrum of the Output of the Bandpass Filter.	44
4.22	Graph of the Output of the Integrator.	45
4.23	Graph of the Transmitted and Received Maximal-length Sequences.	46
4.24	Graph of the Spectrum of the Output of the Bandpass Filter.	46
4.25	Graph of the Output of the Integrator.	47
4.26	Graph of the Transmitted and Received Maximal-length Sequences.	48
4.27	Graph of the Output of the Integrator.	48
4.28	Graph of the Output of the Integrator.	50
4.29	Graph of the Output of the Integrator.	50
4.30	Graph of the Output of the Integrator.	51
4.31	Graph of the Output of the Integrator.	51
4.32	Block Diagram of the Synchronisation Network Adapted to the Data Environment Through Edge Detection.	53
4.33	Graph of the Output of the Multiplexer.	54
4.34	Graph of the Data and Despread Waveforms.	55
4.35	Graph of the Data and Despread Waveforms.	56
4.36	Block Diagram of the Novel Synchronisation Network Adapted to the Data Environment through Data Estimation.	57

4.37	Flowchart of the Operation of the Detection Circuitry for Data Estimation.	58
4.38	Logic Table for the Data Detection Circuitry.	59
4.39	Circuit Diagram of the Detection Circuitry Using the Spikes at the Output of the Maximal Likelihood Phase Discriminator.	60
4.40	Graph of the Spikes Before Inversion by the Detection Circuitry.	61
4.41	Graph of the Spikes After Inversion by the Detection Circuitry.	61
4.42	Plot of the Data and Despread Waveform.	62
4.43	Block Diagram of the Detection Circuitry Using the Despread Input and Receiver Clocking Waveform.	62
4.44	Graph of the Spikes for the Noiseless Case.	63
4.45	Graph of the Data and Invert Signals for the Noiseless Case.	63
4.46	Graph of the Output for the noiseless case.	64
4.47	Graph of the Data and Invert Signals at a Signal-to-noise Ratio of 8.4 dB.	64
4.48	Graph of the Data and Despread Transmitted Waveform at a Signal-to-noise Ratio of 8.4 dB.	65
4.49	Graph of the Data and Invert Signal at a signal-to-noise Ratio of 3 dB.	66
4.50	Graph of the Data and Despread Transmitted Waveform at a Signal-to-noise Ratio of 2.4 dB.	66
4.51	Graph of the Data and Despread Transmitted Waveform at a Signal-to-noise Ratio of 0 dB.	67
 <b>Chapter 5</b>		
5.1	Graph of the Frequency Response of the Ideal Differentiator.	70
5.2	Graph of the Frequency Response of a Practical Differentiator.	70
5.3	Circuit Diagram of a Practical Implementation of a Differentiator.	71
5.4	Circuit Diagram of an Edge Detector.	71
5.5	Circuit Diagram of the Signed Edge Detector.	72
5.6	Block Diagram of the Simulation of the TESLA Differentiator and the Signed Edge Detector.	72
5.7	Plot of the Output of the Low Pass Filter.	73
5.8	Plot of the Output of the TESLA differentiator model.	73
5.9	Plot of the Output of the Signed Edge Detector.	74
5.10	Input Square Wave to the Signed Edge Detector at 1 Megahertz.	75
5.11	Plot of the Output of the Signed Edge Detector.	75
5.12	Input Square Wave to the Differentiator at 1 Megahertz.	76
5.13	Plot of the Output of the Analog Differentiator.	76
5.14	Block Diagram of the Phase-locked Loop.	77
5.15	Graph of the Phase Response of a Tuned Circuit.	80
5.16	Phasor Diagram of the Waveforms Present in an Injection Locked Oscillator.	80
5.17	Graph Depicting the Reduction of Beat Frequencies due to Locking.	82
5.18	Circuit Diagram of the Synchronous Oscillator as Designed by Uzunoglu and White.	84
5.19	Circuit Diagram of the Synchronous Oscillator as Designed by Hickman.	85
5.20	Plot of the Magnitude and Phase Response of the Synchronous Oscillator at 1mV Input.	86
5.21	Plot of the Magnitude and Phase Response of the Synchronous Oscillator at 3mV Input.	86
5.22	Plot of the Magnitude and Phase Response of the Synchronous Oscillator at 4mV Input.	87
5.23	Plot of the Magnitude and Phase Response of the Synchronous Oscillator at 7mV Input.	87
5.24	Plot Displaying the Effects of the Feedback Capacitor $C_3$ on Storage Time. $C_3 = 100$ nF.	88
5.25	Plot Displaying the Effects of the Feedback Capacitor $C_3$ on Storage Time. $C_3 = 47$ nF.	88

5.26	Plot Displaying the Effects of the Feedback Capacitor $C_3$ on Storage Time. $C_3 = 10$ nF.	89
5.27	Circuit Diagram of the Coherent Phase-locked Synchronous Oscillator as Described by Uzunoglu and White.	90
5.28	Plot of the Lissajous Figure when the Input and Output are not in Phase.	91
5.29	Plot of the Lissajous Figure when the Input and Output are in Phase.	91
5.30	Plot of the Magnitude and Phase Response of the Colpitts Oscillator.	92
5.31	Plot of the Magnitude and Phase Response of the Optimised Colpitts Oscillator.	92
5.32	Graph of the 'Current Hump' at Different Injection Levels.	93
5.33	Graph of the Actual and Fitted 'Current Hump' Curves for the Optimised Colpitts Oscillator.	93
5.34	Graph of the Actual and Fitted 'Current Hump' Curves for the Synchronous Oscillator as Designed by Hickman.	94
<b>Chapter 6</b>		
6.1	Photograph of the Two Maximal-length Sequences in Lock at a One Bit Code-phase Offset.	97
6.2	Photograph of the Spikes at the Output of the Maximal Likelihood Phase Detector.	98
6.3	Circuit Diagram of the Passive Butterworth Filter.	100
6.4	Plot of the Two 16 Bit Maximal-length Sequences in Lock.	101
6.5	Plot of the Two 32 Bit Maximal-length Sequences in Lock.	102
6.6	Plot of the Two 132 Bit Maximal-length Sequences in Lock.	102
6.7	Plot of the Transmitted Maximal-length Sequence and the Voltage Spike at the Output of the Maximal Likelihood Phase Discriminator.	103
6.8	Circuit Diagram of the Digital Data Modulator.	103
6.9	Circuit Diagram of the Digital Detection Circuitry.	104
6.10	Block Diagram of the Novel Synchronisation Loop Adapted to the Data Environment During Prototyping.	105
6.11	Plot of the Data and Invert Signals.	105
6.12	Plot of the Two Maximal-length Sequences in Lock in the Data Environment.	106
<b>Appendix 2</b>		
2.1	Detailed Circuit Diagram of Prototype Loop.	114

---

# LIST OF TABLES

## Chapter 3

- 3.1 Table of the parameters used in the simulation to determine the phase information generated by the maximum likelihood phase detector. 21

## Chapter 4

- 4.1 Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the feedback path. 27
- 4.2 Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the receiver frontend for the positive locking case. 33
- 4.3 Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the receiver frontend for the negative locking case. 35
- 4.4 Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the input path in a noisy environment. 42
- 4.5 Table of the parameters used in the simulation of the novel synchronisation with a high order input low pass filter in a noisy environment. 49
- 4.6 Table of the parameters used in the simulation of the novel synchronisation adapted to the data environment through edge detection. 54

## Chapter 6

- 6.1 Logic table for the digital data modulator. 104

---

## PREFACE

No man walks alone. From every meeting he takes with him a small part of the other, to travel with him as friend and helper. I would like to thank the following for the excellent company they keep, both when present and not.

To my mother and father, for their love and support. I owe them more than I could ever repay. To my sister, for her love. She will always be an example to me. To my gran, for her love. She proves that one need never be as old as one's years. To Johan, for his friendship and invaluable advice. He proves that the one you work with need not only be an acquaintance. To Heather, for her love. I will never forget the time we spent together. To all my friends, for your friendship. You all ease the load.

I have also received financial support from the following. Dr. R.M.Braun, the U.C.T. Research Council and the F.R.D.

---

# INTRODUCTION.

Spread Spectrum is a digital modulation scheme whereby the conservation of bandwidth principle is sacrificed due to the advantages the scheme offers. These advantages include multiple access and suppression of multi-path interference. This has led to its growing use in commercial applications.

However, the receivers tend to be complex. This is due to the difficulties of synchronising the receiver to the transmitter's spreading waveform. The common methods of synchronisation tend to perform a two fold synchronisation procedure. Firstly a coarse synchronisation is carried out followed by a fine synchronisation in order to accurately track the received waveform. A simpler synchronisation solution would be to perform all synchronisation in a single loop.

Two such novel maximal-length sequence synchronisation loops have been proposed by J.G. van de Groenendaal. This thesis discusses the investigation, through simulation and proto-typing, of these two loops. Figure 1.1 shows the conceptual layout of the text.

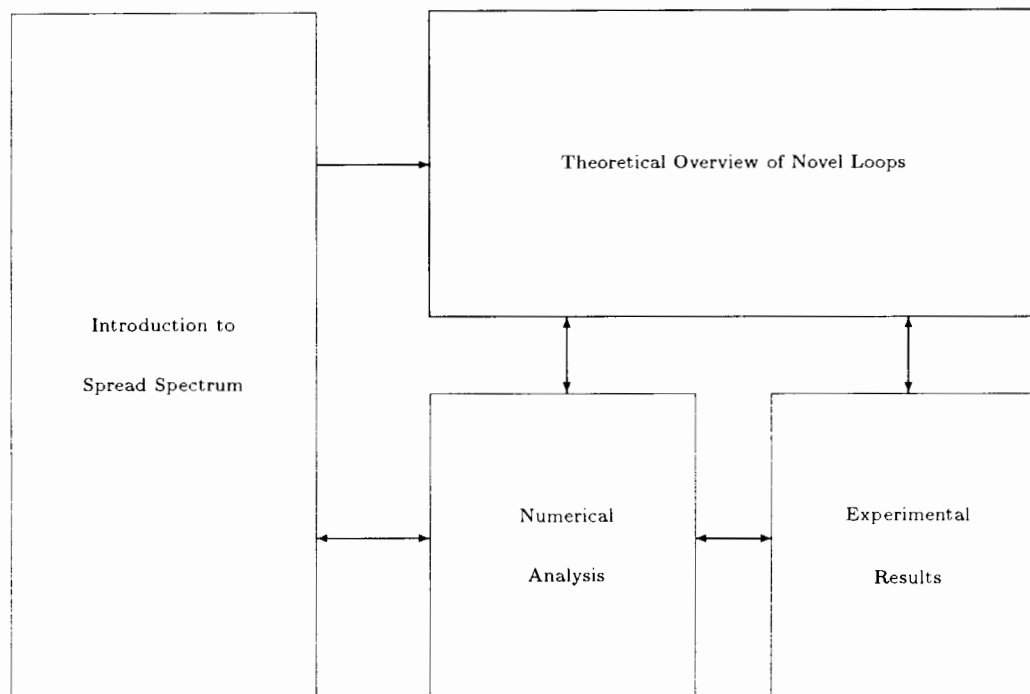
Firstly a background to Spread Spectrum systems is presented. In this chapter the methods of generation of Spread Spectrum, with special reference to the role maximal-length sequences play, as well as its properties are discussed. Common methods of synchronisation are presented. The two pronged approach to maximal-length sequence synchronisation is highlighted.

Secondly a brief overview of the two novel synchronisation loops is presented. Making use of concepts introduced when discussing the background to Spread Spectrum as well as an analysis of the phase information available to the loop, a simple model describing the loop's dynamics and locking mechanism is developed.

Thirdly the simulation of the two loops is presented. The second of the two proposed loops is dealt with in more detail as it is the more stable implementation. The loops noise performance as well as the adaptations that need to be made in order to adapt the loop to the data environment are discussed.

Fourthly experimental results are presented. This is done in two parts. Firstly the different implementations of the various loop components are described. These include the signed edge detector and the synchronous oscillator. Secondly the proto-typing of the two loops is discussed and results are presented.

Lastly conclusions are drawn from the text and recommendations for possible further areas of study are made.



**Figure 1.1** A block diagram of the conceptual layout of this text.

---

# DESCRIPTION OF COMMON SYNCHRONISATION NETWORKS.

Many digital modulation methods are designed to operate effectively in additive white Gaussian noise ( AWGN ) channels while conserving bandwidth. However, not all channels can be accurately modelled as AWGN and, under some conditions, the characteristics of a narrowband signal might not be as desirable as those of a wideband signal. Some applications might require a low probability of detection which a narrow band system cannot provide as the transmitted energy is not spread over a range of frequencies. Jamming signals, both intentional and accidental, cannot be modelled as white Gaussian noise sources. Mobile communication systems may suffer from multi-path interference in areas where more than one transmission path exists each introducing a different amount of phase shift at the receiver. Multiple access of a single resource in a transmission network can reduce costs. These are a few of the considerations that have led to the development of Spread Spectrum systems.

## 2.1 THE GENERATION OF SPREAD SPECTRUM SYSTEMS.

A Spread Spectrum system can be classified as follows:

1. The transmitted signal energy must occupy a bandwidth which is both larger ( often much larger ) and independent of the information bit rate. The bandwidth used is thus greater than the minimum required to accurately transmit the data signal.
2. Demodulation must be accomplished, in part, by correlation of the received signal with the receiver's synchronised replica of the transmitter's spreading waveform.

[17]. A number of methods exist of generating Spread Spectrum waveforms, but the most common ones are generated through:

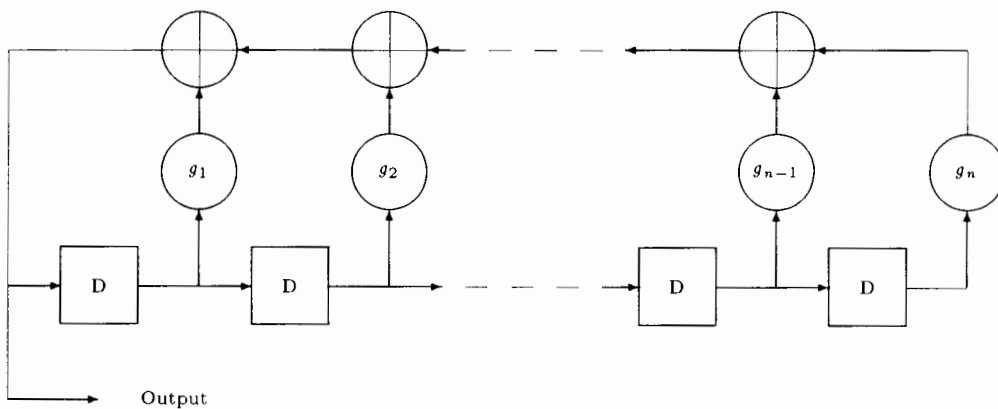
1. Direct modulation ( BPSK for example ) of a carrier by a digital code sequence whose chip rate is much higher than the data signal bandwidth. These systems are referred to as Direct Sequence Spread Spectrum systems.
2. Shifting the carrier frequency in discrete increments in a pattern dictated by a code sequence. The transmitter thus jumps from frequency to frequency within a predetermined set. These systems are referred to as Frequency Hopping Spread Spectrum systems.
3. Sweeping the carrier frequency through a wide band during a given pulse interval. These systems are referred to as Pulsed-FM or Chirp Modulation Spread Spectrum systems.

[6]. This thesis deals only with synchronisation of maximal-length sequences. The synchronisation network could then be further applied in a Direct Sequence Spread Spectrum system.

### 2.1.1 Discussion of the Maximal-length Spreading Codes.

The characteristics of Direct Sequence Spread Spectrum are determined by the digital code sequence used to accomplish the spreading of the data signal. The codes used should have clear maximum in the autocorrelation function when the codes are aligned to allow the receiver to accurately detect a particular code-phase when it has successfully synchronised its replica of the transmitter's spreading waveform. The codes should also have low crosscorrelation values to allow for multiple access and multipath interference suppression. A low crosscorrelation value means that different code-phases do not have much affect on one another when integrated over a period of time. The codes should operate at high speeds to generate wide bandwidths thus effectively spreading the energy of the data signal over a wide bandwidth making detection difficult and have long periods to made detection and regeneration difficult for a jamming device. A group of codes which have these properties are referred to as pseudorandom codes or pn-sequences.

Specific pn-sequences include maximal-length and Gold codes. They are generated using a shift register whose contents during each clock interval is some linear combination of the previous contents of the shift register. In this thesis we will deal only with maximal-length codes. Figure 2.1 shows a method of generating a maximal-length sequence. The correct taps, denoted by  $g_1$  through  $g_n$ , needed to generate a maximal-length code can be found in many texts [17] [6].



**Figure 2.1** Block diagram of a method of generating maximal-length codes. The feedback taps denoted by  $g_1$  through  $g_n$  must be chosen correctly so as to generate a maximal-length sequence.

Maximal-length codes have a number of properties that are useful in their applications to Direct Sequence Spread Spectrum. A few are:

1. A maximal-length sequence has one more 'one' than 'zero'. The number of 'ones' in the sequence is  $\frac{1}{2}(L + 1)$ , where  $L$  is the total number of bits in the sequence.
2. The modulo-2 sum of a maximal-length sequence and any phase shift of the same sequence is another phase of the same maximal-length sequence.
3. If a window of width  $r$ , where  $r$  is the number of registers in the shift register, is slid along the sequence for  $L$  shifts, each  $r$ -tuple except the all 'zero'  $r$ -tuple will appear exactly once.
4. The periodic autocorrelation function  $\omega_m(k)$  is two-valued ( Figure 2.2 ) and is given by

$$\omega_m(k) = \begin{cases} 1 & \text{if } k = zL \\ -1/L & \text{if } k \neq zL \end{cases} \quad (2.1)$$

where  $z$  is an integer and  $L$  is the length of the maximal-length sequence.

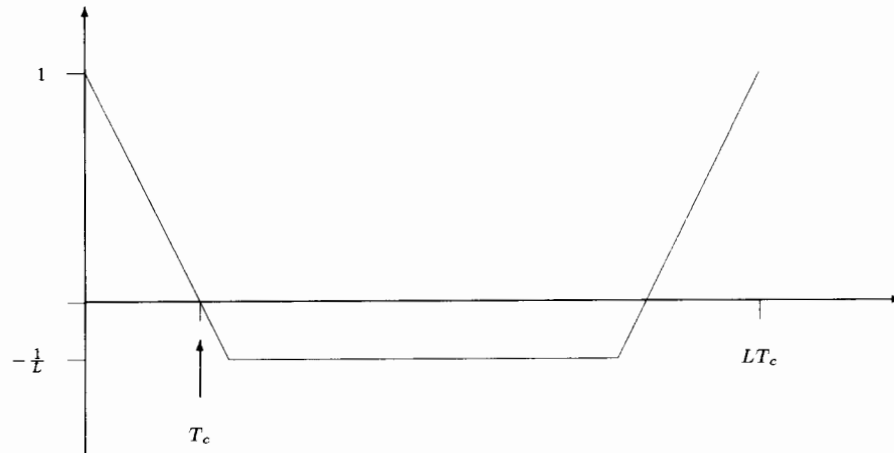


Figure 2.2 Graph showing the two-valued autocorrelation function of a maximal-length code.

Many texts have derived the power spectrum of maximal-length sequences. Only the main characteristics will be highlighted here. The envelope of the power spectrum of a maximal length sequence is given by:

$$\frac{L+1}{L^2} \text{sinc}^2(\omega T_c)$$

where  $L$  is the length of the maximal-length sequence and  $T_c$  is the chipping period. The first null in the power spectrum is therefore at the clocking frequency. This information is therefore not directly available to the receiver.

As maximal-length codes are periodic their Fourier transforms consist of a discrete spectra with the spacing between spectral lines being  $\frac{1}{LT_c}$ . One can therefore determine the length of a maximal-length sequence by counting the spectral lines between the nulls of the envelope. Figure 2.3 shows the power spectrum of a 7 bit pseudorandom sequence. However, in Spread Spectrum systems, as the pseudorandom sequence is modulated by a random data waveform the transmitted waveform will no longer be truly periodic and the spectrum will be continuous.

## 2.2 THE SYNCHRONISATION OF PSEUDORANDOM SEQUENCES.

Due to the autocorrelation property of a maximal-length sequence ( Figure 2.2 and equation 2.1 ) an attempt by the receiver to despread the transmitted waveform with a replica that is offset by more than one chip will not result in enough energy at the output to recover the data. Synchronisation of the receiver to the transmitter is therefore a critical component of any Spread Spectrum system.

A Spread Spectrum receiver has two areas of uncertainty to eliminate to achieve synchronisation. Firstly the correct code-phase of the received signal must be determined to allow for despreading of the waveform. Secondly the correct clocking frequency of the pseudorandom code must be determined to allow for the maintenance of lock.

Causes of uncertainty include:

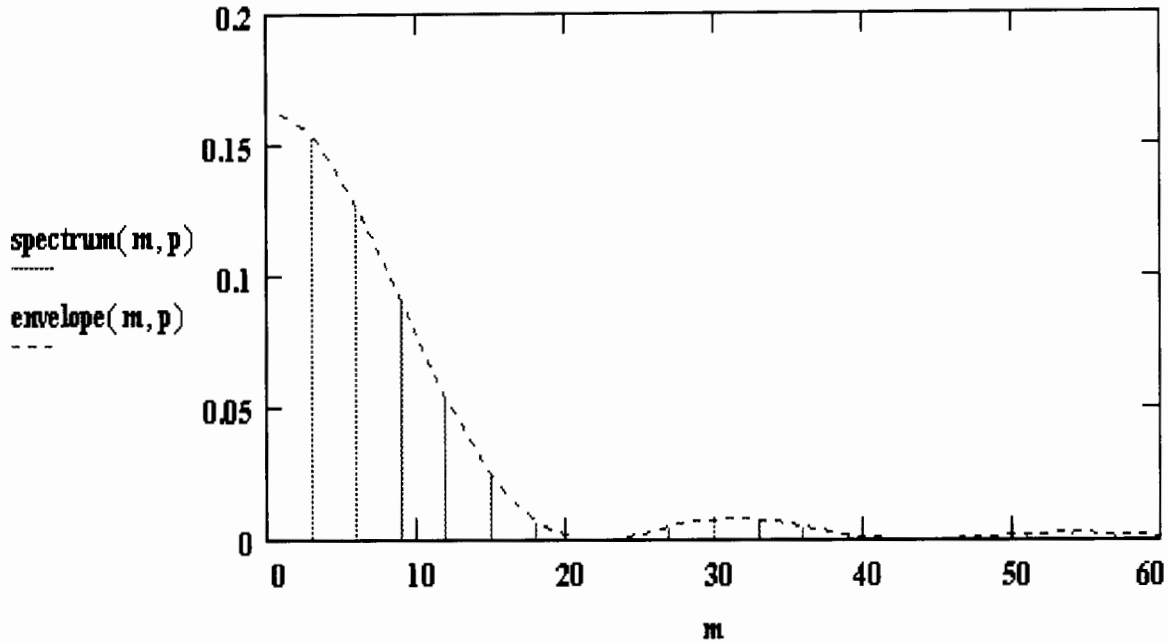


Figure 2.3 Graph of the power spectrum of a 7 bit maximal-length sequence.

1. Propagation delays which could translate into large code-phase uncertainties.
2. Oscillator instabilities in the transmitter as well as the effects of Doppler shift which could affect the perceived clock frequency at the receiver.

As these sources of error are likely to exist in any Spread Spectrum system, the receiver must synchronise to a waveform of undetermined code-phase and clock frequency. The receiver will therefore need to transverse the search grid shown in Figure 2.4 in order to resolve the correct pseudorandom sequence.

A variety of methods of traversing the search grid exist. The majority, however, use the autocorrelation property ( Equation 2.1 ) of maximal-length sequences to detect if acquisition has been achieved. Figure 2.5 shows the block diagram of a code acquisition detection circuit.

Assuming the absence of data modulation of the maximal-length sequence,  $p(t, \omega)$ , the received waveform,  $r(t)$ , may be written as

$$r(t) = p(t - T_d, \omega_{tran}) + n(t)$$

where  $T_d$  is the propagation delay,  $\omega_{tran}$  is the transmitter clocking frequency, and  $n(t)$  is additive white Gaussian noise. The receiver's replica of the spreading waveform may be written as

$$s(t) = p(t - i\Delta t, q\Delta\omega)$$

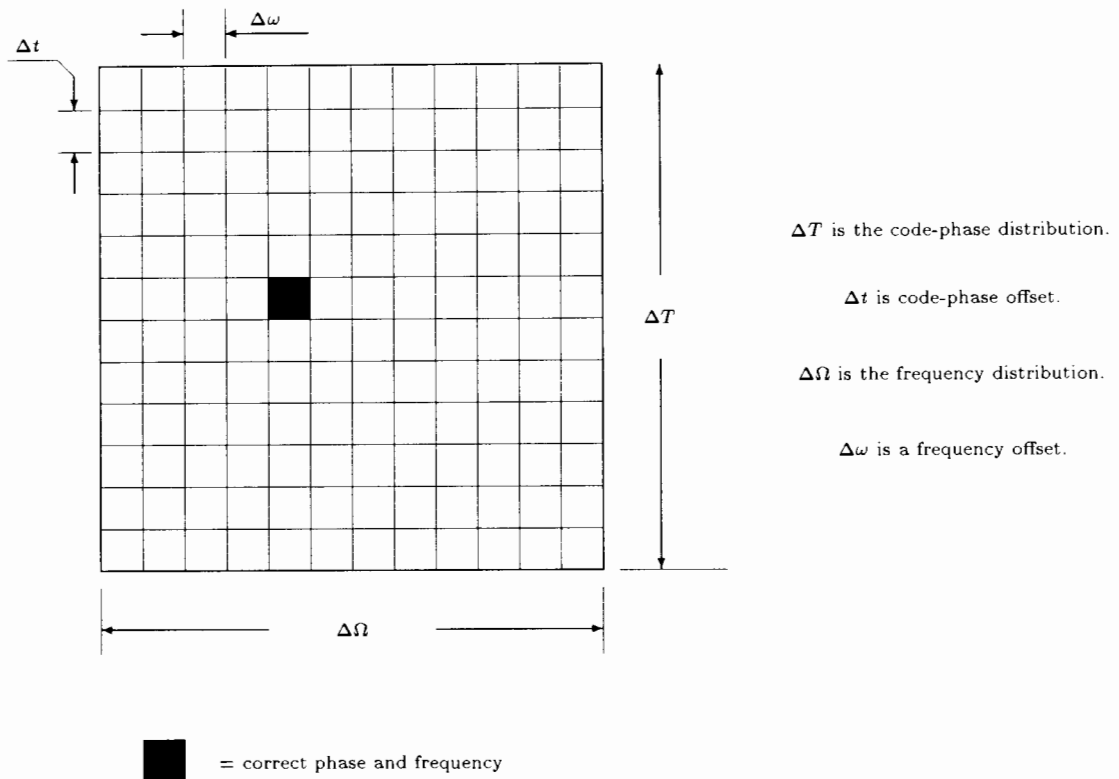
After multiplication and low pass filtering the resulting wave form,  $x(t)$ , can be written as

$$x(t) = p(t - T_d, \omega_{tran})p(t - i\Delta t, q\Delta\omega) + n_{lpf}(t)$$

The energy detector will take the expectation function of  $x(t)$  giving

$$e(t) = E(p(t - T_d, \omega_{tran})p(t - i\Delta t, q\Delta\omega))$$

If the incorrect value of  $q$  is chosen then  $e(t)$  will be the cross correlation of two maximal-length sequences of different frequency. Unless  $\omega_{tran} \approx q\Delta\omega$  this will be a small value and the decision logic would



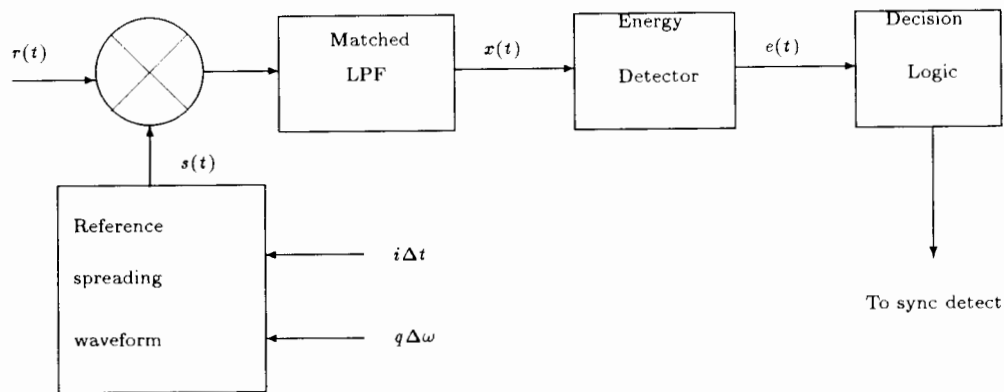
**Figure 2.4** Graph showing the baseband search area of a Spread Spectrum receiver due to uncertainties in code-phase and clock frequency.

not detect synchronisation. Assuming that the correct clock frequency is selected then  $e(t)$  will be the autocorrelation function of the pseudorandom sequence. From figure 2.2 it is clear that unless  $t - T_d \approx t - i\Delta t$  the decision logic will be unable to detect code acquisition.

The synchronisation of maximal-length sequences therefore generally consists of a two pronged approach. As the condition that  $t - T_d \approx t - i\Delta t$  is more critical than  $\omega_{tran} \approx q\Delta\omega$  in determining whether the two sequences are aligned by despreading the input waveform, a coarse search of the baseband Spread Spectrum search grid is executed first to identify the correct code phase or  $t - i\Delta t$  to within one chipping period. This is referred to as code acquisition. After the correct code-phase has been established the correct clock frequency or  $q\Delta\omega$  is then identified and tracked. This is referred to as code tracking [17] [6].

## 2.3 CODE ACQUISITION IN SYNCHRONISATION SYSTEMS.

Code acquisition consists of determining methods of adjusting the receiver's code-phase under varying design constraints so as to reduce locking times as much as possible.



**Figure 2.5** Block diagram of a maximal-length sequence synchronisation detection circuit. As resolving the correct code-phase has more of an effect on the despreading of the input waveform than resolving the correct clocking frequency, circuits of this type are not used in order to track the input code but rather to acquire the input code-phase.

### 2.3.1 Matched Filter Synchronisation Network.

This method determines  $t - i\Delta t$  by using the third property of maximal-length sequences as listed above. A matched filter on the input of the receiver is designed to output a pulse when a particular sequence of code symbols of length  $r$ , where  $r$  is the number of registers in the shift register, is received. As a particular sequence of code symbols of length  $r$  will occur only once in a maximal-length sequence, the pulse will correctly supply the initial condition,  $|t - i\Delta t|$ , needed to the synchronisation circuitry. Although this technique is effective it is not widely used for two reasons.

1. The matched filters need extremely large time-bandwidth products. Such filters are difficult to implement. This constitutes the major drawback of this method.
2. When the spreading code period is long the time taken before the correct sequence of code symbols is detected may be long. This problem can be overcome with programmable filters. Using these filters the propagation delay is estimated and the sequence of code symbols to be detected loaded accordingly.

As this is not a method commonly used it shall not be used for comparison purposes [17].

### 2.3.2 Rapid Acquisition by Sequential Estimation ( RASE ).

At sufficiently high signal-to-noise ratios the probability is high that a received sequence of code symbols will have been estimated correctly. The correct starting code-phase,  $|t - i\Delta t|$ , can therefore be determined directly by loading the receiver's shift register with the received code symbols ( Figure 2.6 ). Once the shift register has been loaded the maximal-length sequence is switched to operate normally. If an incorrect load occurs, synchronisation will not be detected and a new set of code symbols will be estimated and loaded until lock is achieved [17].

Let  $p$  be the probability of correctly estimating a received symbol,  $r$  the number of registers in the shift register,  $T_c$  the chipping period and  $T_e$  the time taken to evaluate the correctness of the load. Ignoring

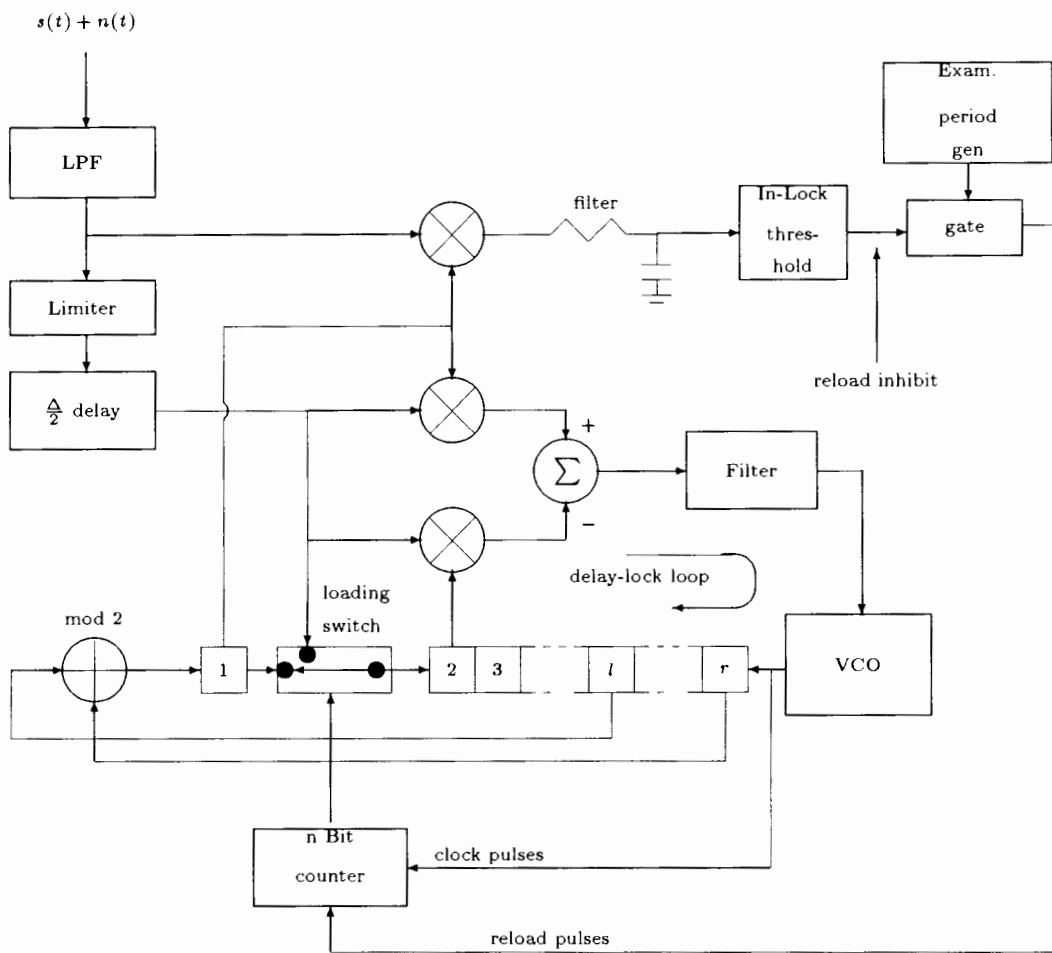


Figure 2.6 Circuit diagram of the Rapid Acquisition by Sequential Estimation method. Note that a full-time early-late code tracking loop has been included. This will be discussed later.

the possibility of false detection of lock the mean synchronisation time for the system is

$$\overline{T_s} = \frac{rT_c + T_e}{p^r} \quad (2.2)$$

Equation 2.2 can be simplified for the two boundary conditions. At high signal-to-noise ratios where the received symbol will be correctly estimated,  $p = 1$ , and the mean synchronisation time is given by

$$\overline{T_s} = rT_c + T_e \quad (2.3)$$

At low signal-to-noise ratios the received symbol is unlikely to be estimated correctly and  $p$  therefore equals 0.5. The mean synchronisation time is then given by

$$\overline{T_s} = 2^r (rT_c + T_e) \quad (2.4)$$

### 2.3.3 Serial Search Techniques.

Serial search techniques are the most common method of code acquisition. This technique involves a systematic search of the all the cells in figure 2.4 [17]. The receiver detects for lock at a certain code-phase. If lock is not detected then the code-phase is advanced by a set amount and the procedure is repeated. This continues until lock is detected.

A great deal of research has gone into the optimal methods of advancing the code-phase to reduce locking times. However, a reasonable prediction of locking time can be obtained by assuming a simple linear time advance,  $\Delta t$ , of the code-phase.

Let  $T_{fa}$  be the time taken to reject a false alarm condition,  $P_{fa}$  be the probability of a false alarm and  $T_e$  be the evaluation time of a cell. Then the dwell time in a false alarm condition,  $T_{da}$ , is given by

$$T_{da} = T_e + T_{fa}P_{fa}$$

Using the above equation the mean synchronisation time is given by

$$\overline{T_m} = \left(\frac{\Delta T}{\Delta t} - 1\right)T_{da}\left(\frac{2 - P_d}{2P_d}\right) + \frac{T_e}{P_d} \quad (2.5)$$

where  $P_d$  is the probability of detection of the correct cell and  $\Delta T$  is the region over which the code-phase is distributed.

Equation 2.5 can be simplified by ignoring the false alarm penalty and assuming that the probability of detecting the correct cell is high. Under these conditions the mean time to synchronisation is given by

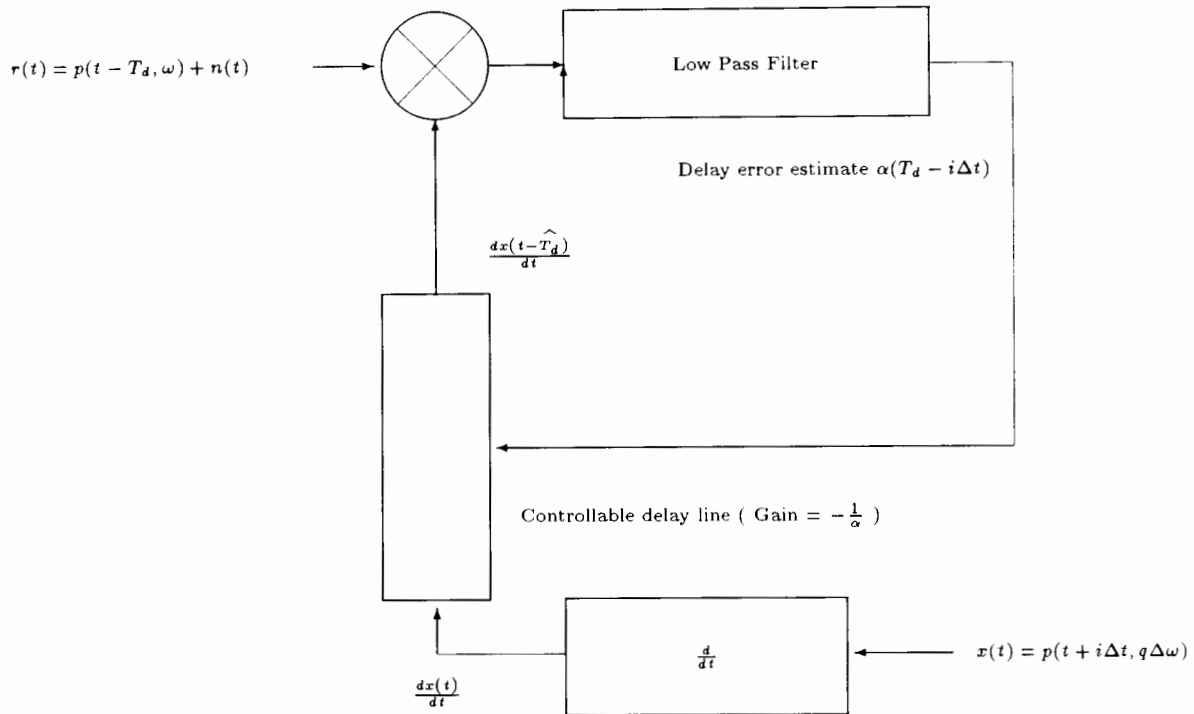
$$\overline{T_m} \approx 2^r T_e \quad (2.6)$$

## 2.4 CODE TRACKING IN SYNCHRONISATION SYSTEMS.

Once the code acquisition network has resolved the received code to within a fraction of a chipping period it is necessary to accurately align the two pseudorandom sequences. This is commonly done using phase-locked loop techniques, the only variation being the implementation of the phase discriminator. Pseudorandom tracking loops are therefore subject to the same constraints as conventional narrowband synchronisation networks. The loop bandwidth needs to be wide so as to facilitate tracking of the changes of the perceived transmitter clock, yet narrow enough not to slip cycles due to tracking jitter. A slipped cycle in a code tracking loop may cause the system to re-enter the code acquisition mode. As a large body of phase-locked loop theory exists concerning these filter considerations, only the implementations of Spread Spectrum phase discriminators will be highlighted in this text.

### 2.4.1 The Optimum Tracking Loop for Wideband Signals.

The optimal phase discriminator for an arbitrary wide band signal is one which forms the product of the received signal plus noise and the first derivative with respect to time of the receiver's replica of the transmitted waveform. A discriminator of this form will give the maximum likelihood estimate of the phase error between the two input signals in a white additive Gaussian noise environment. Spilker and Magill [9] developed a tracking loop based on the maximum likelihood phase discriminator ( Figure 2.7 ).



**Figure 2.7** Block diagram of the optimum maximal-length sequence tracking loop. The controllable delay will be adjusted until the receiver's estimate of the propagation delay  $\widehat{T}_d$  is equal to the delay  $T_d$ .

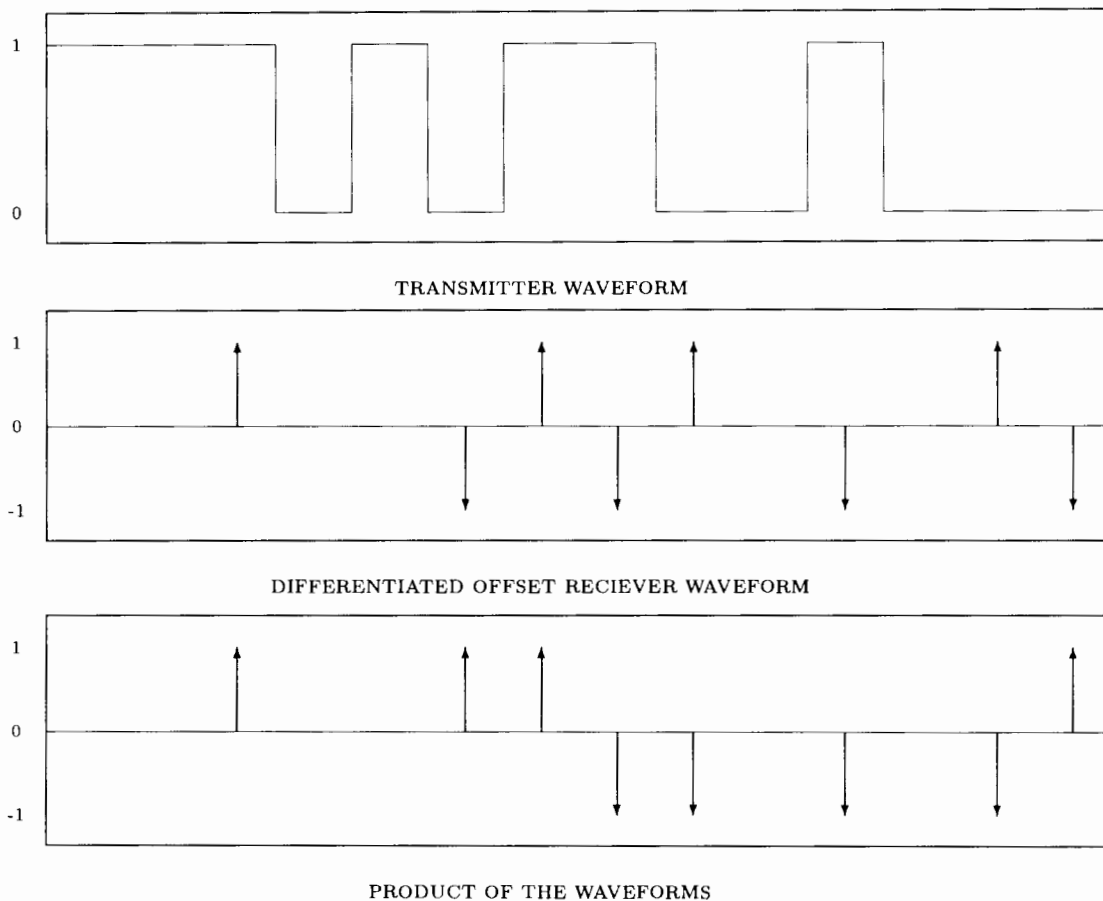
The derivation of the discriminator slope involves a description of the three states the phase discriminator can generate. Figure 2.8 shows the output of the multiplier when the two sequences are more than a chipping period apart. Under these conditions the output of the low pass filter in figure 2.7 will be close to zero. However, when the two sequences are within a chip of each other the spikes due to differentiation of the receiver's maximal-length sequence will multiply either all positive ( Figure 2.9 ) or all negative. The discriminator slope will therefore be a three valued function as shown in figure 2.10.

Multiplying the discriminator slope with the gain of controllable delay line,  $-\frac{1}{\alpha}$ , will result in a stable tracking point at  $T_d - \widehat{T}_d = 0$ . The controllable delay line will therefore have precisely estimated the propagation delay, therefore accurately aligning the two sequences [17].

Although optimal this method is not widely used of two reasons:

1. While the spike are convenient mathematical constructs they can be difficult to handle electronically.
2. Any data modulation of the input maximal-length sequence would invert the discriminator slope therefore driving the loop incorrectly and forcing the loss of lock.

However, the optimal loop remains instructive while considering other phase discriminators as well as the novel synchronisation network to be described in this thesis.



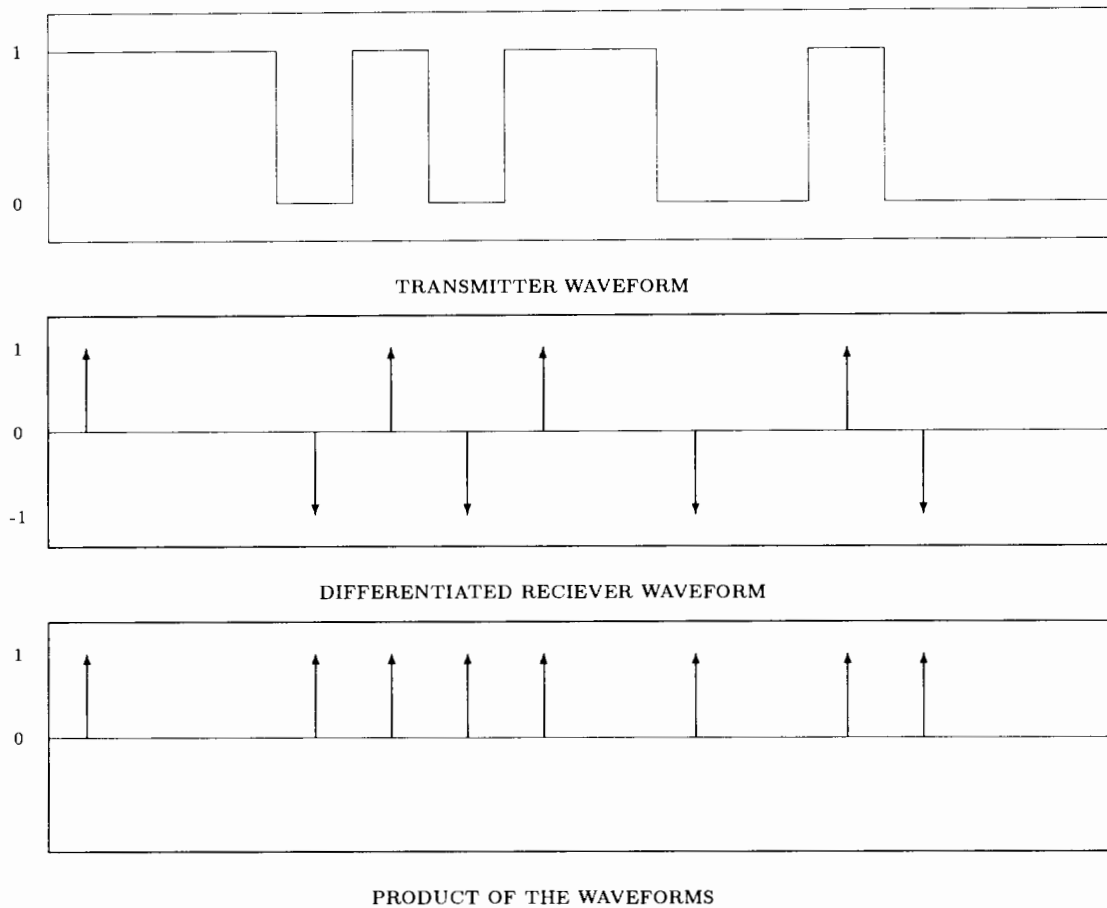
**Figure 2.8** Graph of the waveforms generated by the optimal tracking loop when the sequences are more than a chip apart. The low pass filtered output of the product of the two waveforms will be a small value.

## 2.4.2 The Baseband Full-Time Early-Late Tracking Loop.

The correlation value of pseudorandom sequences is a maximum when the sequences are exactly aligned ( Figure 2.2 ). A phase detector can therefore be designed by comparing the correlation values of a sequence slightly leading and a sequence slightly lagging the receiver's replica of the spreading waveform to detect when one is precisely straddling this maximum. A system based on this form of discriminator is the full-time early-late tracking loop ( Figure 2.11 ).

The full-time early-late tracking loop consists of two correlator arms. One arm to correlate the received signal,  $s_r(t)$ , with the maximal-length sequence leading the receiver's replica of the spreading waveform,  $c_e(t)$ , to generate the output,  $y_e(t)$ . The other to correlate the received signal,  $s_r(t)$ , with the maximal-length sequence lagging the receiver's replica of the spreading waveform,  $c_l(t)$ , to generate the output,  $y_l(t)$ . The early arm,  $y_e(t)$ , is then subtracted from the late arm,  $y_l(t)$ , filtered and used to drive a voltage controlled oscillator.

The phase discriminator slopes can be derived in terms of  $\delta = (T_d - \widehat{T}_d)/T_c$ , the code-phase offset, and  $\Delta$ , the normalised time difference between the early and late discriminator arms. ( Figure 2.12 ). The loop is designed to operate in the linear part of the discriminator curve around  $\delta = 0$ , however the choice of  $\Delta$  is a design criterion.



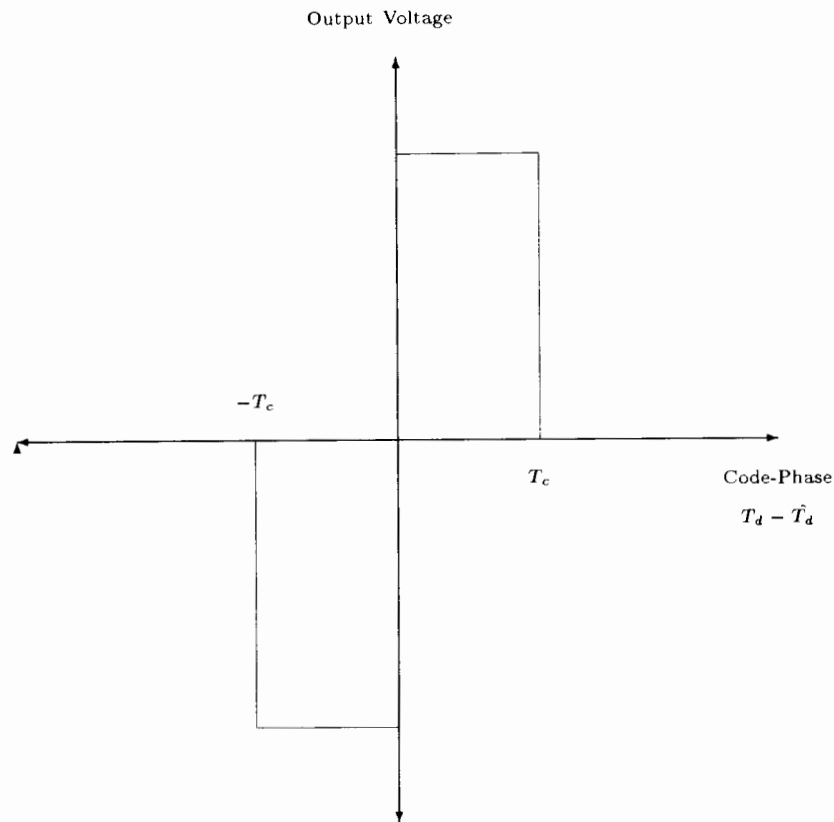
**Figure 2.9** Graph of the waveforms generated by the optimal tracking loop when the sequences are within a chip. The low pass filtered output of the product of the two waveforms will contain a D.C. value.

The loop does however have two constraints :

1. The loop does not operate very well at low signal-to-noise ratios as the spreading waveform must be demodulated down to baseband before tracking. This entails accurately resolving the radio frequency carrier which is difficult at low signal-to-noise ratios.
2. Any data modulation of the input maximal-length sequence would invert the discriminator slopes therefore driving the loop incorrectly and forcing the loss of lock [17].

### 2.4.3 The Full-Time Early-Late Noncoherent Tracking Loop.

The disadvantages of the full-time early-late tracking loop can be overcome by slight variations to the phase discriminator. The need to accurately resolve the carrier can be removed by making the loop noncoherent. The sensitivity to data modulation can be removed by squaring the modulated waveform therefore removing any phase ambiguity. A loop based on this modified phase discriminator is called a full-time early-late noncoherent tracking loop ( Figure 2.13 ).



**Figure 2.10** Graph of the discriminator slope generated by the maximum likelihood phase detector. The controllable delay line will be driven to the stable point at zero.

The full-time early-late noncoherent tracking loop consists of two correlator arms. One arm to correlate the received signal,  $s_r(t)$ , with the maximal-length sequence leading the receiver's replica of the spreading waveform modulated up to approximately the transmitter's carrier frequency,  $a_e(t)$ , to generate the output,  $y_e(t)$ . The other to correlate the received signal,  $s_r(t)$ , with the maximal-length sequence lagging the receiver's replica of the spreading waveform modulated up to approximately the transmitter's carrier frequency,  $c_l(t)$ , to generate the output,  $y_l(t)$ . Both  $y_e(t)$  and  $y_l(t)$  are squared to remove any uncertainty introduced by the data modulation. The resulting signals are filtered and subtracted from one another. This subtraction while providing a signal suitable to drive the voltage controlled oscillator will also remove the error due to the local oscillator not being accurately synchronised to the transmitter's carrier frequency. As the arms are balanced any error due to the carrier being unsynchronised will be equal in both paths. The resulting signal is filtered and used to drive a voltage controlled oscillator.

The phase discriminator slopes for the full-time early-late noncoherent can again be derived in terms of  $\delta$  and  $\Delta$  ( Figure 2.14 ). The slopes differ from those of the full-time early-late tracking loop in that as  $\Delta$  approaches 2 the slope of the curve around  $\delta = 0$  tends to zero. As the tracking loop is designed to drive  $\delta$  to zero this flattening out of the slope can lead to a degradation of the performance characteristics of the tracking loop. The designer must therefore choose a value of  $\Delta$  less than 2.

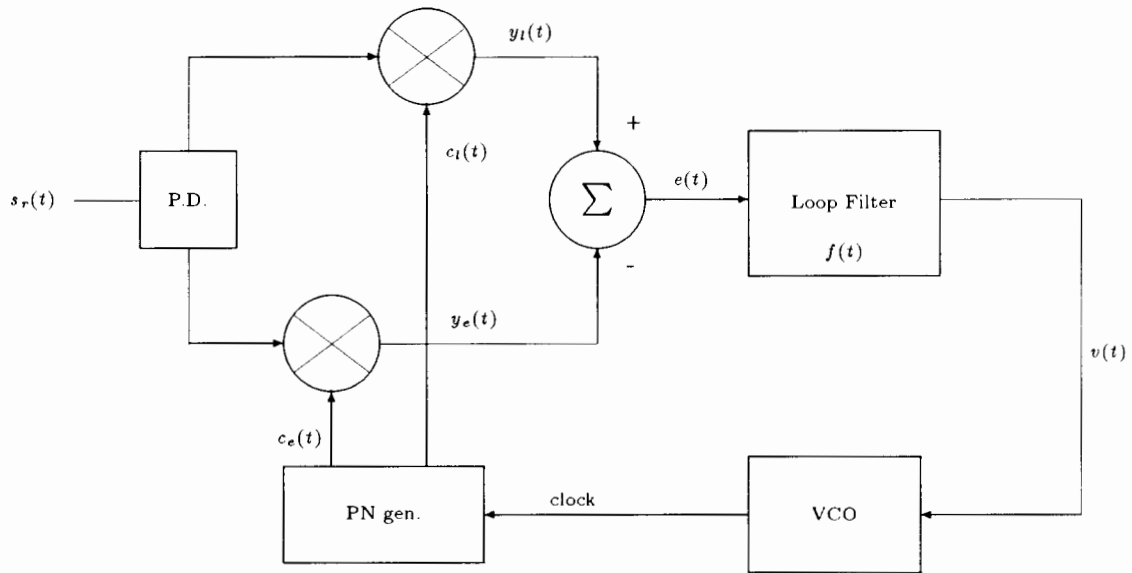


Figure 2.11 Block diagram of the full-time early-late tracking loop.

$$R_{cc}(\delta, \Delta) := R_{cc1}(\delta, \Delta) - R_{cc2}(\delta, \Delta)$$

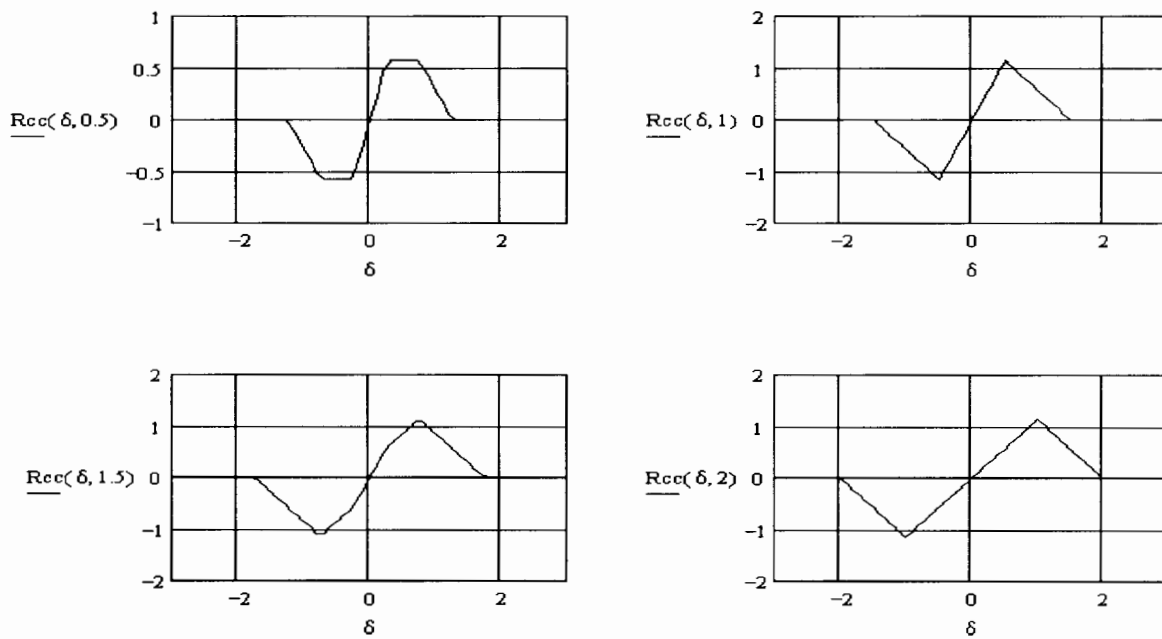


Figure 2.12 Graphs of the phase discriminator slopes of the full-time early-late tracking loop. In the above plots  $\delta$  is kept constant. Only  $\Delta$ , the time difference between the early and late receiver's replica of the spreading waveform, is altered.

The full-time early-late noncoherent tracking loop, while theoretically satisfactory, suffers from two practical considerations :

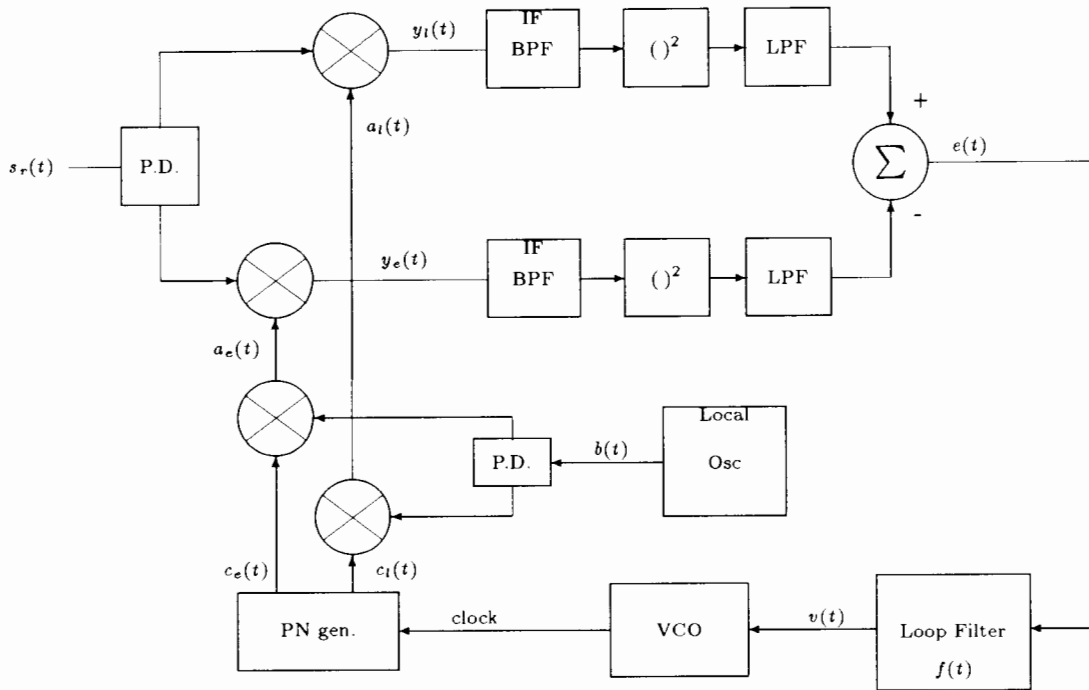


Figure 2.13 Block diagram of the full-time early-late noncoherent tracking loop.

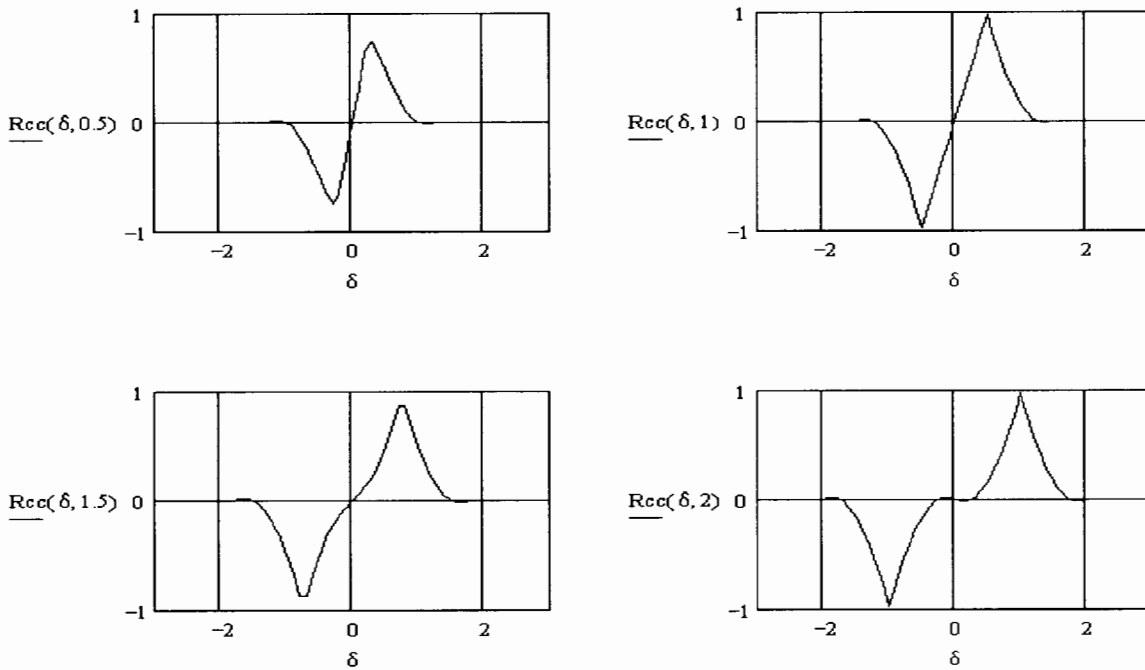


Figure 2.14 Graphs of the phase discriminator slopes of the full-time early-late noncoherent tracking loop. Only  $\Delta$ , the time difference between the early and late receiver's replica of the spreading waveform, is altered.

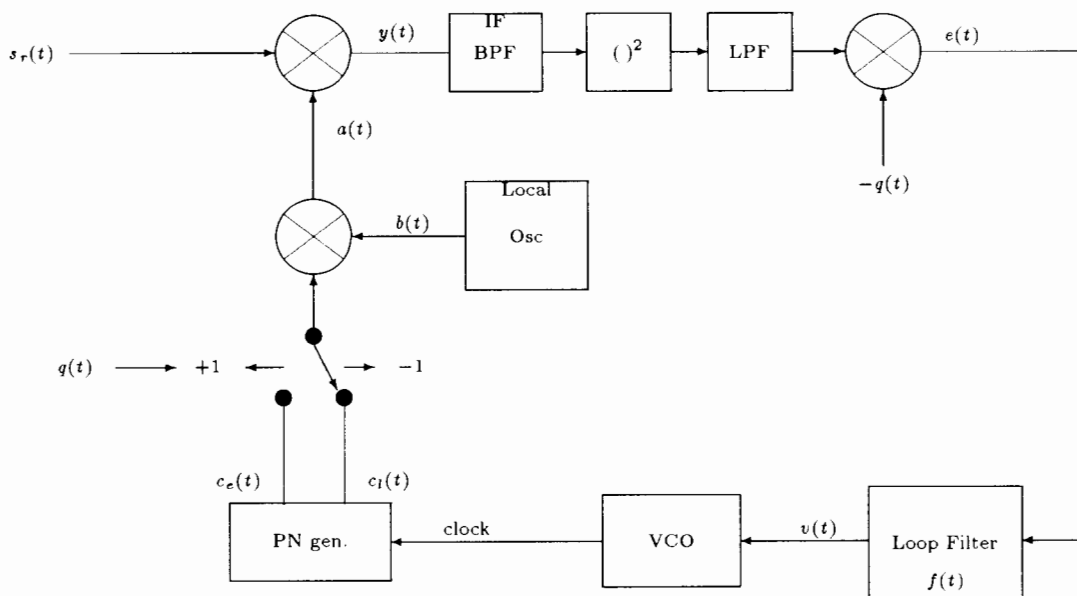
1. The loop requires that the two correlation arms be accurately balanced. If they are not the errors due to the unsynchronised carrier as well as the amplitude imbalances of the correct

output will offset the discriminator curves. If this is the case the stable  $\delta = 0$  point will no longer occur when the sequences are accurately aligned. The loop will therefore not track accurately.

2. The components of the loop are expensive. They are furthermore made more so by the need for them to be balanced [17].

#### 2.4.4 The Tau-Dither Early-Late Noncoherent Tracking Loop.

The tau-dither early-late noncoherent tracking loop is a variation on the full-time early-late noncoherent tracking loop ( Figure 2.13 ). To overcome the problem of imbalances in the two correlator arms, as well as the cost of the components, a single correlator path is used. The arm is switched between an early pseudorandom and a late pseudorandom sequence ( Figure 2.15 ).



**Figure 2.15** Block diagram of the tau-dither early-late noncoherent tracking loop.

The phase discriminator curves are the same as those for the full-time early-late noncoherent tracking loop ( Figure 2.14 ). However, as a single correlator arm is used, amplitude imbalances between the two correlator arms cannot cause a shift in the phase discriminator curves. The curves can be shifted however if the switching waveform,  $q(t)$ , does not equally divide the integration time between the early and the late maximal-length sequences.

The tau-dither early-late tracking loop is not as effective as a full-time ( unswitched ) loop in the presence of noise as the disturbances on the one arm are not subtracted from the same disturbances on the other arm. A variation to overcome this and the imbalance problem is to use the switching waveform  $q(t)$  to switch between each arm. This does however result in a more complex and expensive loop [17].

## **2.5 A SYSTEM LEVEL OVERVIEW OF THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION NETWORK.**

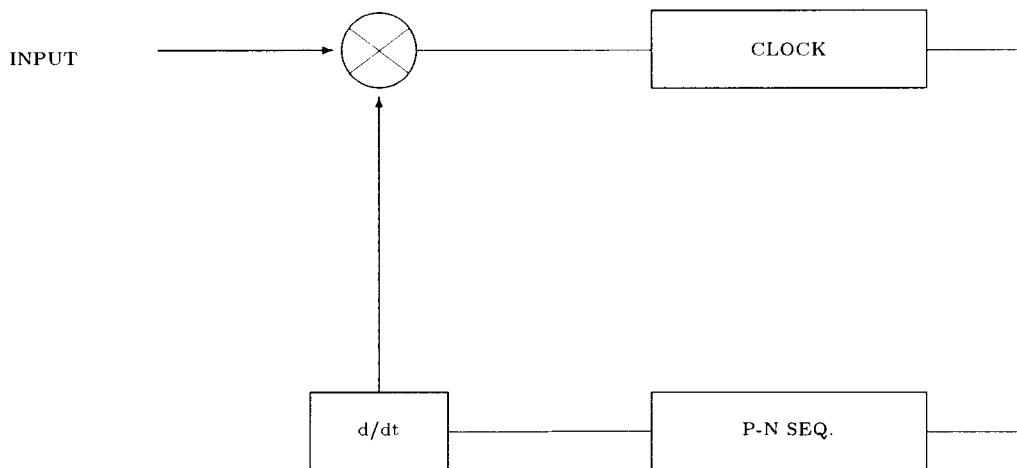
Maximal-length sequences may be characterised by two variables, the initial code-phase and the clock frequency. Synchronisation of two sequences therefore requires the resolving of both variables. Common techniques therefore divide the synchronisation procedure into two distinct parts namely code-phase acquisition and code tracking. Various circuits have been specifically designed to accomplish each task under varying conditions of signal-to-noise ratio. However the division of pseudorandom sequence synchronisation leads to a certain complexity in the synchronisation circuits and a corresponding increase in cost.

J.G. van de Groenendaal, as part of his doctoral dissertation, has developed two synchronisation networks that do code-phase acquisition and code tracking in a single implementation. It was hoped that the combining of the two operations could result in a robust and cost effective synchronisation network. The remainder of this thesis deals with the simulation and prototyping of these two networks.

---

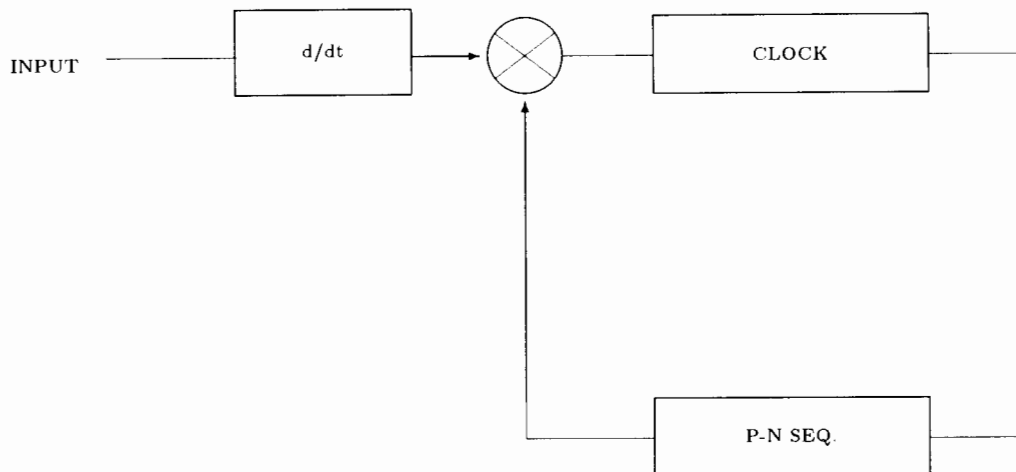
## OVERVIEW OF THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION NETWORK.

As the definition of Spread Spectrum requires that demodulation of the data must be accomplished, at least in part, by correlating the transmitted sequence by the receiver's replica of the transmitted waveform, synchronisation of the spreading waveforms is a fundamental aspect of Spread Spectrum. However, common methods of achieving synchronisation are complex. This might make a Spread Spectrum system unattractive even though from a design point of view it is a good solution. J.G. van de Groenendaal [13] [16] has proposed two baseband synchronisation networks that are simple ( Figures 3.1 and 3.2 ).



**Figure 3.1** Block diagram of the novel maximal-length sequence synchronisation network with the differentiator in the feedback path.

The loops are reminiscent of the optimal code tracking loop discussed in section 2.4.1 in that they are based on variations of the maximal likelihood phase discriminator. In figure 3.1 the transmitted waveform is multiplied by the first derivative with respect to time of the receiver's replica of the spreading waveform. The resultant waveform, however, is not integrated with respect to time over the length of a sequence nor is it used to drive a controllable delay line but rather an adjustable clock. In figure 3.2 the first derivative with respect to time of the received waveform is multiplied by the receiver's replica of the spreading waveform. The resultant waveform is again used to alter the frequency of an adjustable clock.



**Figure 3.2** Block diagram of the novel maximal-length sequence synchronisation network with the differentiator in the receiver frontend.

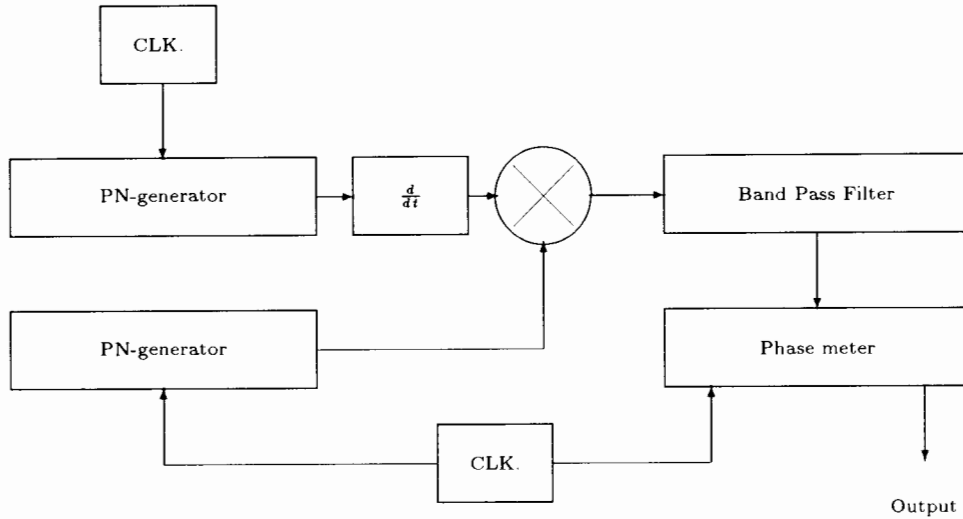
As the novel loops are non-linear, analysis is difficult ( See J.G. van de Groenendaal's dissertation [15] for a detailed mathematical treatment. ) This thesis will present the results of simulating the loops, using the TESLA<sup>1</sup> environment, as well the results from prototype loops built on the bench.

### 3.1 PHASE INFORMATION GENERATED IN THE MAXIMUM LIKELIHOOD PHASE DISCRIMINATOR.

Both novel synchronisation loops are based on variations of the maximum likelihood phase discriminator. This discriminator is formed by taking the product of the first derivative with respect to time of a reference waveform and a second waveform. In the optimum tracking loop, which was also based on the maximum likelihood phase detector, ( Section 2.4.1 ) not all the information generated by the phase discriminator was utilised. The optimum tracking loop only detected for the D.C. level which occurs when the two maximal-length sequences are less than a chipping period apart ( Figure 2.9 ). The case depicted in figure 2.8 was discarded and synchronisation under this condition was delegated to the code acquisition loop. Yet the phase discriminator contains frequency components that could drive the loop out of its present incorrect state in the search grid ( Figure 2.4 ) into another possibly correct state.

To determine the frequency information generated by the maximum likelihood phase discriminator the block diagram depicted in figure 3.3 was simulated in TESLA. The system consists of two independent clocks. The input clock drives a fifteen bit pseudorandom sequence, the output of which is differentiated. The other offset clock drives a second fifteen bit maximal-length sequence generator, the code-phase of which is adjustable, as well as a TESLA phase meter model. The output of the differentiator and the adjustable pseudorandom sequence generator are multiplied together. The output of the multiplier is band pass filtered. The band pass filter drives the phase meter which takes a coherent measurement of the phase difference between its input and the clock driving the adjustable pn-sequence generator. The system parameters are given in Table 3.1 and the results in figures 3.4 through 3.6.

<sup>1</sup> © 1990-1993, by TESOFTE, Inc.



**Figure 3.3** Block diagram of the system used to investigate the maximum likelihood phase discriminator. As phase is the integral of frequency the phase meter will give an indication of the frequency component present at the output of the band pass filter.

Parameter	Case One	Case Two	Case Three
Sequence Length	15	15	15
Centre of Band Pass Filter	1 kHz	1 kHz	1 kHz
Bandwidth	200 Hz	200 Hz	200 Hz
Input Frequency	1 KHz	1 KHz	1 KHz
Frequency Offset	5 Hz	5 Hz	0 Hz
Code-phase Offset	0 bits	9 bits	0 bits

**Table 3.1** Table of the parameters used in the simulation to determine the phase information generated by the maximum likelihood phase detector.

As the output of the phase meter would form the input to the adjustable clock in the novel synchronisation loops it gives an indication of the network dynamics. Figures 3.4 through 3.6 highlight the three synchronisation conditions namely:

1. **THE TRACKING CONDITION.** In figure 3.4 the two sequences are aligned to within a chipping period, as evidenced by the unipolar positive-going spikes, but offset slightly in frequency. The output phase is not constant under this condition which means that this cell in the baseband Spread Spectrum search grid is not a stable point. The novel synchronisation network would thus alter its frequency and enter another cell.
2. **THE CODE ACQUISITION CONDITION.** In figure 3.5 the two sequences are offset in both code-phase and frequency. The output phase changes regularly indicating that this is not a stable condition. The synchronisation network will alter its frequency and enter another cell in the baseband Spread Spectrum search grid.
3. **THE SYNCHRONISED CONDITION.** In figure 3.6 the two sequences are aligned in code-phase and frequency. The output phase remains constant. This is thus a stable condition and the loop would not alter its parameters. Lock is thus achieved.

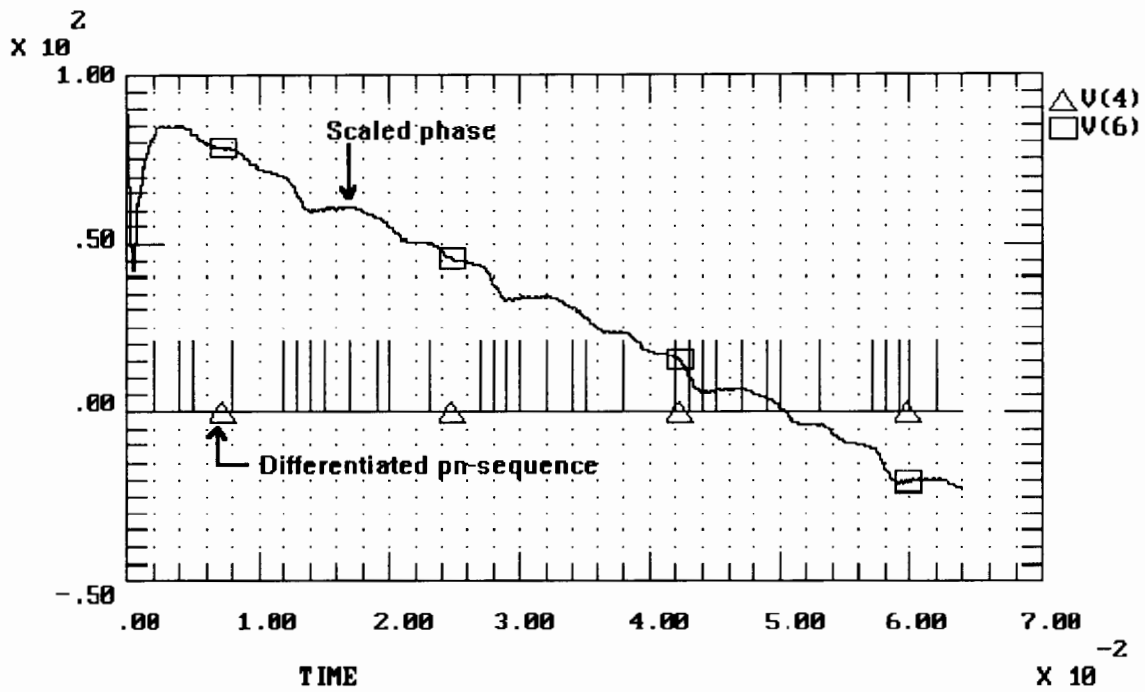


Figure 3.4 Graph of the output of the maximum likelihood phase discriminator and the resultant phase measurement for the parameters given in Table 3.1, Case One.

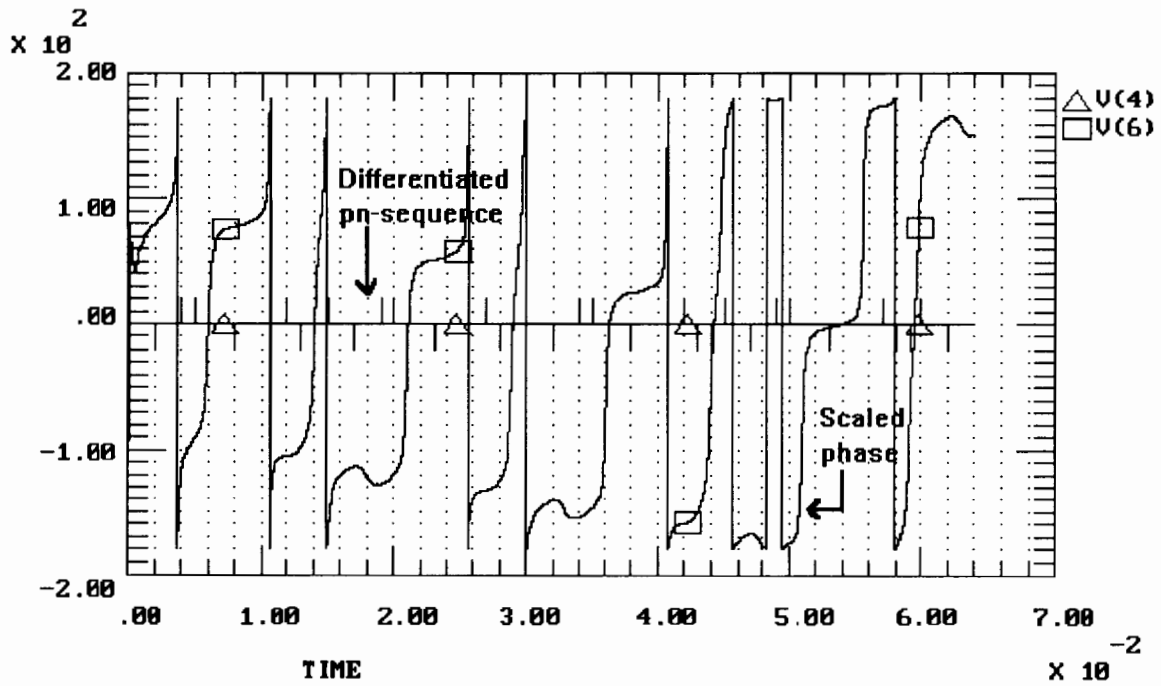


Figure 3.5 Graph of the output of the maximum likelihood phase discriminator and the resultant phase measurement for the parameters given in Table 3.1, Case Two.

The novel synchronisation loops should therefore traverse the search grid by moving from unstable cells until lock is achieved.

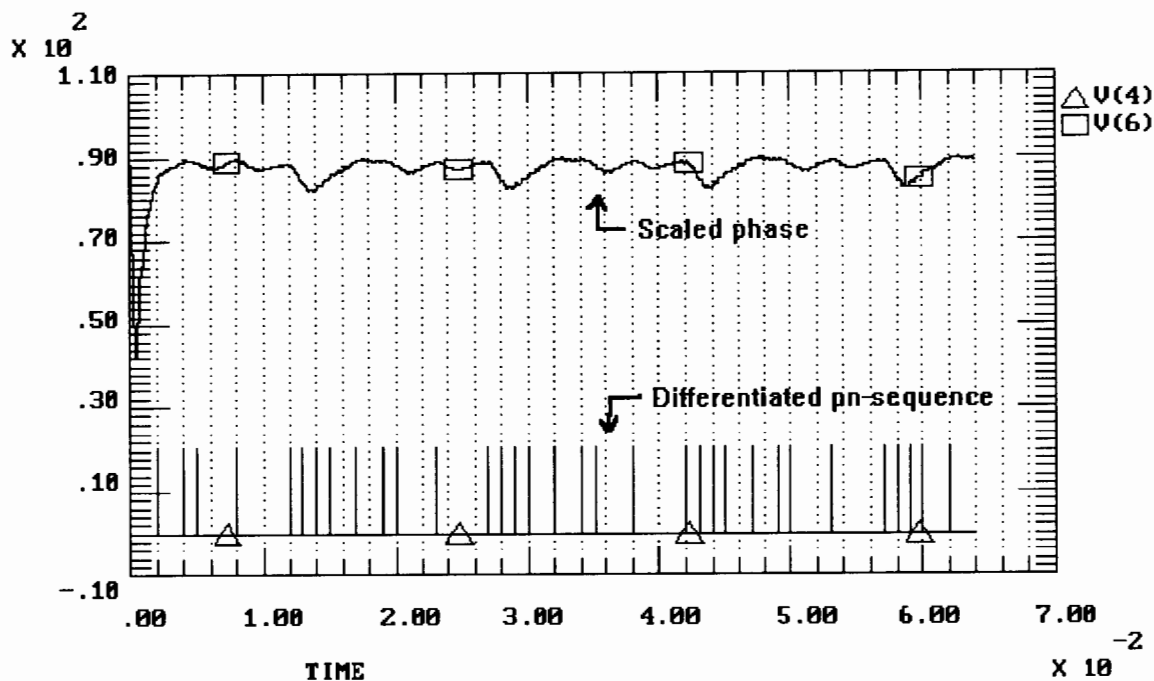


Figure 3.6 Graph of the output of the maximum likelihood phase discriminator and the resultant phase measurement for the parameters given in Table 3.1, Case Three.

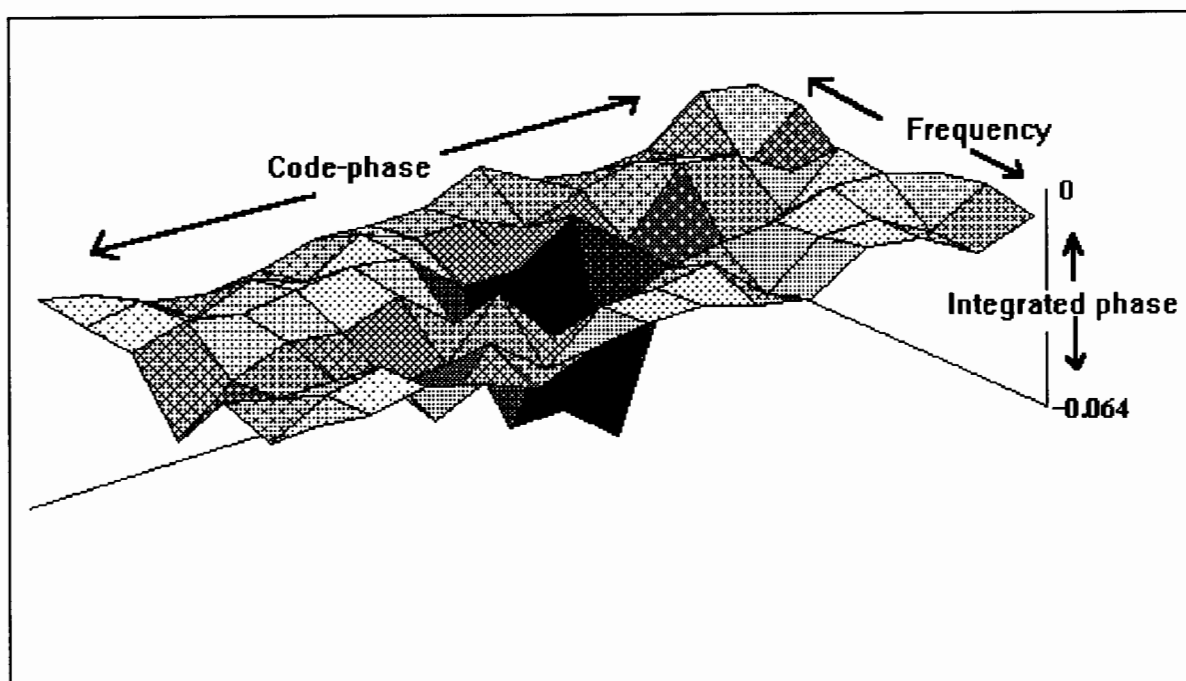
### 3.2 CHARACTERISATION OF THE BASEBAND SPREAD SPECTRUM SEARCH GRID BY INTEGRATED PHASE.

A stable cell in the baseband Spread Spectrum search grid may be defined as one where the phase measurement as generated using the method in figure 3.3 is a constant. As the integration of a constant will yield a higher value than that of the integration of a varying waveform, providing the limits on the amplitude are the same, the integrating of the phase output over a certain number of maximal-length sequences could give an indication of the stability of that cell. Characterising the baseband Spread Spectrum search grid in this manner could thus give insight into the manner in which the novel synchronisation loops traverse the grid.

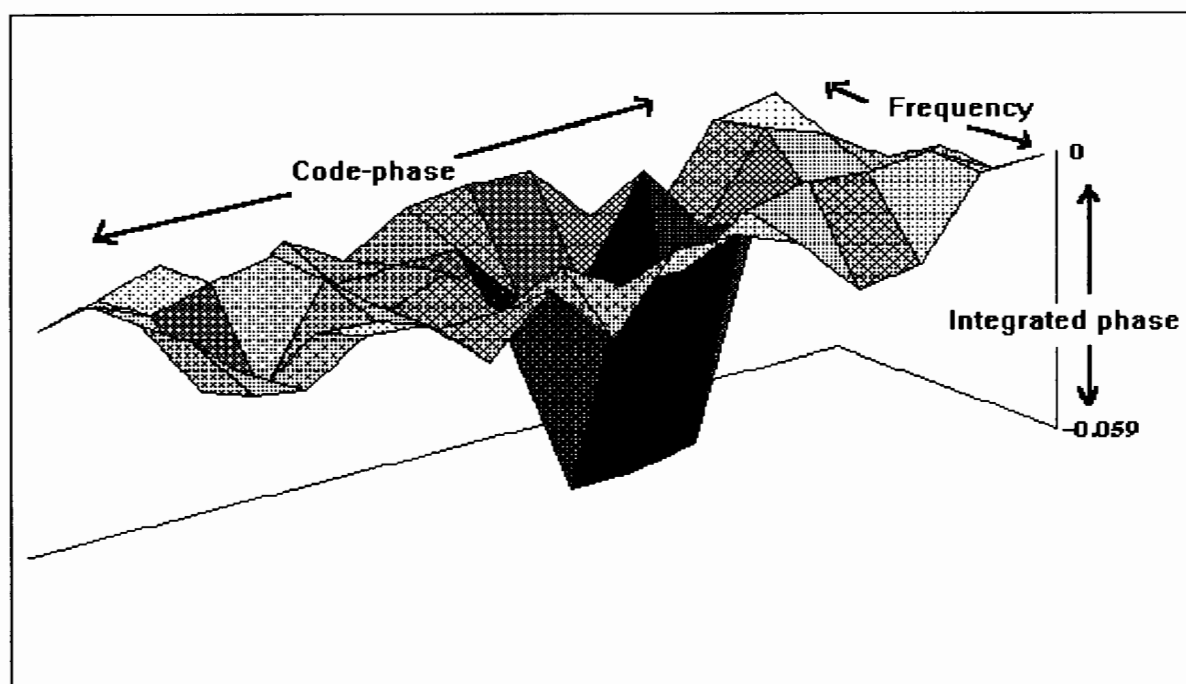
Figures 3.7 and 3.8 characterise the baseband Spread Spectrum search grid in terms of integrated phase. The  $x$  and  $y$  axes denote frequency and code-phase offsets while the  $z$  axis is the negative absolute value of the phase integrated over four maximal-length sequences. Using the negative absolute value means that a hollow or dip in the search grid indicates a stable position. Both figures contain a stable point. This point occurs at the input frequency at either a zero or a one bit code-phase offset. These points correspond to either an all positive-going spike lock condition or an all negative-going lock condition.

The existence of two stable points is a disadvantage of the novel loops in that the number of available code-phases is halved. However, as the points are well defined they can be easily detected and corrected by introducing a shift register before correlation. This does not make the loop more complex.

Figures 3.7 and 3.8 also allow for an heuristic explanation of the locking procedure. Let the present conditions of the loop be viewed as a ball resting on the integrated phase surface. This 'ball' would be under the influence of perturbations caused by the pseudorandom sequences multiplying together. Under these disturbances the 'ball' will roll around the surface until it rolled into the dip that represents



**Figure 3.7** Graph of the baseband Spread Spectrum search grid characterised by integrated phase. The input frequency was at the center frequency of the band pass filter.



**Figure 3.8** Graph of the baseband Spread Spectrum search grid characterised by integrated phase. The input frequency was offset from the center frequency of the band pass filter.

a stable point. The 'ball' is unable to exit the hollow and lock is achieved. This approach, while simplistic, describes the locking procedure quite well.

J.G. van de Groenendaal has shown that the loop displays chaotic behavior when achieving lock. This means that a slight change in the initial conditions of the loop result in very different locking paths across the baseband Spread Spectrum search grid. This is the type of path the 'ball' analogy would suggest.

### **3.3 METHODS OF DETECTING LOCK IN THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION LOOP.**

For the purposes of simulating the two novel synchronisation networks it is important to be able to detect whether or not the network has in fact performed synchronisation. The three conditions that are considered adequate for the purposes of this thesis are :

1. **A LONG RUN OF UNIPOLAR SPIKES.** As discussed in section 2.4.1 a run of either all positive or all negative-going spike indicates that the two maximal-length sequences are aligned in code-phase. However, figure 3.4 had a run of unipolar spikes that resulted in only a slowly varying phase. This means that this unstable state will not be exited quickly by the novel synchronisation network. A run of unipolar spikes for seven or eighth maximal-length sequences will be considered sufficient to indicate that lock has in fact been achieved.
2. **A CONSTANT D.C. LEVEL IN THE ADJUSTABLE CLOCK'S PHASE DISCRIMINATOR.** As figure 3.6 a constant level will only be detected under the synchronisation condition. A constant level in the adjustable clock's phase discriminator thus corresponds to the stable lock condition.
3. **DESPREADING OF THE RECEIVED SIGNAL BY THE RECEIVER.** Due to the auto-correlation property of maximal-length sequences ( Figure 2.2 ) the received signal will only be despread if the receiver's replica of the spreading waveform is accurately aligned to the transmitted waveform.

These conditions will be used in the simulations of the loop to determine locking times and noise performance.

---

# RESULTS OF THE SIMULATION OF THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION NETWORKS.

Simulation has become a valuable engineering tool. It allows for quick and easy alteration of design parameters therefore giving insight into a system. The results can be accurate providing certain conditions are met. They include:

1. **ACCURATE MODELS.** The simulation package used should be accurate enough to generate the effects one would expect on the bench. TESLA is widely used and considered a good systems level simulation package.
2. **SAMPLING RATE.** Any simulation, in order to produce intelligible results, must sample at the Nyquist rate or higher. The Nyquist rate for a bandlimited signal is between two and four times the bandwidth. For a given bandwidth, this minimum rate requirement for a bandpass signal approaches the limit of twice the bandwidth as the centre frequency of the signal increases [10]. If the system is not sampled at this rate spectral fold over will occur, thereby aliasing the data.

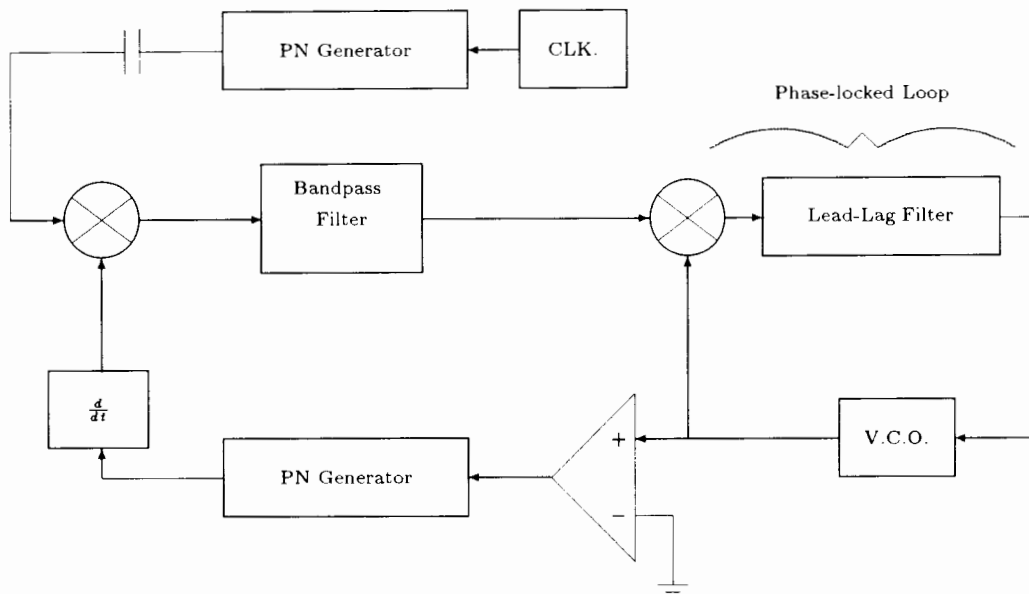
The simulations were performed in co-operation with J.G. van de Groenendaal [5].

## 4.1 SIMULATION OF LOOP WITH THE DIFFERENTIATOR IN THE FEEDBACK PATH.

This is the loop closest in design to the optimal tracking loop discussed in section 2.4.1. The received waveform is multiplied with the first derivative with respect to time of the receiver's replica of the transmitted maximal-length sequence ( Figure 3.1 ). The loop must, however, be adjusted to the TESLA environment by implementing the various components with TESLA models.

Figure 4.1 depicts the simulated loop. The loop input consists of an unmodulated fifteen bit maximal-length sequence of adjustable code-phase. The capacitor allows the code to swing positive and negative so as not to multiply by zero. This input is multiplied by the first derivative with respect to time of the receiver's replica of the transmitted pseudo-noise sequence. The resultant waveform stimulates the adjustable clock. In the case of this simulation this is modelled as a phase-locked loop preceded by a second order Butterworth bandpass filter. The sine-wave output of the voltage controlled oscillator ( V.C.O. ) contained in the phase-locked loop is hard-limited so as to drive the receiver's maximal-length sequence generator.

The implementation of the adjustable clock as a phase-locked loop preceded by a bandpass filter raises some design criterion. A phase-locked loop's acquisition time and noise bandwidths are fundamentally linked by the choice of loop filter. Designing a phase-locked loop which will respond rapidly to changes in the input by using a wideband loop filter could leave the loop susceptible to errors due to noise.



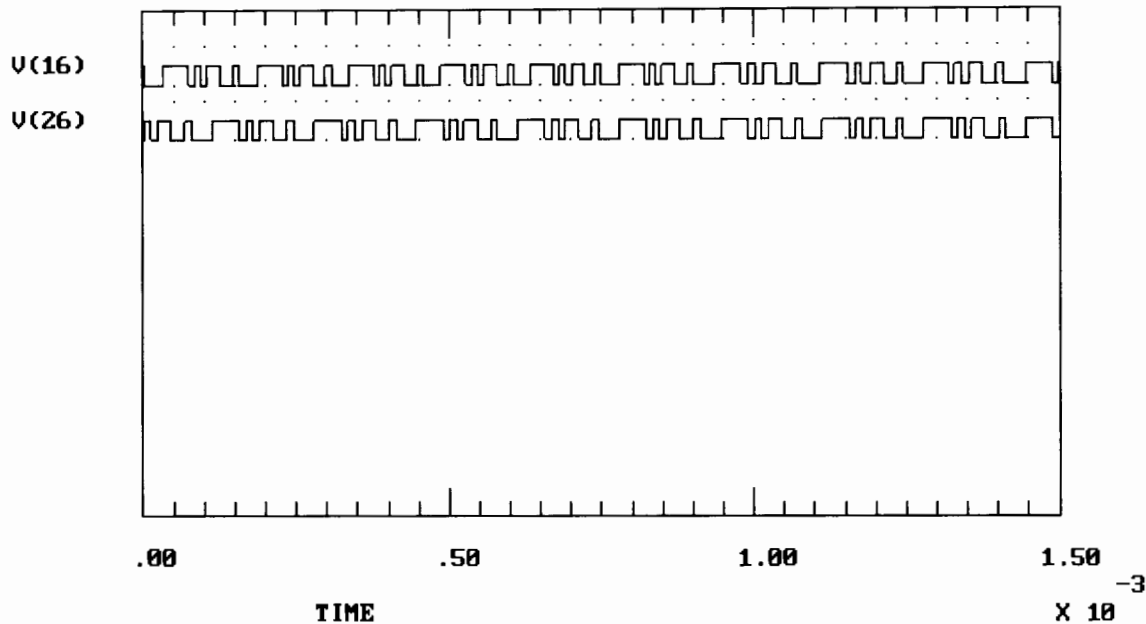
**Figure 4.1** Block diagram of the novel maximal-length sequence synchronisation network with the differentiator in the feedback path as implemented in TESLA. The adjustable clock is modelled as a phase-locked loop preceded by a bandpass filter.

However, as this simulation was to be conducted in a noiseless environment the noise performance of the phase-locked loop does not need to be considered. The loop filter was therefore altered in a few simulation runs until acceptable results were obtained. The TESLA voltage controlled oscillator model allows one to place limits on the maximum frequency deviation that it is possible to generate. This was the major constraint on the adjustable clock.

Parameter	Simulation Value
Sequence Length	15
Centre of Band Pass Filter	100 kHz
Bandwidth of Bandpass Filter	20 kHz
Input Frequency	90 kHz
Code-phase Offset	2 bits
V.C.O. Centre Frequency	100 kHz
Maximum Possible Frequency Deviation	40 kHz
Bandwidth of the Loop Filter	20 kHz

**Table 4.1** Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the feedback path.

The loop was simulated using the parameters as given in Table 4.1. Figure 4.2 shows the two maximal length sequences having achieved code-phase acquisition at  $0.8 \times 10^{-3}$  seconds. The lower trace on the graph depicts how the loop alters the frequency of the receiver's replica of the transmitted waveform in order to align the two sequences. The receiver's maximal-length sequence runs slower than that of the transmitter until code-phase acquisition is achieved. That the code-phase acquisition is maintained is depicted in figure 4.3 which details the time from  $3.6 \times 10^{-3}$  to  $4.0 \times 10^{-3}$  seconds.



**Figure 4.2** Graph of the two maximal-length sequences during the simulation of the synchronisation loop with the differentiator in the feedback path. The upper trace is the transmitted sequence, while the lower trace is the receiver's replica of the transmitted waveform.

However, figure 4.3 shows that the synchronisation loop is unstable and does not keep the receiver's replica of the transmitted waveform precisely aligned with the transmitted waveform. During the interval 3.8 to 3.9 milliseconds the edges of the two maximal-length sequences are not aligned accurately. This indicates that the clocking frequency at which the phase-locked loop is driving the receiver's pseudorandom sequence differs slightly from that of the transmitted sequence. The synchronisation loop with the differentiator in the feedback path is therefore not precisely synchronising the two sequences.

Investigation of the maximal likelihood phase discriminator output waveform ( Figure 4.4 ) clearly depicts the loop's instability. It has been stated earlier that unipolar spikes at this node are an indication that the loop has accurately resolved the code-phase and frequency. Figure 4.4 shows a predominantly negative-going case of lock yet the positive spike after the code-phase acquisition time of  $0.8 \times 10^{-3}$  seconds indicate that the loop is not correctly resolving the transmitter clock frequency. Figure 4.5 was generated using the same simulation parameters given in Table 4.1, but with a different sampling rate. While more stable, the loop does still display small slips with negative spike at  $3.5 \times 10^{-3}$  and  $3.8 \times 10^{-3}$  seconds. Using other sampling times produced traces where the loop instability was worse. The loop is therefore parameter sensitive.

Further evidence of frequency instability in the loop can be found by investigating the adjustable clock's phase discriminator output as this is the waveform which drives the voltage controlled oscillator. The vertical axis on plots of the phase-locked loop's loop filter's output represents the scaled output frequency deviation of the voltage controlled oscillator as a voltage controlled oscillator's output frequency is proportional to input voltage. Figures 4.6 and 4.7 are plots of the voltage controlled oscillator's driving waveform. It has been stated earlier that a constant D.C. value at this node indicates that the code-phase and frequency of the input maximal-length sequence have been accurately resolved. Although a D.C. value does exist after the code-phase acquisition time of  $0.8 \times 10^{-3}$  there are disturbances superimposed upon it. These disturbances indicate incorrect code tracking and are responsible for the erroneous spikes in figure 4.4. The loop is therefore unstable and therefore unable to correctly resolve the input frequency.

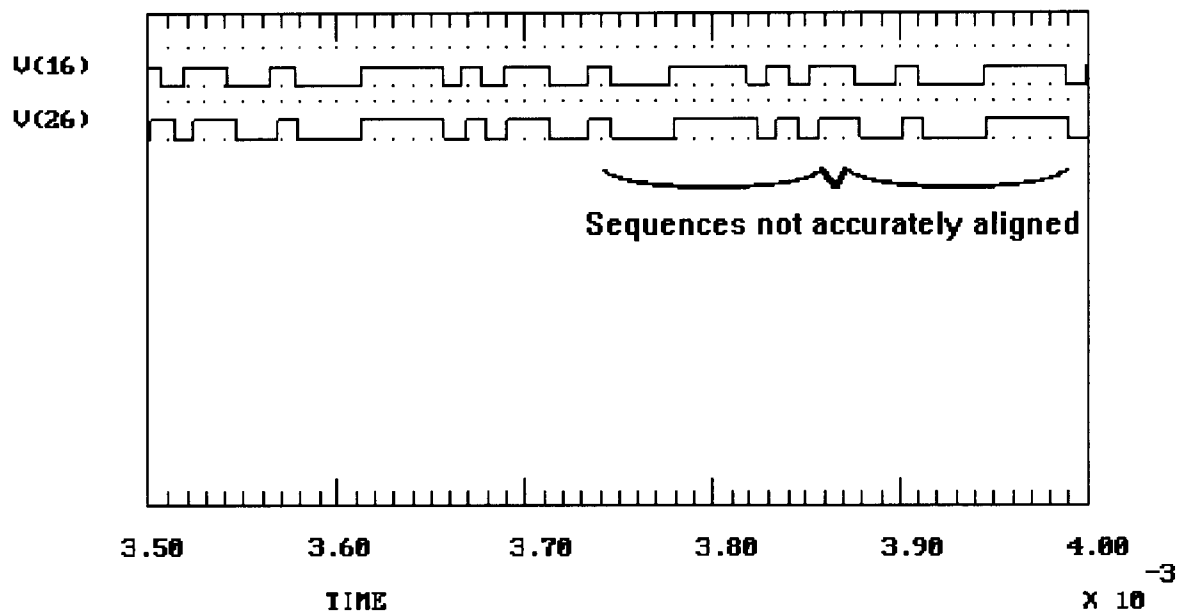


Figure 4.3 Graph of the two maximal-length sequences during the simulation of the synchronisation loop with the differentiator in the feedback path for the final 0.4 milliseconds. The receiver's replica of the transmitted waveform, the lower trace, is not precisely aligned with the transmitted waveform, the upper trace.

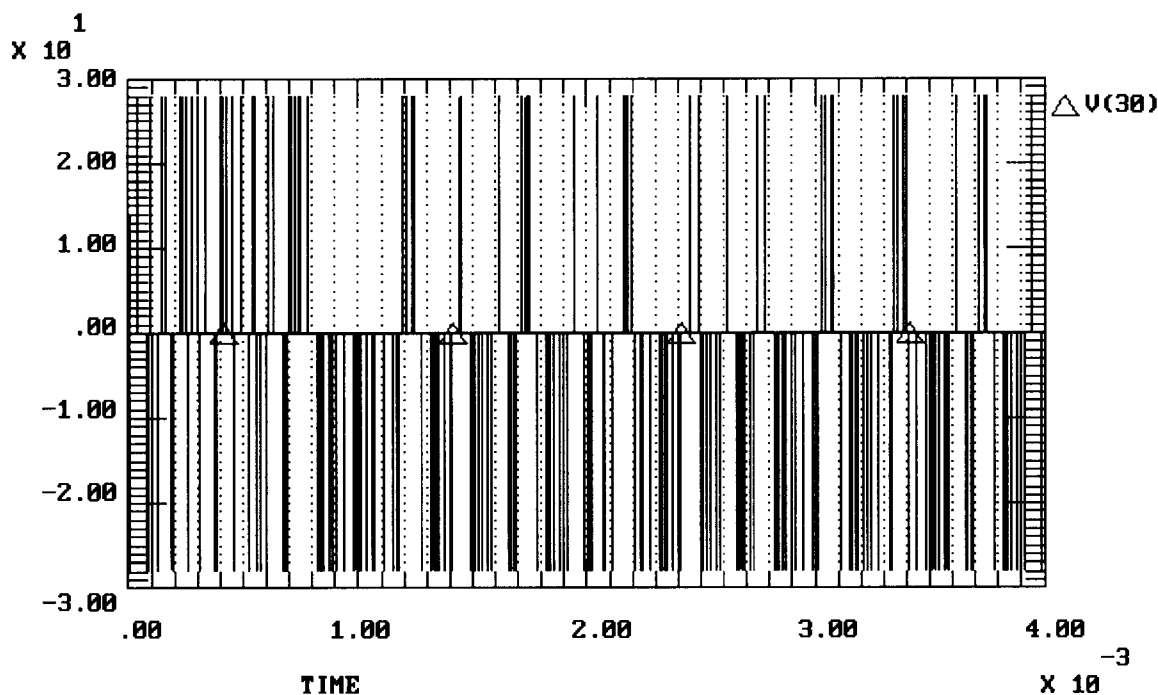


Figure 4.4 Graph of the output of the maximum likelihood phase discriminator. The positive going spikes after 0.8 milliseconds indicate instability.

The loop instabilities can be traced to having the differentiator in the feedback path of the novel synchronisation network. Frequency information in the loop is generated by detecting for the edges of the

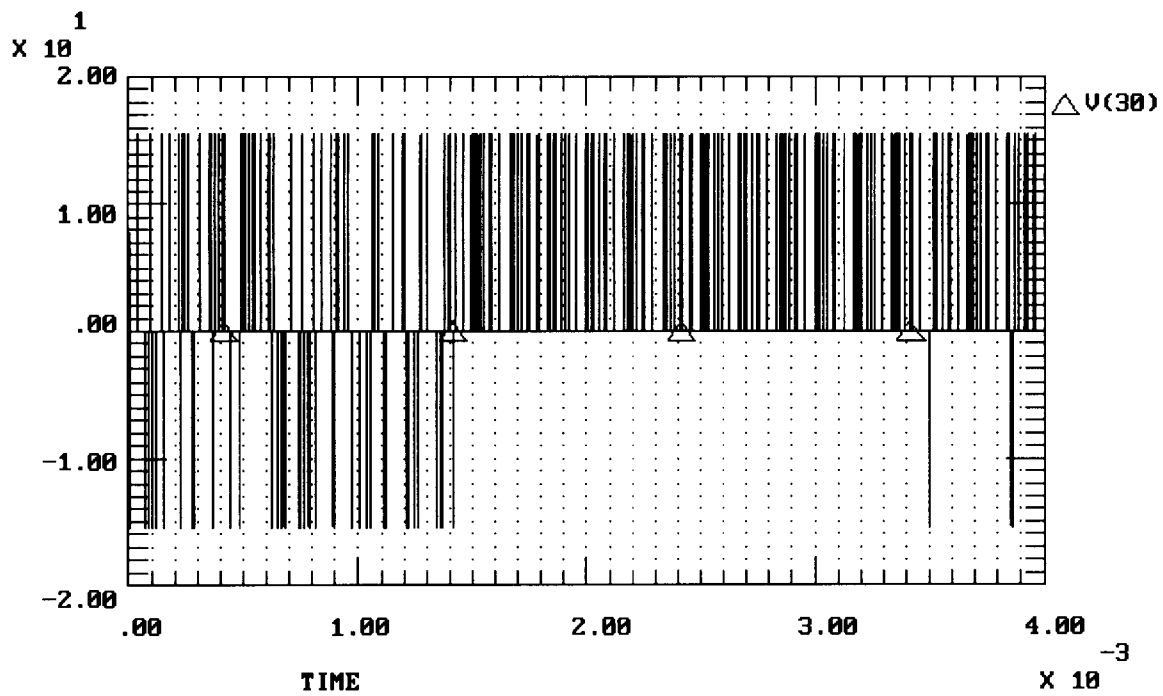


Figure 4.5 Graph of the output of the maximum likelihood phase discriminator. The loop is stable though the two negative going spike at 3.5 and 3.8 milliseconds indicate some slippage.

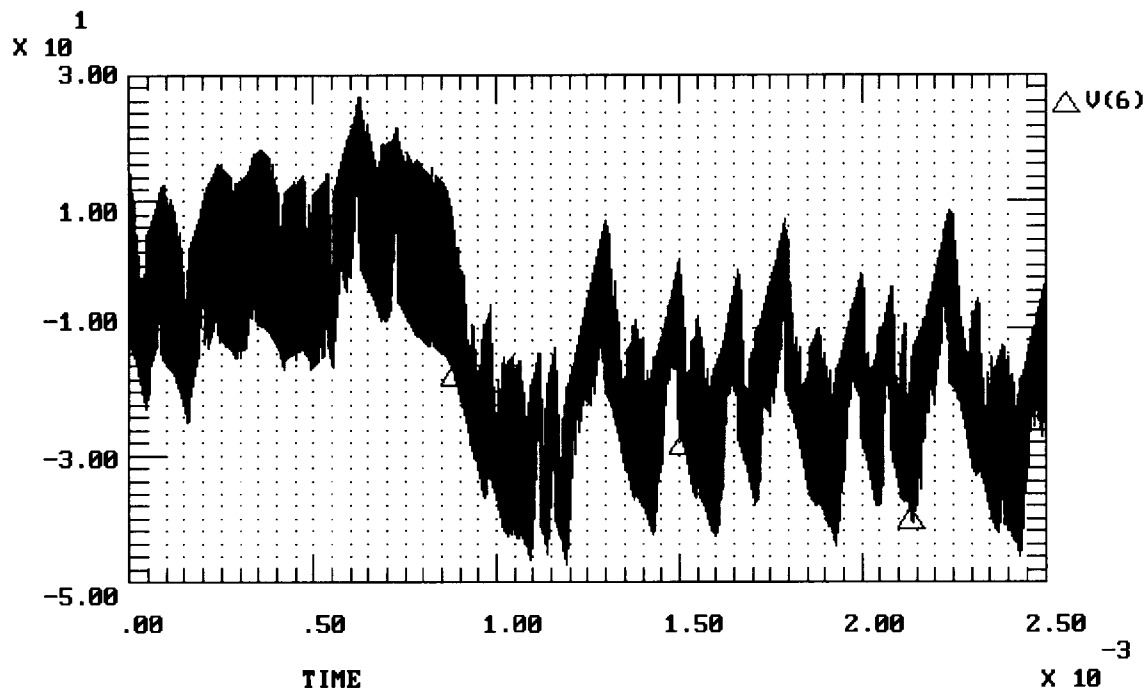
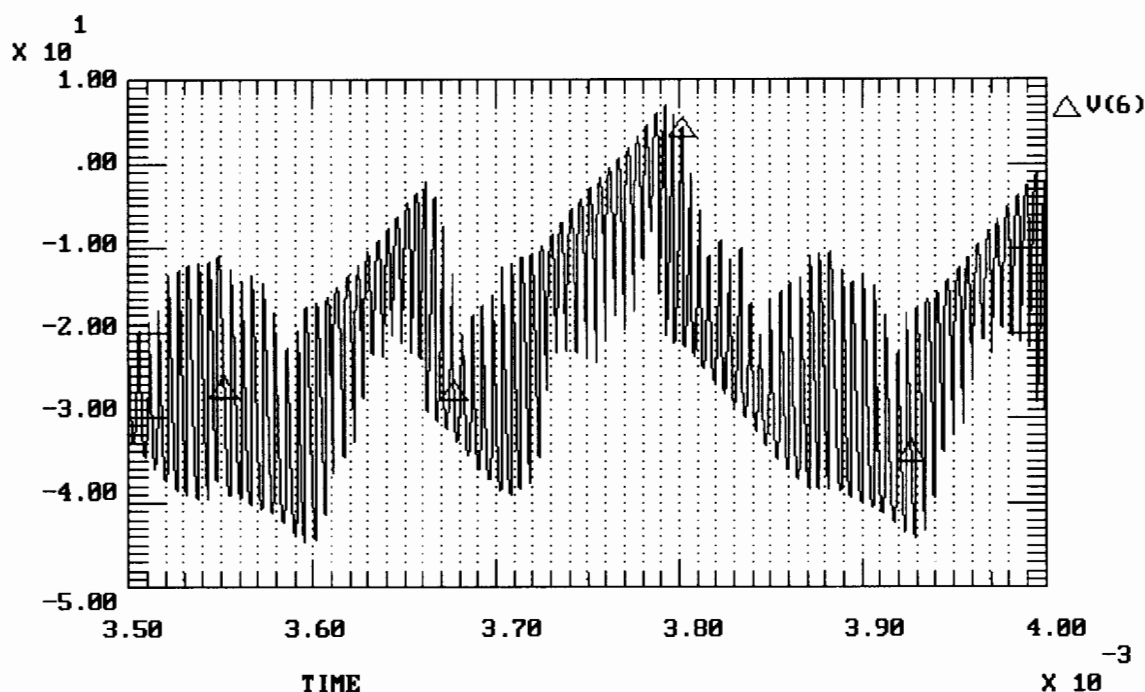


Figure 4.6 Graph of the phase-locked loop's loop filter output. The D.C. level indicates the locked condition while the superimposed disturbances indicates a measure of instability.

maximal-length sequence by differentiation. These spike then stimulate the impulse response of the bandpass filter therefore generating the waveform that the phase-locked loop tracks. Having the differentiator in the feedback path therefore means that the synchronisation network is resolving its own



**Figure 4.7** Graph of the phase-locked loop's loop filter output from 3.5 to 4 milliseconds. The disturbances on the waveform are clearly visible.

frequency information. The jitter that most phase-locked loops exhibit in their outputs is therefore magnified by positive feedback and the loop becomes unstable. This instability in turn implies that the loop is unable to resolve the input code frequency accurately. The transmitter, however, does supply code-phase information due to its deciding the polarity of the spikes by multiplication in the maximal likelihood phase detector. The loop cannot therefore lose code-phase lock. The novel synchronisation network with the differentiator in the feedback path is therefore able to acquire the correct code-phase but unable to resolve the transmitted frequency accurately.

#### 4.1.1 Comparison of the Synchronisation Loop with the Differentiator in the Feedback Path to Other Common Synchronisation Networks.

Although the loop does not resolve both parts of the maximal-length sequence synchronisation condition accurately, it does acquire and maintain code-phase. Whether to use this loop exclusively as a code-phase acquisition network is a design decision. Some of the criterion include :

1. **LOOP COMPLEXITY.** The loop is simple when compared to the common code-phase acquisition networks such as serial search or RASE. However, as is the case with the more complex loops an extra tracking loop would need to be included in the synchronisation loop to accurately resolve the transmitter's clock frequency. This will add to the complexity of the synchronisation solution.

Figure 4.3 does show however that the sequences, once code-phase has been acquired, are never more than half a chip out of code-phase. This would mean that enough energy would be despread by the correlator to detect data. It is therefore likely that the loop might not need an extra tracking loop if one is prepared to accept a slight loss in performance.

2. LOCKING TIMES. The time taken by any synchronisation network to achieve lock is an important design criterion. Taking the equations to determine mean locking time as stated for the low noise case in Chapter 1, the parameters as given in Table 4.1 and setting the evaluation of lock time,  $T_e$  at half a sequence length or  $8 \times T_c$  the following times were calculated.

$$RASE \quad T_m = rT_c + T_e = 0.13 \times 10^{-3}$$

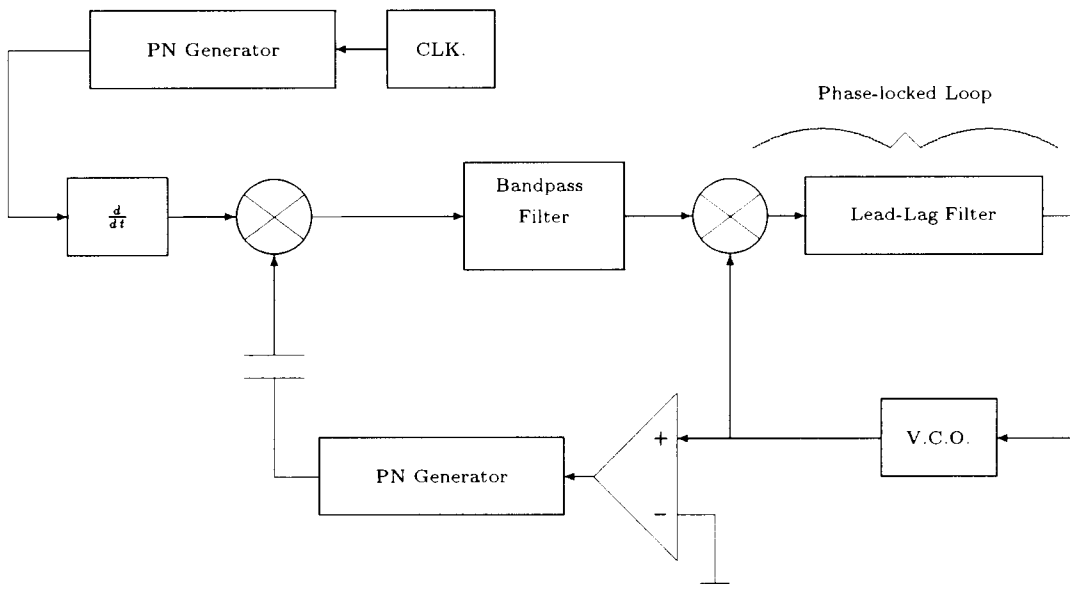
$$Serial \ Search \quad T_m = 2^r T_e = 1.44 \times 10^{-3}$$

From these times it is clear that the novel synchronisation network code-acquisition time of  $0.8 \times 10^{-3}$  is of the same order as those of the more common code-phase acquisition methods.

Although the loop with the differentiator in the feedback loop is a viable design option it does not offer a complete synchronisation solution in one simple, robust loop. As this was the aim of the investigation the loop will not be simulated further. The remainder of the simulations will concentrate on the second loop implementation with the differentiator in the receiver frontend.

## 4.2 SIMULATION OF THE LOOP WITH THE DIFFERENTIATOR IN THE RECEIVER FRONTEND.

The second of novel synchronisation networks ( Figure 3.2 ) to be suggested by J.G. van de Groenendaal employs a variation on the maximum likelihood phase discriminator by incorporating the differentiator in the receiver's frontend rather than in the feedback path. For the purposes of simulation the loop depicted in figure 4.8 was used.



**Figure 4.8** Block diagram of the novel maximal-length sequence synchronisation network with the differentiator in the receiver frontend as implemented in TESLA. The adjustable clock is modelled as a phase-locked loop preceded by a bandpass filter.

The loop input consists of an unmodulated fifteen bit maximal-length sequence of adjustable code-phase. This input is differentiated. The resulting voltage spikes are multiplied by the receiver's replica of the transmitted pseudo-noise sequence. The capacitor allows the receiver's code to swing both positive and negative so as not to multiply by zero. The resultant waveform stimulates the adjustable clock. In this simulation the clock recovery network is modelled as a phase-locked loop preceded by a second order Butterworth bandpass filter. The sine-wave output of the voltage controlled oscillator contained in the phase-locked loop is hard-limited so as to drive the receiver's maximal-length sequence generator.

As in the case of the simulation of the synchronisation network with the differentiator in the feedback path the use of a phase-locked loop preceded by a bandpass filter as the adjustable clock raises some design criterion. As the purpose of the simulation is to investigate only the loop's synchronisation properties and not its noise performance it was decided that the phase-locked loop's bandwidth is not a critical parameter. The bandwidth of the lead-lag filter was therefore set at a reasonable value derived through simulating the loop with different loop parameters. The only limit placed on the adjustable clock was the maximum possible frequency deviation from the centre frequency. The simulation parameters are given in Table 4.2.

Parameter	Simulation Value
Sequence Length	15
Centre of Band Pass Filter	100 kHz
Bandwidth of Bandpass Filter	7 kHz
Input Frequency	101.5 kHz
Code-phase Offset	10 bits
V.C.O. Centre Frequency	100 kHz
Maximum Possible Frequency Deviation	3 kHz
Bandwidth of the Loop Filter	1 kHz

**Table 4.2** Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the receiver frontend for the positive locking case.

Figure 4.9 depicts how the loop obtains code-phase acquisition at a  $1.1 \times 10^{-3}$  seconds. However, the code-phase is acquired at a one bit offset. This is the second of two stable points that the maximal likelihood phase discriminator generates when driving an adjustable clock ( Figures 3.7 and 3.8 ). This is due to the possibility of the spikes generated by the differentiator multiplying either all positive or all negative depending on the alignment of the two maximal-length sequences. As either unipolar pulse train will contain a frequency component at the clocking frequency an adjustable clock network will be able to resolve this frequency and achieve lock.

A train of unipolar spike at the output of the maximum likelihood phase discriminator indicates synchronisation. Figure 4.10 depicts this node. As was not the case in the simulation of the novel synchronisation loop with the differentiator in the feedback path no erroneous spike are noted after the code acquisition time of 1.1 milliseconds. This indicates that the loop in this configuration is stable and able to do code tracking. Further proof of correct frequency tracking is given by the signal driving the voltage controlled oscillator in the phase-locked loop. As this signal is directly proportional to the output frequency any disturbances indicate an altering frequency. As the input frequency is kept constant the output frequency should not alter if correct frequency tracking is achieved. Figure 4.11 depicts the V.C.O. driving frequency. A clear D.C. level is noted, after 1.1 milliseconds, with superimposed, periodic, high frequency jitter. However, as the superimposed jitter is a natural effect in second-order phase-locked loop and as its integrated effect with respect to time tends to zero the loop is stable. The loop therefore tracks the input pseudorandom sequence.

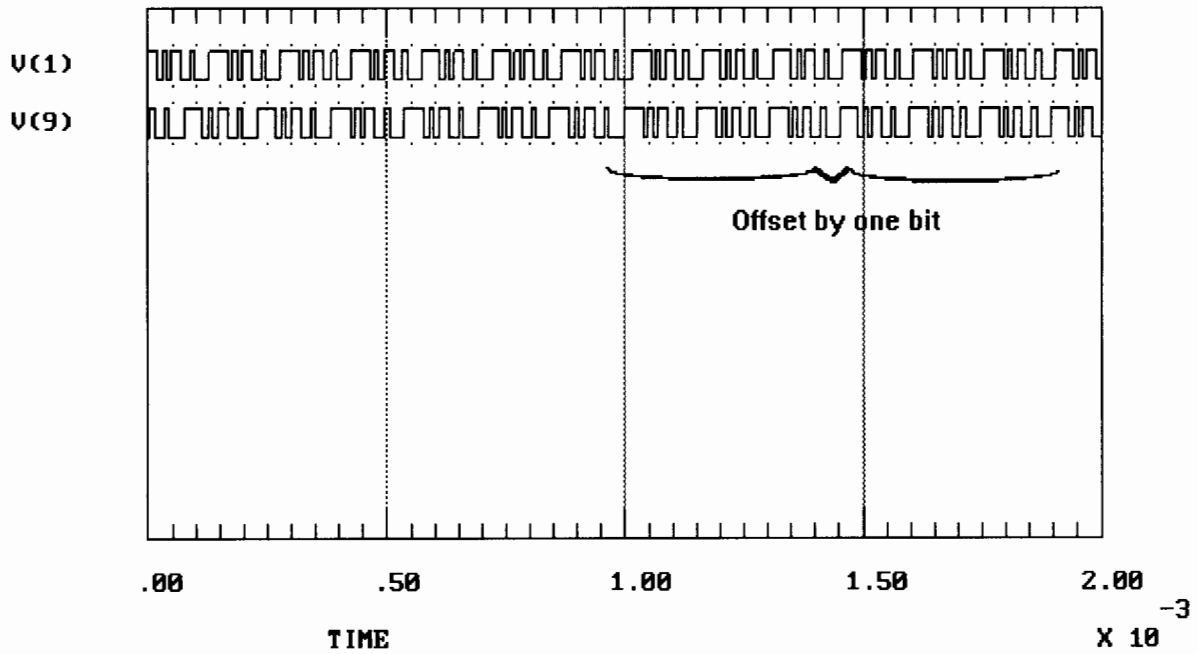


Figure 4.9 Graph of the two maximal-length sequences achieving lock during the simulation of the synchronisation loop with the differentiator in the receiver frontend. The receiver's replica of the spreading waveform, the lower trace, achieves lock at a one bit offset.

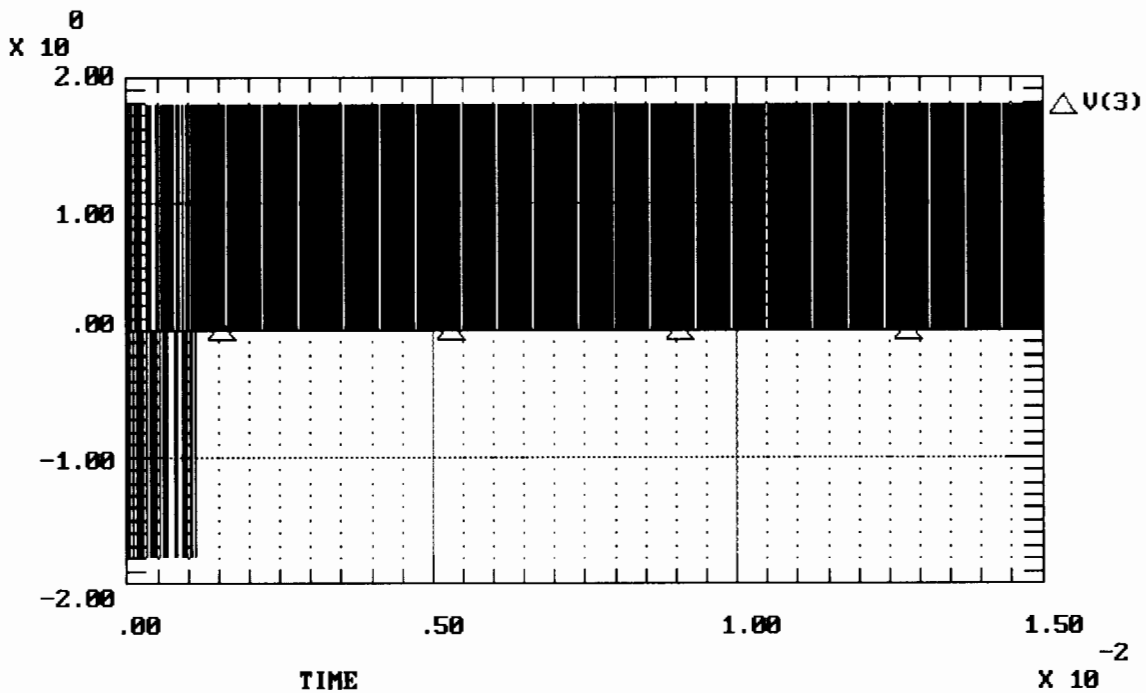


Figure 4.10 Graph of the output of the maximum likelihood phase discriminator. The positive going spikes after 1.1 milliseconds indicate a stable lock.

Further simulation, using the parameters as given in Table 4.3, reveal that the second stable point, negative going lock, in the baseband Spread Spectrum search grid as expressed in terms of integrated

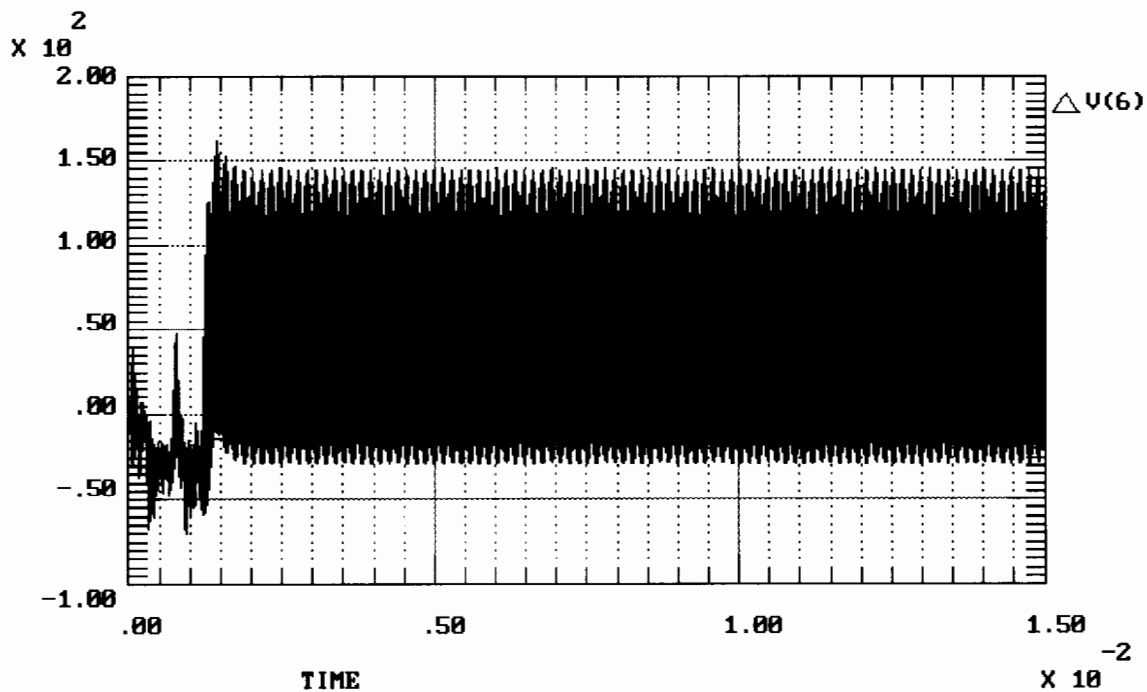


Figure 4.11 Graph of the V.C.O. driving signal. The D.C. level indicates that the loop is accurately performing code tracking.

phase, ( Figures 3.7 and 3.8 ) is stable enough, in this loop configuration, to allow for code tracking. Figure 4.12 depicts the negative locking condition. Lock is achieved at 0.6 milliseconds. As no erroneous spike occur after this time the loop is accurately tracking the input sequence. Investigation of the voltage controlled oscillator's driving waveform ( Figure 4.13 ) shows a steady state D.C. level after the locking time. This is further evidence that the loop in this configuration is stable and able to accurately track the input clock frequency.

Parameter	Simulation Value
Sequence Length	15
Centre of Band Pass Filter	100 kHz
Bandwidth of Bandpass Filter	7 kHz
Input Frequency	99 kHz
Code-phase Offset	7 bits
V.C.O. Centre Frequency	100 kHz
Maximum Possible Frequency Deviation	3 kHz
Bandwidth of the Loop Filter	1 kHz

Table 4.3 Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the receiver frontend for the negative locking case.

Under the negative going lock condition the code-phase of the receiver is accurately aligned with that of the transmitter. Figure 4.14 shows how the receiver's maximal-length sequence, the lower trace, is pulled into lock by running slower than the transmitter's pseudorandom sequence, the upper trace. After 0.6 milliseconds the two sequences are aligned.

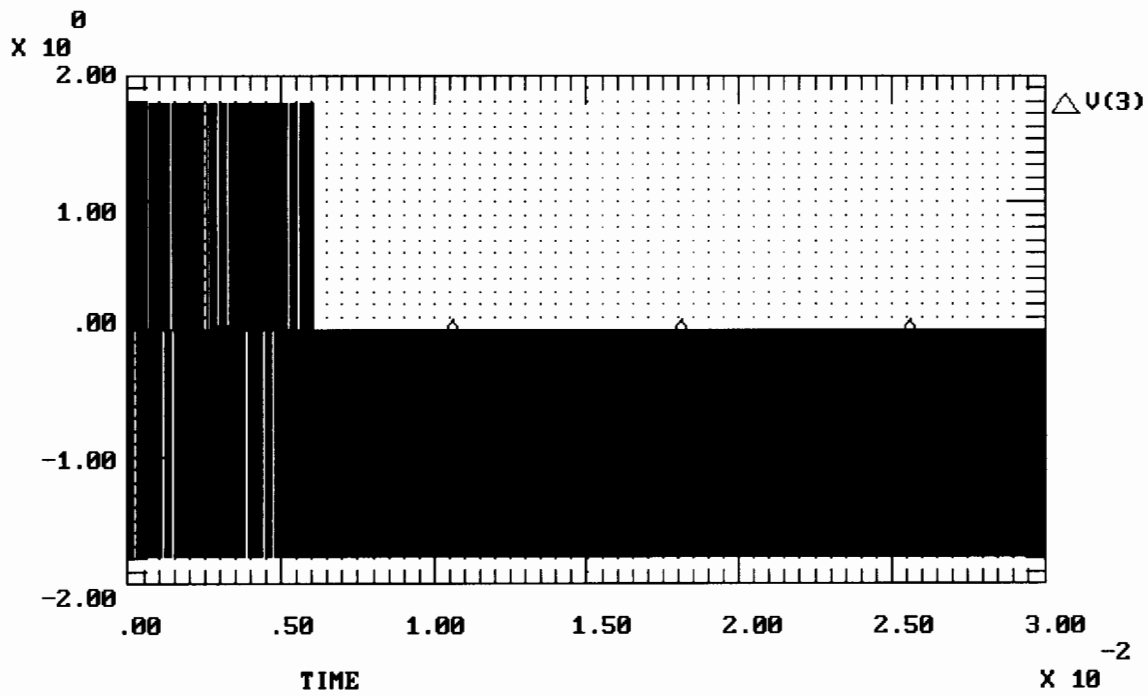


Figure 4.12 Graph of the output of the maximal likelihood phase detector for the negative locking case. No erroneous spike after the locking time of  $0.6 \times 10^{-3}$  indicate accurate code tracking of the input pseudorandom sequence.

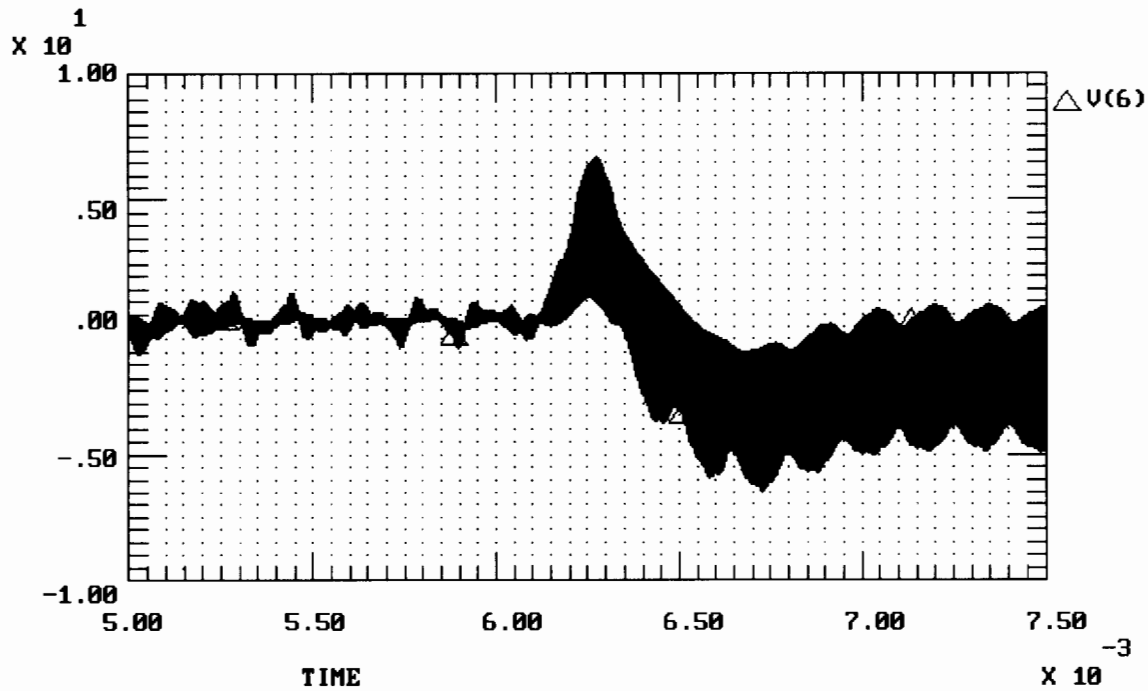
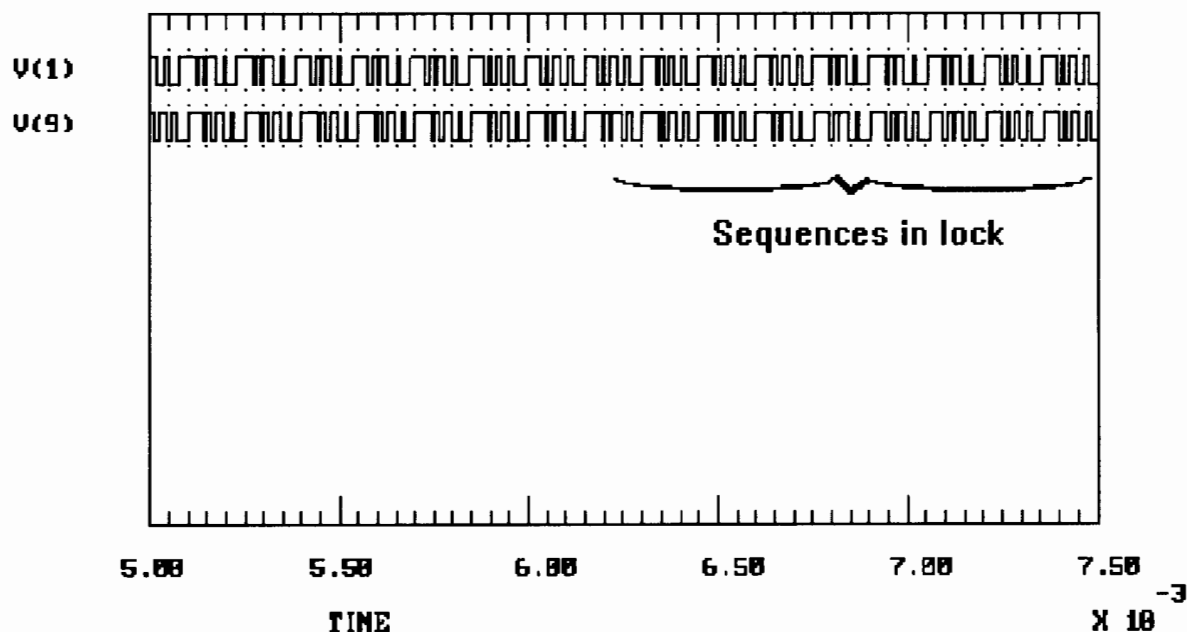


Figure 4.13 Graph of the V.C.O. driving signal for the negative going case. The D.C. level indicates that the loop is accurately performing code tracking.

Placing the differentiator in the synchronisation network's input path results in a stable loop which is able to perform code-phase acquisition and tracking. Unlike the novel synchronisation loop with the



**Figure 4.14** Graph of the two maximal-length sequences achieving lock during the simulation of the synchronisation loop with the differentiator in the receiver frontend. The receiver's replica of the spreading waveform, the lower trace, achieves lock precisely.

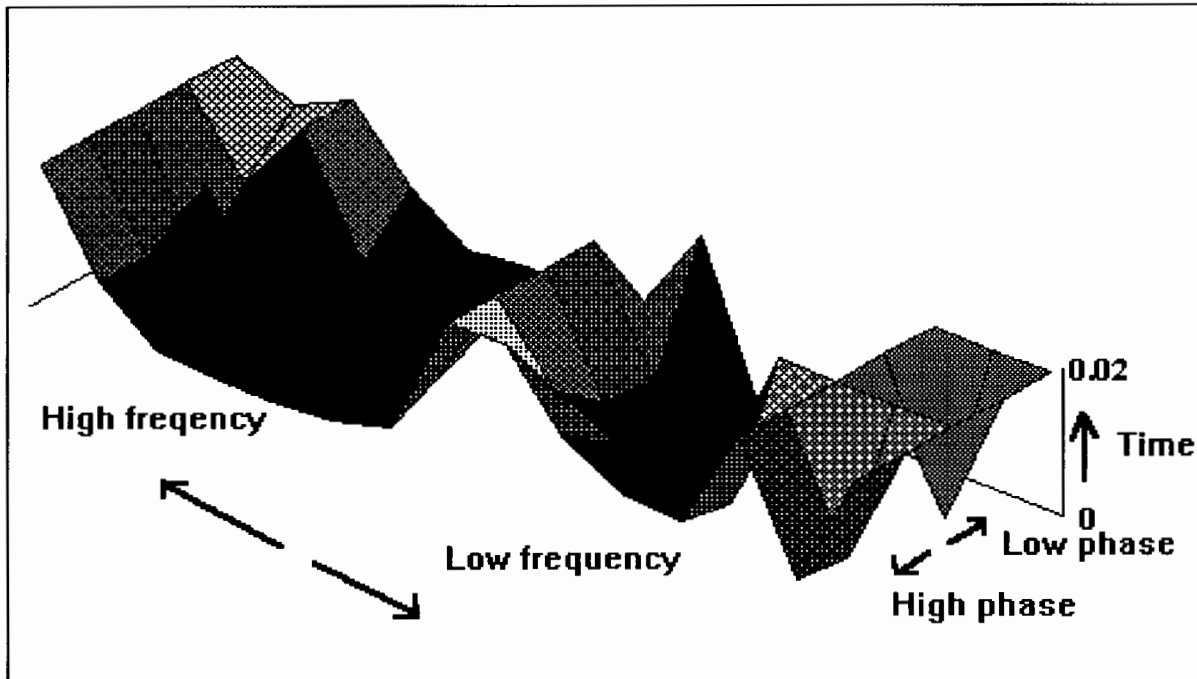
differentiator in the feedback path, the voltage spike due to differentiation are now a function of the transmitter's clock frequency and not the synchronisation loop's instantaneous frequency. The train of voltage spikes therefore contain a frequency component at the transmitter's clock frequency rather than at the receiver's clock frequency. As code tracking requires that the receiver must accurately resolve the transmitted clock frequency this synchronisation loop configuration with the differentiator in the receiver frontend is able to resolve and track the required component. The loop is therefore able to track the transmitted code.

The feedback path of the synchronisation loop with the differentiator in the receiver's frontend determines code-phase acquisition. If the two maximal-length sequences are not aligned the spike generated by the differentiator will not all multiply either positive or negative and the loop will change its conditions. As code-phase acquisition is a 'coarse' synchronisation procedure slight variations due to the receiver's clock will not be fed back to drive the loop out of lock. This configuration of having the synchronisation loop feedback path regulating code-phase and having the correct frequency at the input is more stable and accurate than the inverse, as is done in the synchronisation loop with the differentiator in the feedback path. The synchronisation loop with the differentiator in the input path is therefore able to perform code tracking as well as code acquisition. The loop therefore offers a complete synchronisation solution in a simple configuration.

#### 4.2.1 Locking Times of the Synchronisation Loop with the Differentiator in the Input Path.

As the novel synchronisation loop with the differentiator in the input path follows a chaotic path round the baseband Spread Spectrum search grid deriving a locking time is difficult. Locking times were therefore determined by simulation for a variety of input frequencies and code-phase offsets.

Figures 4.15 and 4.16 show the locking times for a fifteen bit maximal-length sequence and a seven bit maximal-length sequence respectively. The  $x$  and  $y$  axes are the code-phase and frequency offsets while the  $z$  axis denotes the locking time. The simulations were run for  $2 \times 10^{-3}$  seconds. The system was deemed not able to achieve lock if this time was exceeded.

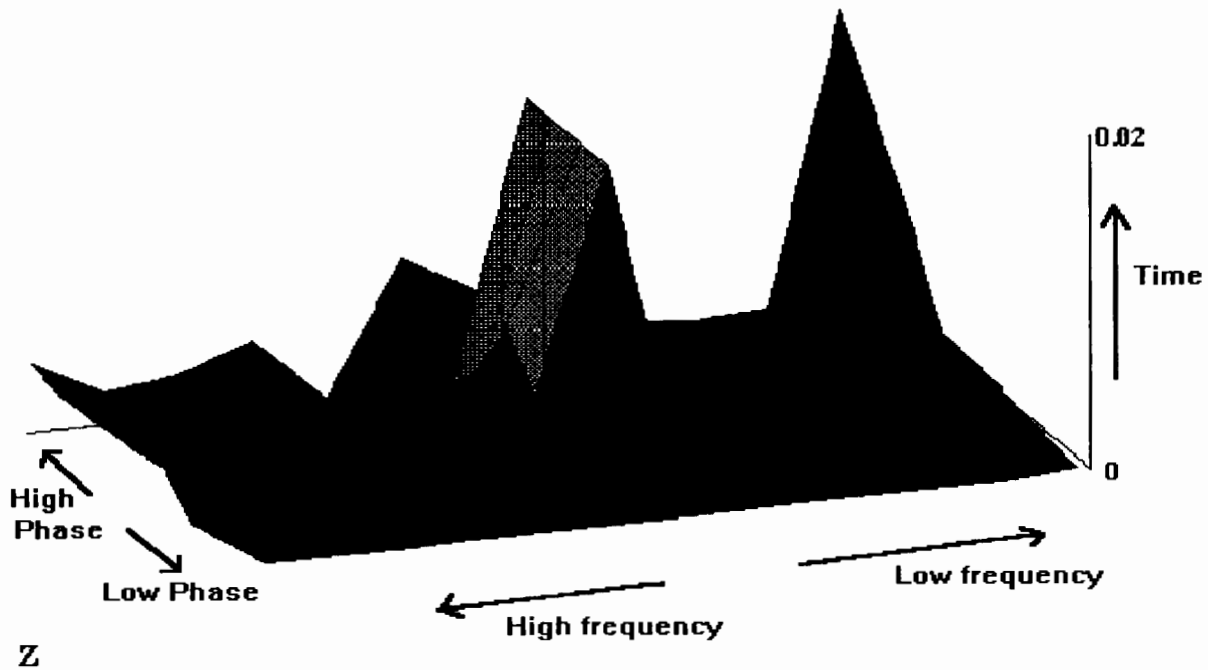


**Figure 4.15** Graph of the locking times for a 15 bit maximal-length sequence. The  $x$  and  $y$  axes are the code-phase and frequency offsets while the  $z$  axis denotes the locking time.

The locking surfaces display some interesting trends. The times form a valley around the centre frequency of the receiver's voltage controlled oscillator stretching up to the maximum frequency offset the clock can generate. It was also found that the spikes were generally negative going for a frequency offset below that of the centre frequency of the receiver clock and positive going for frequencies greater than the centre frequency. Furthermore, the locking times are mirrored around the receiver's voltage controlled oscillator's centre frequency and code-phase differences. A locking time for a negative frequency offset and a three bit code-phase offset will be of the order of a positive frequency offset and a twelve bit code-phase offset in the case of the fifteen bit maximal-length sequence.

The 'ridge' of locking times found at zero input frequency offset indicate that the loop finds it difficult to adjust just its code phase when the receiver's clocking frequency is equal to that of the transmitter. These points are however very localised. Offsetting the transmitter clock's starting phase by a few degrees from that of the receiver, while not altering the frequency, resulted in lock within the required time frame. Furthermore, as frequency uncertainty due to oscillator drift and Doppler shift is present in practice the real system is unlikely to dwell at these points.

The time taken by the novel synchronisation network to achieve lock is an one important characteristic in deciding whether the loop is a viable synchronisation alternative to the more common networks. Taking the equations to determine mean locking time as stated for the low noise case in chapter 2.5, the parameters as given in Table 4.2 and setting the evaluation of lock time,  $T_e$ , at half a sequence length the times for the common synchronisation techniques were calculated. The times given for the novel synchronisation loop with the differentiator in the receiver frontend were calculated by taking the sum of the simulated locking times,  $T_s$ , and dividing by the number of points,  $N$ . As it was felt that the times



**Figure 4.16** Graph of the locking times for a 7 bit maximal-length sequence. The  $x$  and  $y$  axes are the code-phase and frequency offsets while the  $z$  axis denotes the locking time.

at zero frequency offset were misleading they were discarded as well as the times for frequency offsets larger than the phase-locked loop's tracking range.

#### MEAN LOCKING TIMES FOR THE FIFTEEN BIT MAXIMAL-LENGTH SEQUENCE CASE.

$$\begin{aligned} \text{RASE} \quad T_m &= rT_c + T_e = 0.13 \times 10^{-3} \\ \text{Serial Search} \quad T_m &= 2^r T_e = 1.44 \times 10^{-3} \\ \text{Novel Loop} \quad T_m &= \frac{\sum T_s}{N} = 3.39 \times 10^{-3} \end{aligned}$$

#### MEAN LOCKING TIMES FOR THE SEVEN BIT MAXIMAL-LENGTH SEQUENCE CASE.

$$\begin{aligned} \text{RASE} \quad T_m &= rT_c + T_e = 7 \times 10^{-5} \\ \text{Serial Search} \quad T_m &= 2^r T_e = 0.32 \times 10^{-3} \\ \text{Novel Loop} \quad T_m &= \frac{\sum T_s}{N} = 1.93 \times 10^{-3} \end{aligned}$$

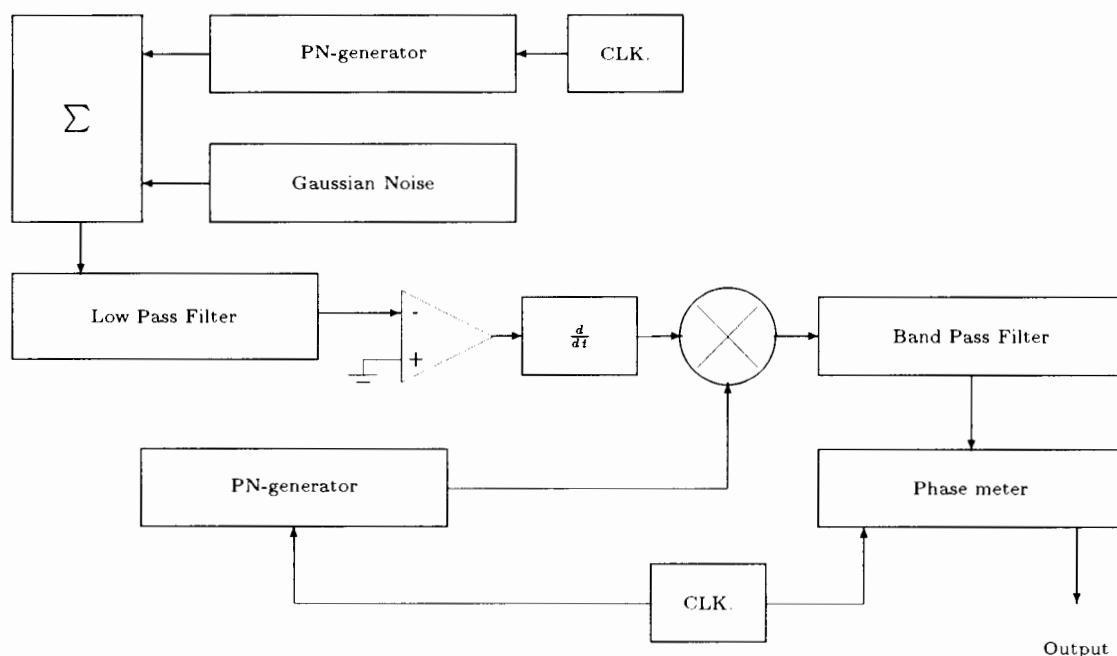
The locking time of the novel synchronisation network with the differentiator in the input path is approximately three times longer than those of the other methods given.

However, the times are not strictly comparable. The novel loop offers a complete synchronisation solution while the RASE and serial search techniques are only used for code-phase acquisition. Systems based on the RASE and serial search techniques still need another synchronisation loop to correctly track the input code. The stated mean synchronisation times do not take this extra synchronisation time into account. The times are presented to allow for a rough comparison only. The novel loop's locking time is still however of the same order as those of the common code acquisition networks.

### 4.3 NOISE PERFORMANCE OF THE SYNCHRONISATION NETWORK WITH THE DIFFERENTIATOR IN THE RECEIVER FRONTEND.

All real world transmission channels will introduce some noise and hence cause errors in a communications system. The amount of noise that can be tolerated at a certain transmitted signal strength before error rates become too high is an important design consideration. In the case of Spread Spectrum the errors will become unacceptable if the synchronisation network is unable to achieve lock due to the interference caused by the noise as the receiver will be unable to despread the transmitted waveform.

In chapter 3.3 the concept of integrated phase was used to determine whether the novel synchronisation loops would be able to achieve lock. Figure 4.17 shows the block diagram used in TESLA simulations to generate the baseband Spread Spectrum search grid expressed in terms of integrated phase in a noisy environment. White additive Gaussian noise at a level of 25 Volts r.m.s. is added to the transmitter pseudorandom sequence using TESLA's noise model. The resultant waveform is low pass filtered by a fourth order Butterworth filter and then hard-limited. The bit edges are detected by differentiation. After multiplication by the second maximal-length sequence the waveform is band pass filtered by a second order Butterworth filter. TESLA's phase meter measures the phase difference between the band pass filtered waveform and the clock driving the second maximal-length sequence. The phase measurement is then integrated.

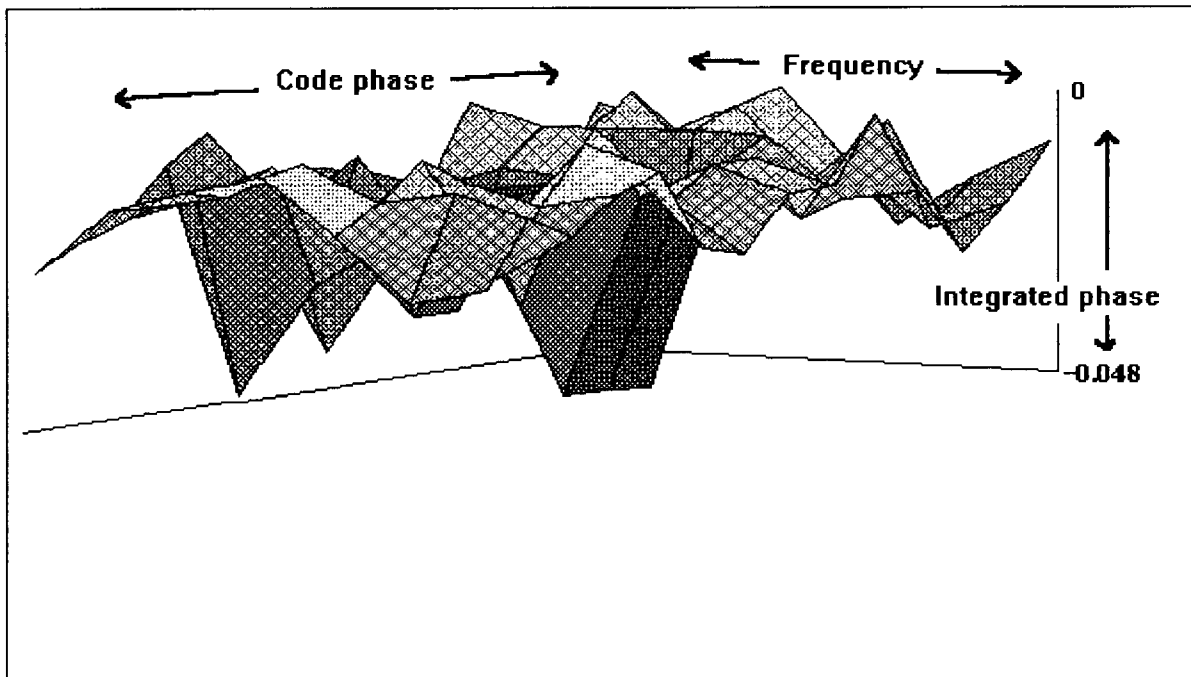


**Figure 4.17** Block diagram of the TESLA simulation used to calculate the integrated phase in a noisy environment.

Figure 4.18 depicts the baseband Spread Spectrum search grid in terms of the negative absolute value of the integrated phase. A hollow in the surface therefore represents a more stable search cell than the surrounding cells. The addition of noise 'deepens' the dips that exist in the incorrect search cells. Using the 'ball' analogy ( Chapter 3.3 ) to describe the mechanism whereby the novel synchronisation loop acquires lock, these dips will result in the loop being attracted to these incorrect cells. However, as

the dips, other than the one when the sequences are aligned, are the result of noise they will randomly appear and disappear. The 'ball' will therefore eventually roll into the stable point generated by the maximal-length sequences being synchronised.

The integrated phase surface gives two insights into the operation of the novel maximal-length sequence synchronisation network in a noisy environment. Firstly, due to the extra depth of the dips, as opposed to the noiseless case, the loop will dwell longer in incorrect cells than it did in the noiseless case. Locking times will therefore be extended. Secondly, a level of noise will exist when the dip, due to the maximal-length sequences being aligned, will not be the dominant stable cell. Under this condition the loop will be unable to achieve and maintain code tracking. The novel synchronisation loop should therefore be able to operate correctly under the influence of noise, providing enough information is still present in the loop to allow for lock.



**Figure 4.18** Graph of the baseband Spread Spectrum search grid characterised by integrated phase in a noisy environment. The stable points due to the maximal-length sequences being aligned are still dominant. The novel loop will therefore enter and remain in these cells, thereby synchronising the two pseudorandom sequences.

A common method of determining noise levels in relation to the system parameters is the signal-to-noise ratio,  $S_N$ . This is a log scale value and can be calculated using

$$S_N = 10 \log\left(\frac{V_s^2}{V_n^2}\right)$$

where  $V_s$  is the signal voltage and  $V_n$  is the root mean square noise voltage bounded by the system noise bandwidth. This equation can be manipulated for the TESLA simulation environment. The TESLA noise model allows one to set the root mean square noise voltage. However, the noise power is distributed over the simulation bandwidth which is equal to half the inverse of the sampling rate. The noise power must therefore be scaled to the system noise bandwidth. The signal-to-noise ratio is therefore given by

$$S_N = 10 \log\left(\frac{2T_s V_s^2}{B_s V_n^2}\right)$$

where  $T_s$  is the sampling rate and  $B_s$  is the system bandwidth.

The signal-to-noise ratio therefore gives an indication of the amount of desired information per unit of noise in the system bandwidth. This measurement can therefore describe noise levels in terms of whether the synchronisation loop is able to resolve the transmitted information. The TESLA simulations to determine the noise performance of the novel synchronisation loop with the differentiator in the input path will be performed in terms of this parameter.

Figure 4.19 depicts the block diagram used in the TESLA simulations of the novel synchronisation network with the differentiator in the receiver input path in a noisy environment. The transmitter maximal-length sequence is added to white additive Gaussian noise generated by the TESLA noise model. The resultant waveform is low pass filtered and then hard-limited. This input low pass filter determines the system noise bandwidth. The resultant waveform, possibly containing errors, forms the input to the novel synchronisation network with the differentiator in the receiver's input path. The received waveform is also multiplied by the receiver's replica of the spreading waveform and the resultant waveform is integrated. This will allow one to detect despreading of the input waveform and therefore determine whether lock has been achieved. This is necessary as the noise should cause errors in the spikes forming the output of the maximal likelihood phase discriminator and hence disturbances in the driving waveform of the phase-locked loop's voltage controlled oscillator. These might therefore no longer offer good indications of lock. However, as the expectation function of additive white Gaussian noise symmetrically centred on zero D.C. is zero, the despreading of the waveform should give a good indication of lock as the effects of noise are lessened.

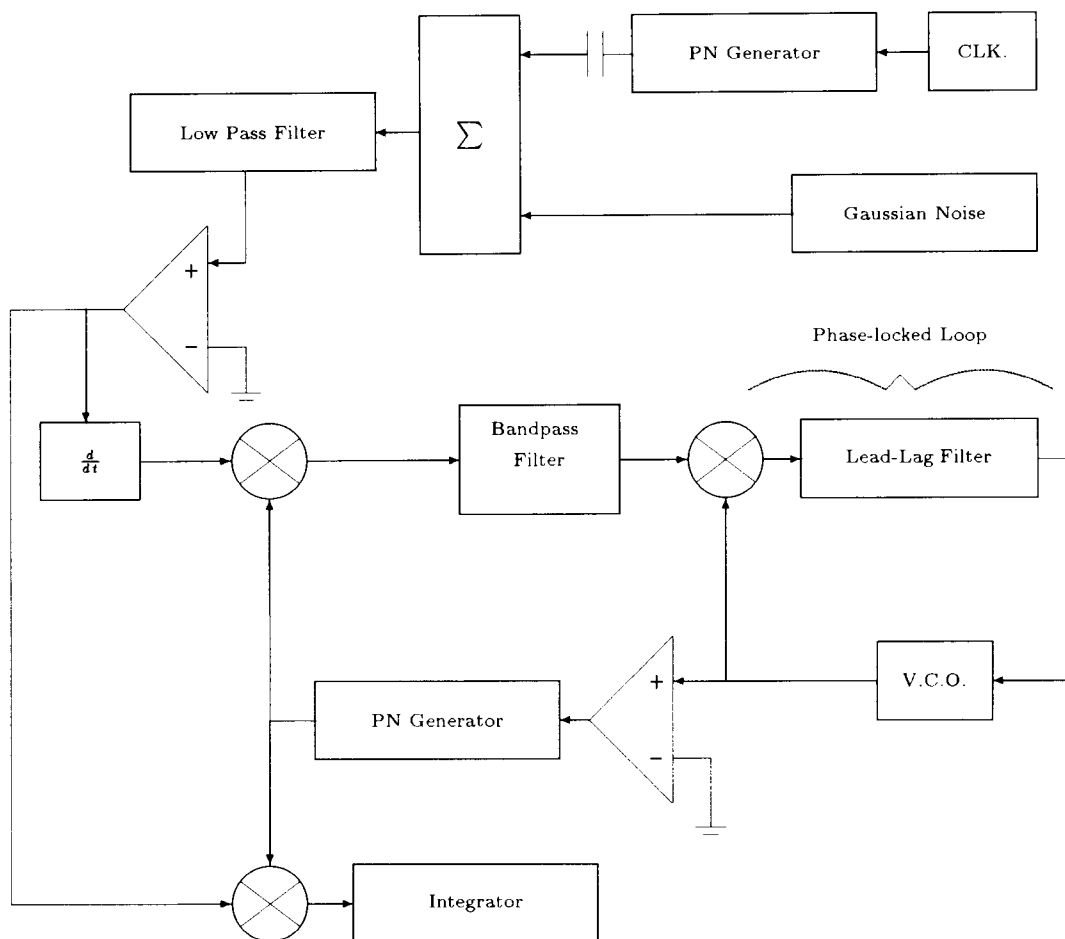
### 4.3.1 The Novel Synchronisation Loop at a Signal-to-Noise Ratio of 4 dB.

Table 4.4 gives the parameters used in the TESLA simulation of the novel synchronisation network ( Figure 4.19 ) in a noisy environment. The input low pass filter used was the TESLA R.C. low pass model. Although the roll off, in the frequency domain, of a filter of this type is not steep, and hence its noise rejection not ideal, it was used as it models an easy practical implementation. Furthermore, the initial simulations were performed to investigate the noise performance rather than to obtain detailed results. Simulations with a tighter input low pass filter will be presented later in the text.

Parameter	Simulation Value
Sequence Length	15
Centre of Band Pass Filter	100 kHz
Bandwidth of Bandpass Filter	7 kHz
Input Frequency	101 kHz
Code-phase Offset	7 bits
V.C.O. Centre Frequency	100 kHz
Maximum Possible Frequency Deviation	3 kHz
Bandwidth of the Loop Filter	1 kHz
Type of Input Low Pass Filter	Resistor-Capacitor
Bandwidth of Input Low Pass Filter	99 kHz
Noise Root Mean Square Value	10 Volt
Sampling Rate	$1 \times 10^{-7}$ sec

**Table 4.4** Table of the parameters used in the simulation of the novel synchronisation loop with the differentiator in the input path in a noisy environment.

As this simulation is run in a noisy environment care must be taken with the design of the adjustable clock. As the adjustable clock is modelled as a phase-locked loop preceded by a bandpass filter, two



**Figure 4.19** Block diagram of the circuit used in the simulation of the novel maximal-length sequence synchronisation network with the differentiator in the input path under the influence of noise.

possibilities exist for the suppression of noise by either tightening the loop or input filter. It was decided that the phase-locked loop's loop filter would remain as it was in the noiseless simulations in order to allow the loop to quickly acquire the correct frequencies, but that the bandpass filter would be narrowed so as to reduce the effects of possible erroneous spikes at other frequencies. The bandpass filter was therefore modelled as a fourth order Butterworth filter with a seven kilohertz bandwidth. It was found that this configuration was stable. Further alterations to the bandwidth of the bandpass filter did little to change the locking time, possibly as the response time of the entire novel Maximal-length sequence synchronisation loop was the dominant factor.

Figure 4.20 shows the transmitted, lower trace, and received, upper trace, waveforms. Errors in the estimation of the incoming bits are made. Each error generates an edge which will result in an erroneous spike at the output of the maximal likelihood phase discriminator. However, the clocking frequency of the transmitter is not lost due to the influence of the noise. Figure 4.21 shows the spectrum of the output of the bandpass filter. A component, which corresponds to a dip in the baseband Spread Spectrum search grid as expressed in terms of integrated phase, exists at the correct frequency to allow the novel synchronisation loop to track the transmitted maximal-length sequence.

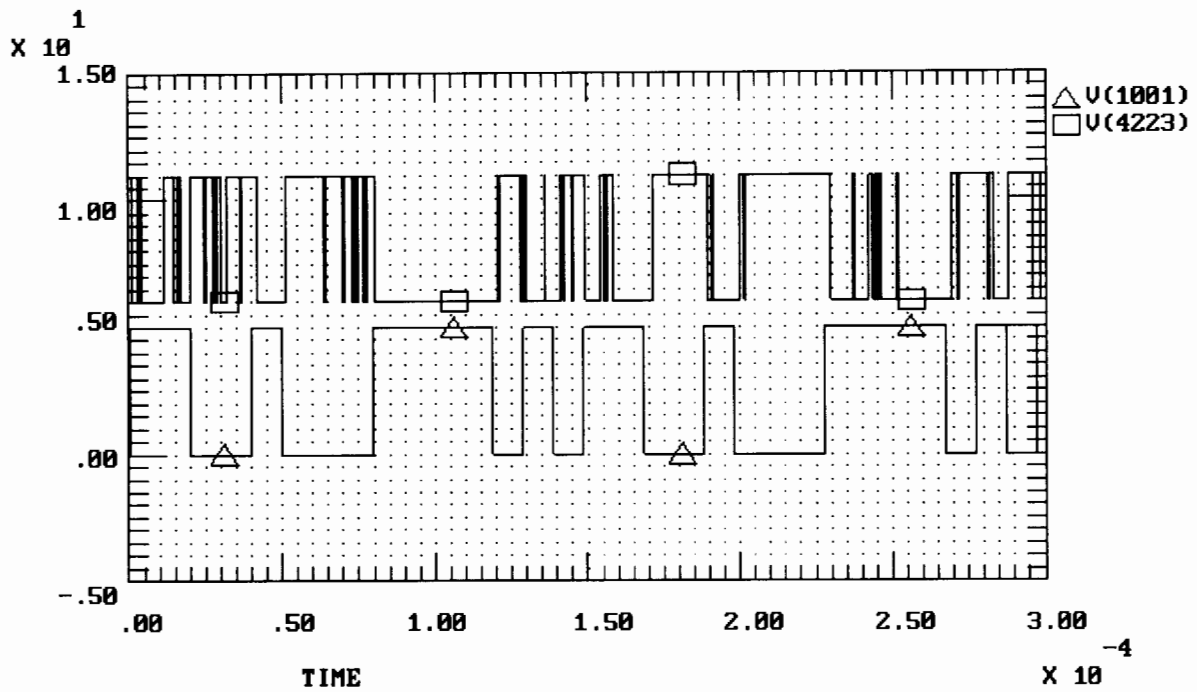


Figure 4.20 Graph of the transmitted, lower trace, and received, upper trace, maximal-length sequences. Errors are made by the receiver in its estimation of the received bit.

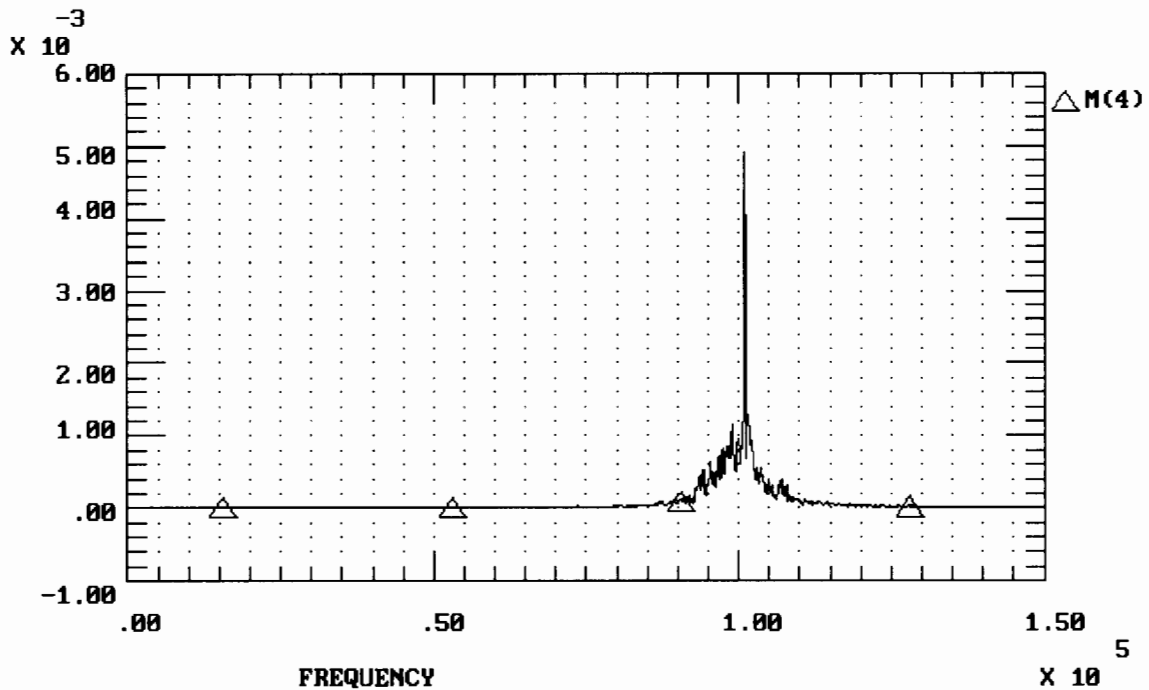
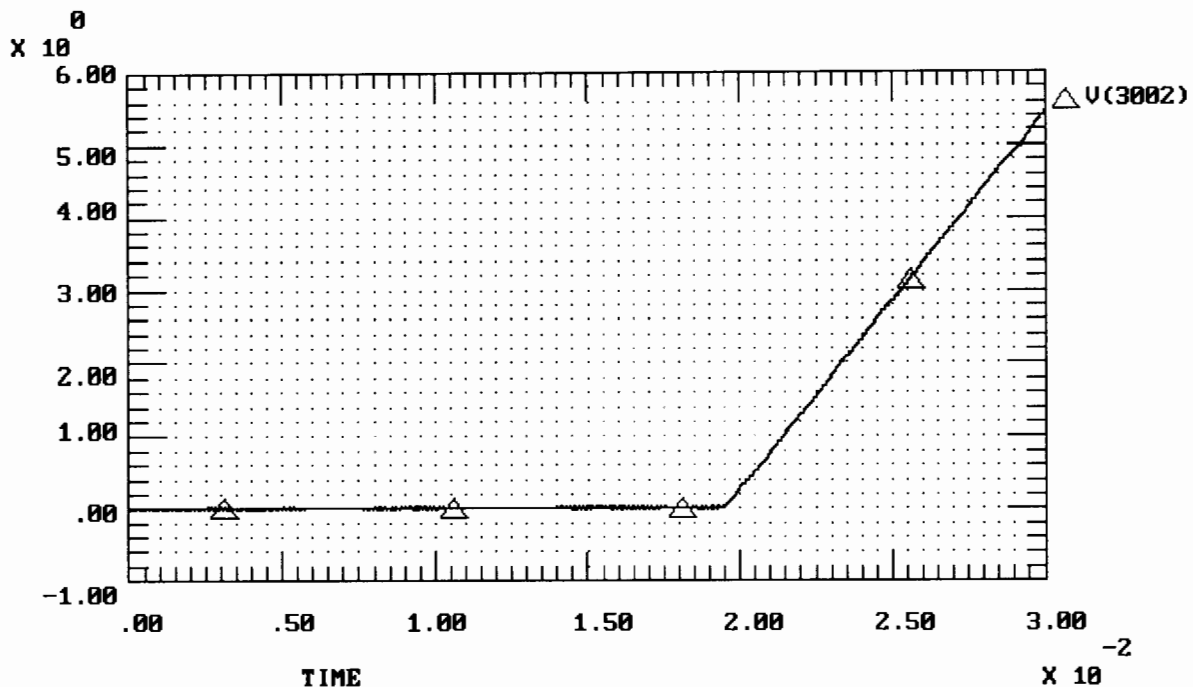


Figure 4.21 Graph of the spectrum of the output of the bandpass filter. A strong component exists at the transmitter's clocking frequency therefore allowing the loop to track the input.

Due to the influence of the noise lock must be detected through the despreading of the transmitted waveform. Figure 4.22 shows the output of the integrator in figure 4.19. The despreading of the input

at 20 milliseconds indicates lock. The locking times are therefore increased from the noiseless case. The novel synchronisation loop is nevertheless able to achieve lock under the influence of noise.



**Figure 4.22** Graph of the output of the integrator. Despreading of the transmitted waveform occurs at  $2 \times 10^{-2}$ .

### 4.3.2 The Novel Synchronisation Loop at a Signal-to-Noise Ratio of 0 dB.

The signal-to-noise ratio at which the synchronisation loop fails to achieve lock is an important criterion. Using the other parameters given in Table 4.4 the noise was increased to sixteen Volts ( root mean square ).

Figure 4.23 shows that under these noise levels the amount of errors made by the receiver in its estimating of the transmitted waveform increases. However, enough resolvable information still exists at the input to the novel synchronisation loop's adjustable clock to allow for synchronisation to occur ( Figure 4.24 ). The frequency component at the transmitter's clock frequency of 101 kilohertz is discernible. As all components correspond to dips in the baseband Spread Spectrum search grid as expressed in terms of integrated phase it is clear from the spectrum that the dip due to the maximal-length sequences being aligned is still the dominant stable point.

Investigation of the integrator output ( Figure 4.25 ) shows that the input waveform is locked at 25 milliseconds. However, the waveform is despread for short periods before this. This could indicate that the loop is not stable immediately on the acquisition of lock, but needs a few cycles to settle. However, the novel synchronisation loop is still able to achieve code-phase acquisition and code tracking at this signal-to-noise ratio with the other parameters as given in Table 4.4. The loop therefore, even under the influence of noise, provides a complete synchronisation solution in a simple configuration.

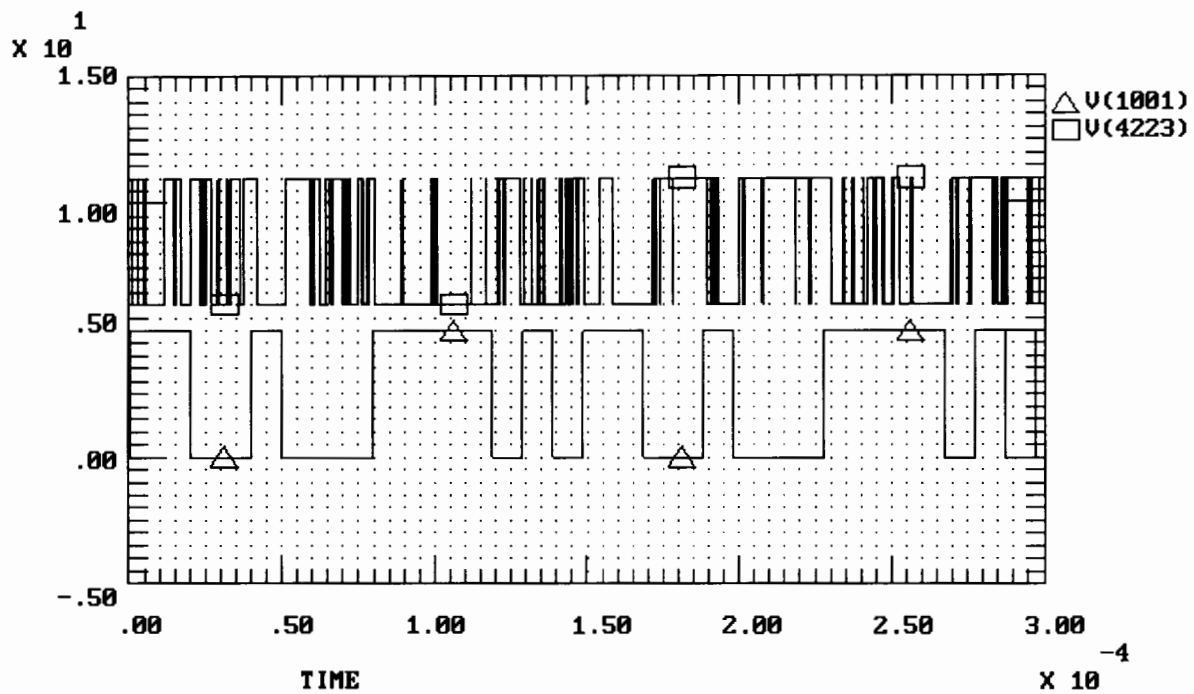


Figure 4.23 Graph of the transmitted, lower trace, and received, upper trace, maximal-length sequences. Errors are made by the receiver in its estimation of the received bit.

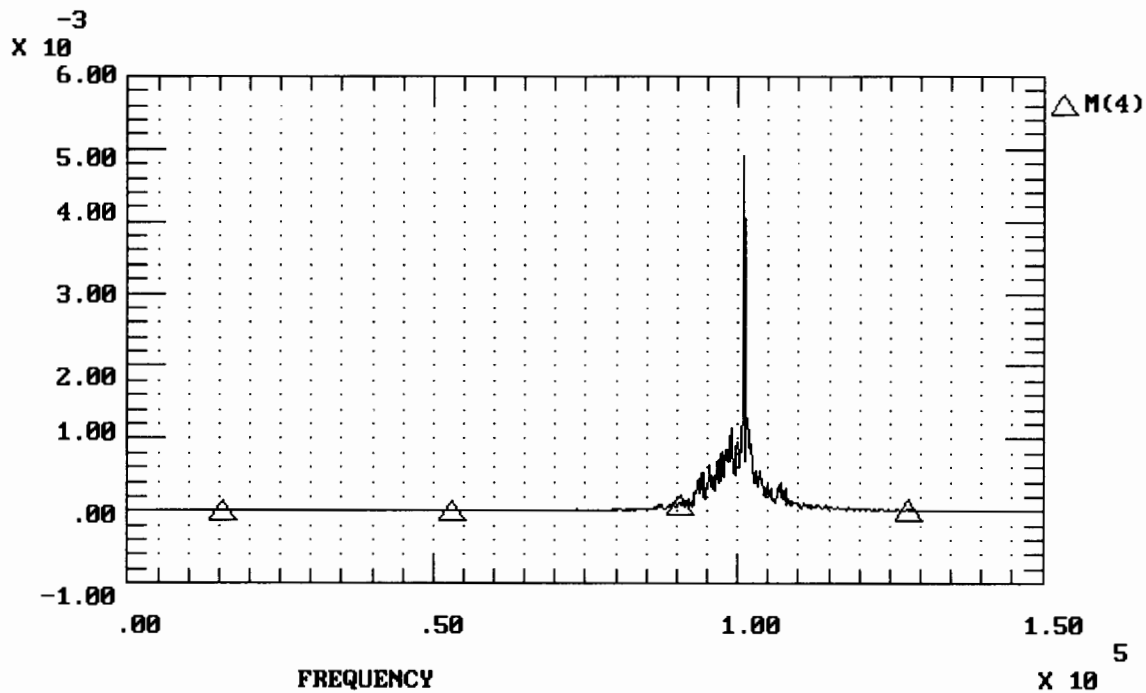


Figure 4.24 Graph of the spectrum of the output of the bandpass filter. A strong component exists at the transmitter's clocking frequency therefore allowing the loop to track the input.

### 4.3.3 The Novel Synchronisation Loop at a Signal-to-Noise Ratio of -2 dB.

Using the other parameters given in Table 4.4 the noise was increased to twenty Volts ( root mean square ). At this noise level the loop failed to achieve lock during the time window over which the novel loop was simulated.

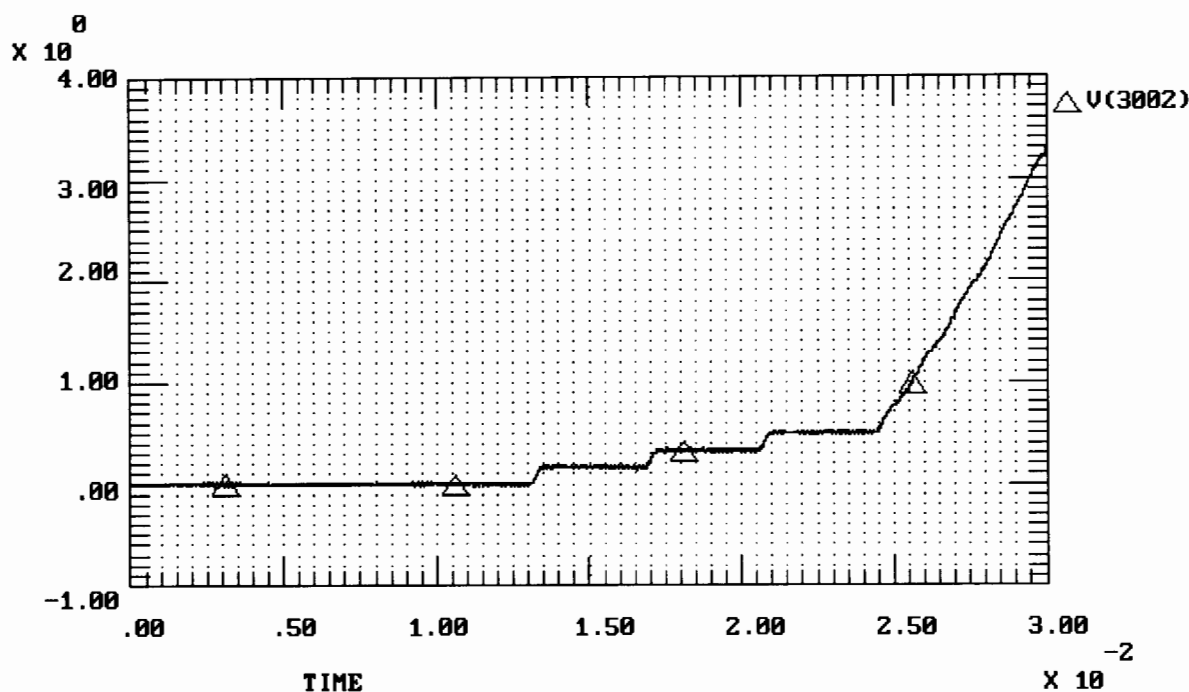


Figure 4.25 Graph of the output of the integrator. Synchronisation occurs at  $2.5 \times 10^{-2}$  although despreading of the transmitted waveform occurs before this time.

Figure 4.26 shows how the errors made by the receiver have increased to levels at which the loop cannot achieve lock. Noise spikes that are not suppressed by the input low pass filter trigger the comparator. This results in errors in the estimation of the bit edges as extra edges are generated. As it is these detected edges that drive the loop's adjustable clock the clock is incorrectly modulated. It therefore starts to track frequencies other than the transmitter's clocking frequency. This corresponds to entering an incorrect dip in the baseband Spread Spectrum search grid as expressed in terms of integrated phase. As this is an incorrect cell the transmitted waveform will not be despread when multiplied by the receiver's replica of the spreading waveform. This is demonstrated by the output of the integrator ( Figure 4.27 ). As no despreading occurs no data could be demodulated and no data transfer could take place. Other synchronisation methods would need to be used at this signal-to-noise ratio with the input bit estimation method as described in figure 4.19 and Table 4.4.

#### 4.3.4 Noise Performance of the Novel Synchronisation Loop with a Higher Order Input Low Pass Filter.

Using a resistor-capacitor low pass filter model as the input filter in the initial TESLA simulations gave insights into the effects of noise on the novel synchronisation loop with the differentiator in the input path. However, to accurately define the noise levels at which the loop can operate a higher order low pass filter at the input is needed. This is needed for two reasons. Firstly it allows one to more accurately define the system noise bandwidth and therefore to more accurately calculate the signal-to-noise ratio. Secondly a filter of this type will more closely resemble a 'brickwall' filter as it has a steep roll off in the frequency domain. This will limit the noise only to the bandwidth needed by the synchronisation loop. The effects of noise at higher frequencies than those needed to carry the transmitted information are suppressed, thereby giving an accurate indication of the noise levels the loop can operate in.

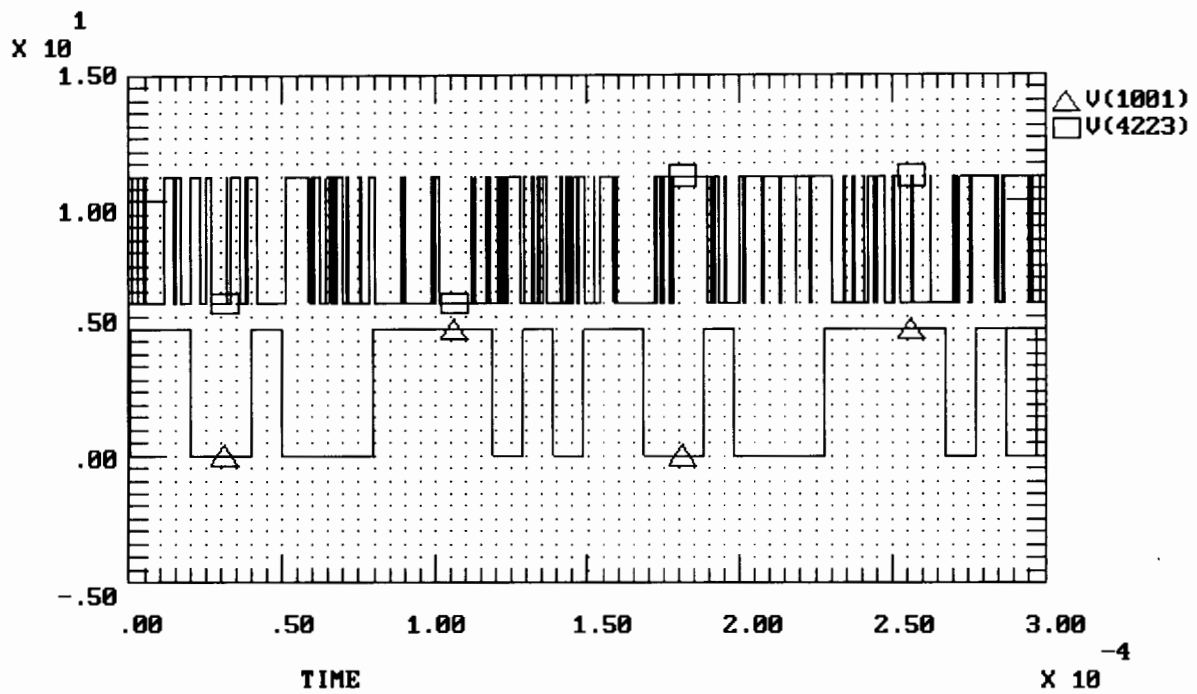


Figure 4.26 Graph of the transmitted, lower trace, and received, upper trace, maximal-length sequences. Errors are made by the receiver in its estimation of the received bit.

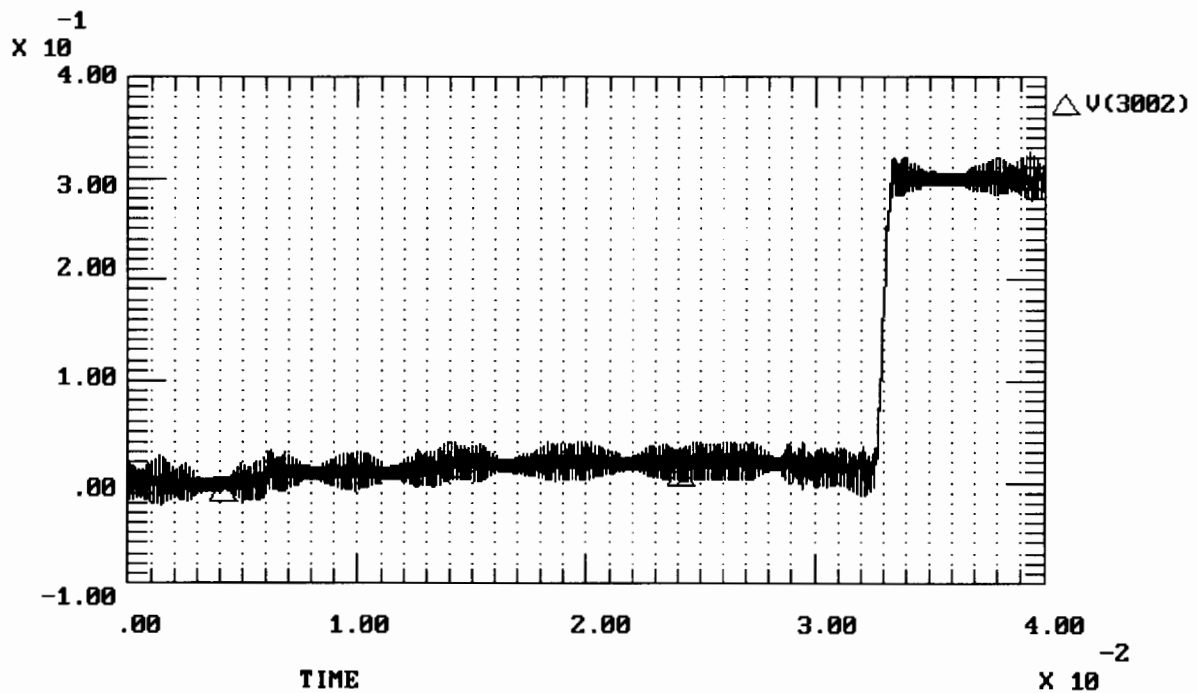


Figure 4.27 Graph of the output of the integrator. Despreading of the transmitted waveform does not occur due to the loop being unable to synchronise the two maximal-length sequences at this signal-to-noise ratio.

Table 4.5 gives the parameters to be used in the TESLA simulation of figure 4.19 in order to more accurately define the signal-to-noise ratios at which the novel maximal-length sequence synchronisation

Parameter	Simulation Value
Sequence Length	15
Centre of Band Pass Filter	100 kHz
Bandwidth of Bandpass Filter	7 kHz
Input Frequency	101 kHz
Code-phase Offset	7 bits
V.C.O. Centre Frequency	100 kHz
Maximum Possible Frequency Deviation	3 kHz
Bandwidth of the Loop Filter	1 kHz
Type of Input Low Pass Filter	Fourth Order Butterworth
Bandwidth of Input Low Pass Filter	101 kHz
Sampling Rate	$1 \times 10^{-7}$ sec

**Table 4.5** Table of the parameters used in the simulation of the novel synchronisation with a high order input low pass filter in a noisy environment.

loop can operate. The input low pass filter is modeled as a fourth order Butterworth filter with its bandwidth set to the transmitter's clocking frequency.

Figures 4.28 through 4.30, depicting the output of the integrator, indicate that the signal-to-noise ratios can be reduced further using a higher order input low pass filter than they were with the resistor-capacitor input low pass filter. Each error due to noise in estimating the incoming bit results in an extra bit edge. This in turn generates an instantaneous frequencies at frequencies other than the clocking frequency. Reducing errors therefore by suppressing the effects of noise at the input lessens the amount of incorrect frequency information at the input to the adjustable clock. The dips in the baseband Spread Spectrum search grid as expressed in terms of integrated phase that correspond to these incorrect frequencies therefore become shallower. The novel synchronisation loop is thus able to resolve the stable point due to the sequences being aligned and achieve lock.

However, at a signal-to-noise ratio -2.9dB the loop fails to achieve lock ( Figure 4.31 ). At his level the errors introduced by the noise become too large. However, the use of a higher order filter in the input allowed one to reduce the signal-to-noise ratio by approximately 3dB while still achieving lock in the time window over which the loop was simulated.

The novel maximal-length sequence synchronisation loop with the differentiator in the input path is able to achieve lock in a noisy environment. This is provided that the receiver does not make too many errors in the estimation of the incoming bit. This requires two conditions to be met. Firstly, the input must be adequately filtered so as to limit the noise only to the bandwidths required by the loop to resolve the transmitted waveform. Secondly, the receiver must be able to resolve the phase of the transmitter's carrier. The novel maximal-length sequence synchronisation loop is a baseband system. The transmitted waveform therefore needs to be demodulated down to baseband by the receiver before the novel synchronisation network can effect lock. If the phase of the transmitter's carrier cannot be accurately resolved the demodulation down to baseband will introduce a host of errors. This is a constraint placed on all baseband code-phase acquisition and tracking networks. This would include the RASE code-phase acquisition network and the full-time early-late tracking loop. In some cases the receiver's ability to resolve the phase of the transmitter's carrier, rather than the synchronisation network's operational noise level, might in fact determine the minimum signal-to-noise ratio at which lock can be achieved in a radio based communication system. As a phase-locked loop becomes unable to resolve an input phase at 0 dB this tends to be the limit at which baseband synchronisation networks can operate.

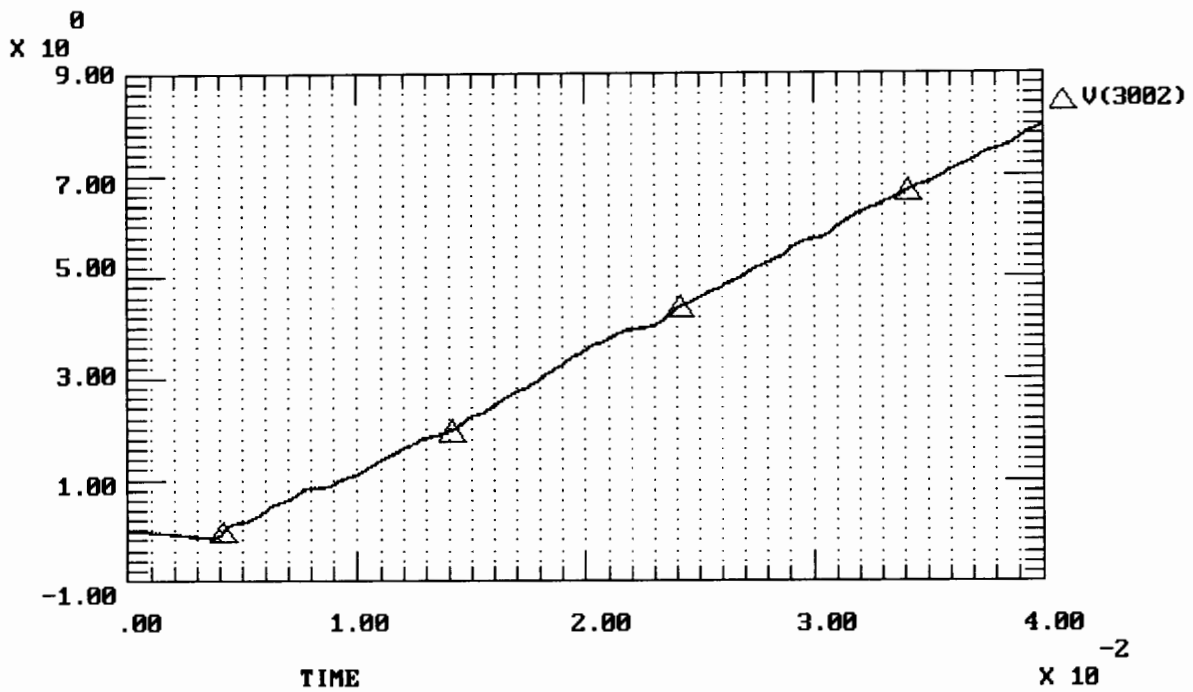


Figure 4.28 Graph of the output of the integrator at a signal-to-noise ratio of 0dB. The despreading of the waveform at  $0.4 \times 10^{-2}$  indicates that the novel synchronisation loop has achieved lock at this time.

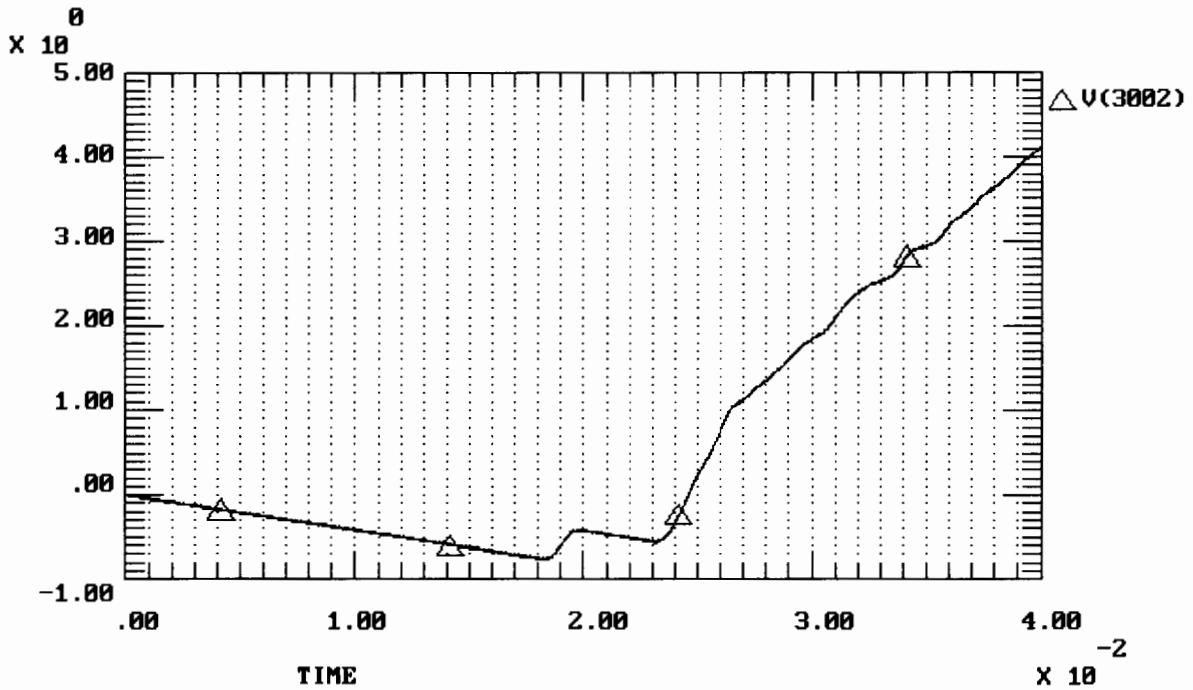


Figure 4.29 Graph of the output of the integrator at a signal-to-noise ratio of -1dB. The despreading of the waveform at  $2.4 \times 10^{-2}$  indicates that the novel synchronisation loop has achieved lock at this time.

#### 4.4 ADAPTATION OF THE NOVEL SYNCHRONISATION LOOP TO A DATA ENVIRONMENT.

Both the optimal ( Section 2.4.1 ) and the baseband full-time early-late tracking loop will not track a data modulated maximal-length sequence. Due to their code-phase discriminator slopes a swap in the

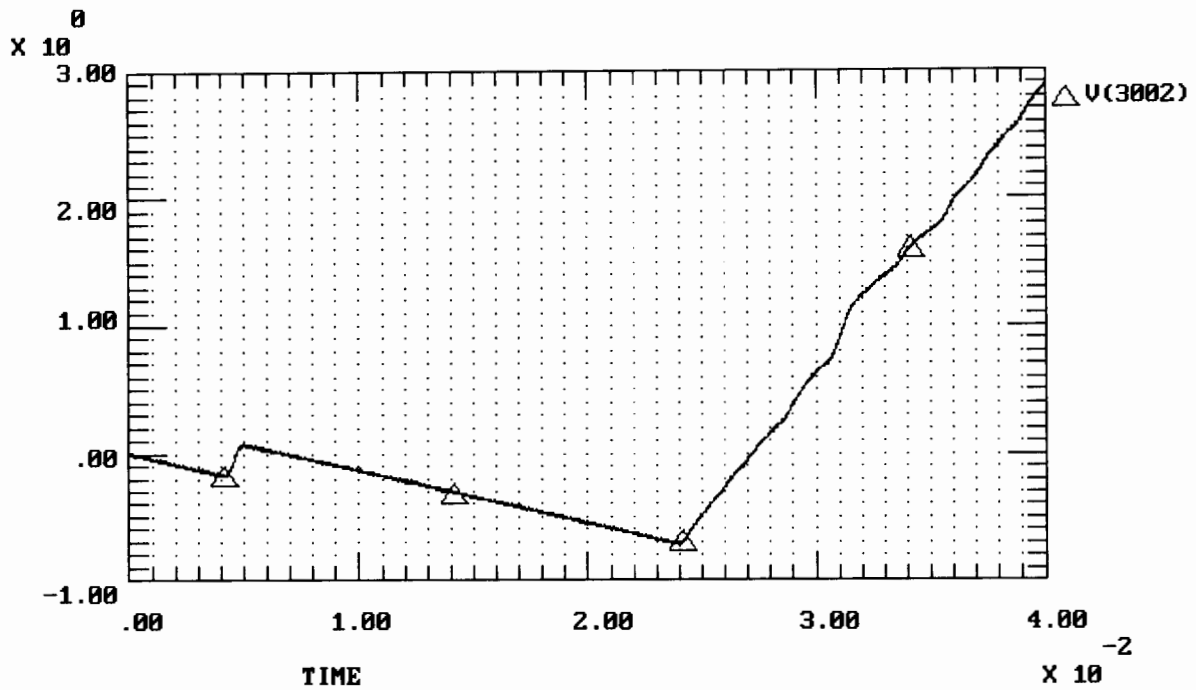


Figure 4.30 Graph of the output of the integrator at a signal-to-noise ratio of -2dB. The despreading of the waveform at  $2.4 \times 10^{-2}$  indicates that the novel synchronisation loop has achieved lock at this time.

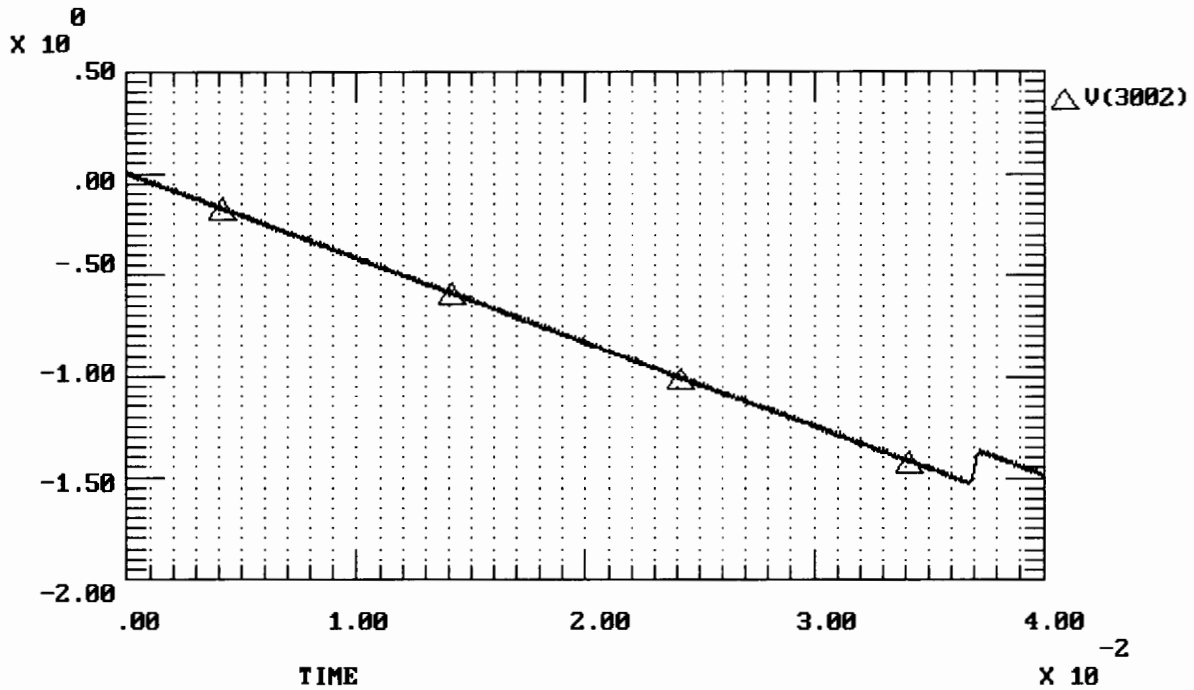


Figure 4.31 Graph of the output of the integrator at a signal-to-noise ratio of -2.9dB. The loop is unable to achieve lock and therefore despreading of the transmitted waveform does not occur.

sign of the input pseudorandom sequence will drive the loops in the wrong direction and therefore out of lock. As the novel synchronisation networks are based on variations on the maximal likelihood phase

discriminator they too suffer from this problem. A swap in the polarity of the input pseudorandom sequence will alter the sign of the spikes at the output of the maximum likelihood phase discriminator. The loop's adjustable clock will then have to alter its output phase by 180 degrees to track the input. This change would drive the loop out of lock.

It is possible to increase the number of times the maximal-length sequence repeats itself per data bit and allow the loop to reacquire lock each time an inversion due to data occurs. As, however, no despreading of the data would occur during the times that the loop was reacquiring lock the transmitter would not be transmitting useful data over these periods. This is therefore a wasteful solution. It would be preferable to allow the loop to maintain lock, even in a data environment.

A possible method of keeping the loop in lock in the data environment was identified. After lock the information generated by the inversion of the spike at the output of the maximal likelihood phase discriminator is redundant as this determines code-phase only. Once two sequences are aligned in code-phase it is only necessary to track the input clocking frequency to maintain lock. As the input path to the novel synchronisation network with the differentiator in the receiver frontend supplies this frequency information, a possible solution requires only that the loop maintain unipolar spikes at the input to the adjustable clock after lock. The effects of data would then no longer drive the loop out of lock.

Two possible methods of generating these unipolar spike were identified. Both require, however, that before the data is transmitted a string of either 'ones' or 'zeros' must be transmitted so as to allow the loop time to achieve lock. Such a design criterion is not uncommon in transmission systems when ambiguities at the receiver may need to be resolved before transmission takes place.

The two proposed methods differ in their approach of dealing with the modulation due to data and hence to the alterations that need to be made to the loop.

1. Method one removes all ambiguities due to data modulation by switching from differentiation to edge detection to detect the bit edges. Edge detection is a digital method of producing a positive going spike for every edge, either positive or negative going, in a digital bit stream. The spike are therefore unipolar. This however requires the removal of the feedback path from the novel synchronisation network as the edge detected spike would no longer be unipolar if multiplied by the receiver's replica of the spreading waveform.
2. Method two detects for an inversion from the original or 'correct' maximal-length sequence due to data modulation and then re-inverts the received waveform. This results in unipolar spike that will not incorrectly modulate the adjustable clock. This however requires that the loop monitor its condition in order to make a decision on the action to take when a change is detected.

The first method therefore disregards the effect of data modulation while the second uses data estimation techniques to correct for the effect of data modulation on the input pseudorandom sequence.

#### **4.4.1 Adaptation of the Synchronisation Loop to a Data Environment Through Edge Detection.**

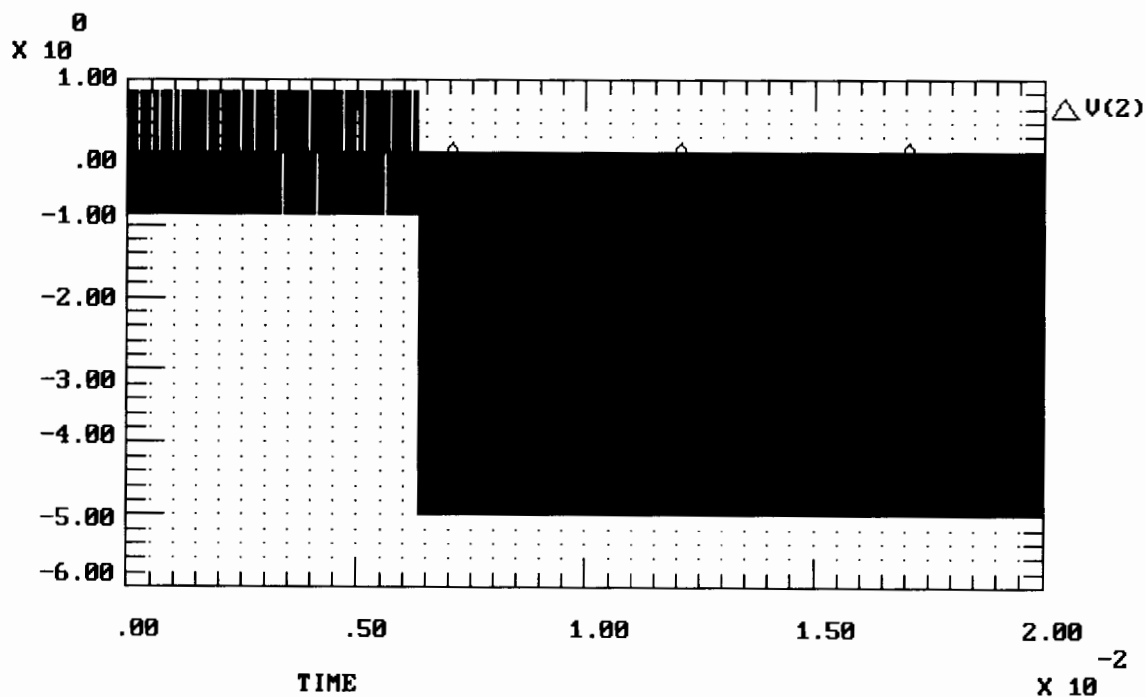
Figure 4.32 shows the block diagram of the novel synchronisation network with the differentiator in the input path adapted to a data environment by switching in an edge detector after lock. The transmitted pseudorandom sequence is modulated by a data source clocked at an integer division of the pseudorandom sequence. White additive Gaussian noise from the TESLA noise source is then added to the modulated waveform. The receiver then performs bit estimation through low pass filtering and comparison. The resulting waveform is then differentiated and edge detected ( Section 4.1 ). Which output is then used



Parameter	Simulation Value
Sequence Length	15
Centre of Band Pass Filter	100 kHz
Bandwidth of Bandpass Filter	7 kHz
Input Frequency	101.5 kHz
Code-phase Offset	4 bits
V.C.O. Centre Frequency	100 kHz
Maximum Possible Frequency Deviation	3 kHz
Bandwidth of the Loop Filter	1 kHz
Type of Input Low Pass Filter	Resistor-capacitor
Bandwidth of Input Low Pass Filter	99 kHz
Sampling Rate	$1 \times 10^{-7}$ sec

**Table 4.6** Table of the parameters used in the simulation of the novel synchronisation adapted to the data environment through edge detection.

detector. This operation removes the phase changes due to data modulation as evidenced by the unipolar spikes after the locking time.



**Figure 4.33** Graph of the output of the multiplexer. After the locking time of 6 milliseconds the input is edge detected. Phase changes due to data modulation are removed as the spike remain unipolar.

The novel synchronisation loop adapted to the data environment through edge detection was able to achieve and maintain lock. Figure 4.34 shows the data waveform and the despread transmitted waveform. The data is accurately demodulated by the synchronisation loop after the edge detection circuit is switched in at this signal-to-noise ratio. The novel synchronisation loop can therefore be adapted to the data environment through the use of edge detection.

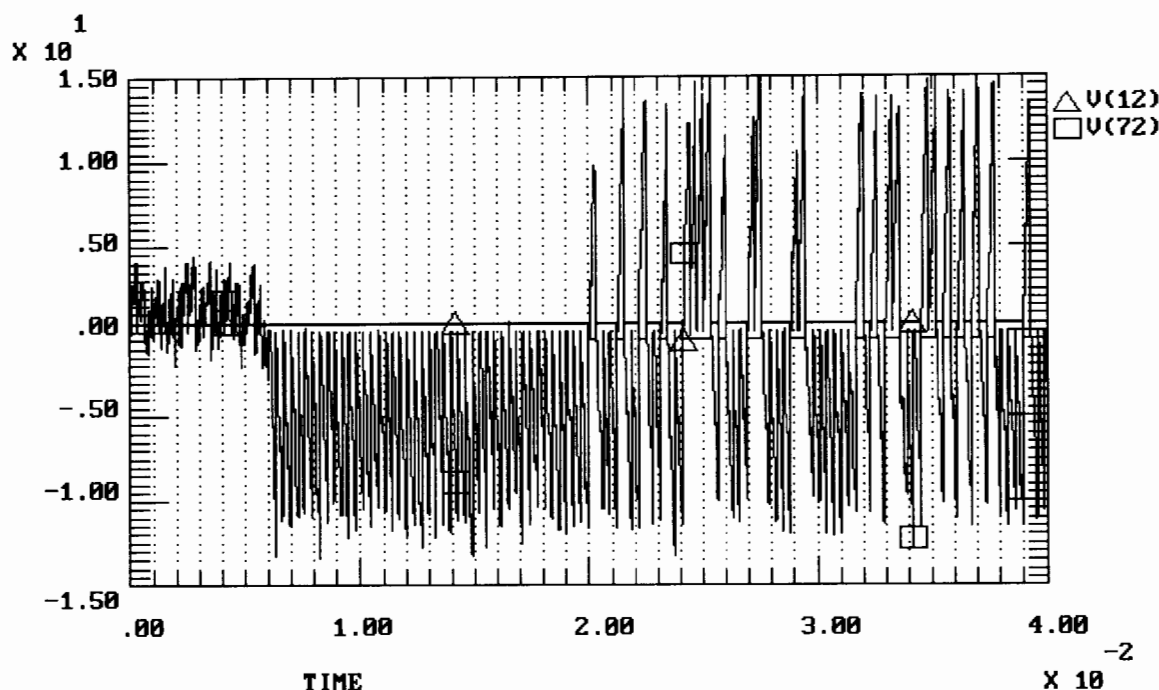


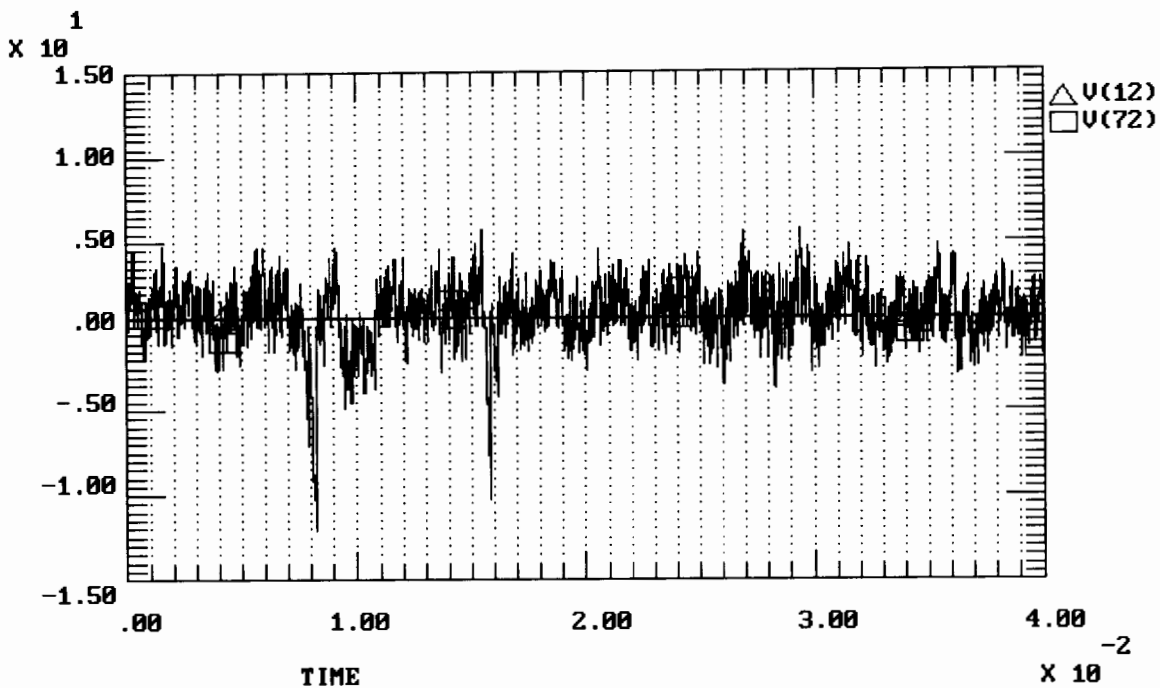
Figure 4.34 Graph of the data and despread waveforms. One can therefore use edge detection to adapt the loop to a data environment.

However, the use of edge detection does have implementation losses. Increasing the noise to 12 Volt ( root mean square ) results in the loop being unable to achieve synchronisation ( Figure 4.35 ). Although the data waveform is despread at  $0.8 \times 10^{-2}$  seconds and therefore the edge detection circuit switched in to the input, lock is quickly lost. This is approximately a 4dB loss when compared to the nondata modulated system with the same bit estimation circuit at the input.

#### 4.4.2 Adaptation of the Synchronisation Loop to a Data Environment Through Data Estimation.

This adaptation of the loop attempts to keep the spikes unipolar by counter acting the effects of data modulation. Figure 4.36 shows the basic block diagram of the loop. The detection circuitry will therefore re-alter the polarity of the spikes due to differentiation depending on the polarity of the transmitted data bit and noise.

The performance of the system is determined largely by the detection circuitry. The operations that the detection circuitry must perform are described by the flowchart in figure 4.37 and the logic table in figure 4.38. The upper block defines the operations during the code-phase acquisition period. During this time the detection circuitry must determine the 'type' of lock, either positive or negative going, the novel synchronisation network has achieved. Once the correct stable cell in the baseband Spread Spectrum search grid as expressed in terms of integrated phase has been determined the data estimation block is entered into. In this block the detection circuitry must compare the input to the natural or 'correct' locking waveform as determined in the code-phase acquisition period and if a change from this natural state due to data modulation is detected correct for it.



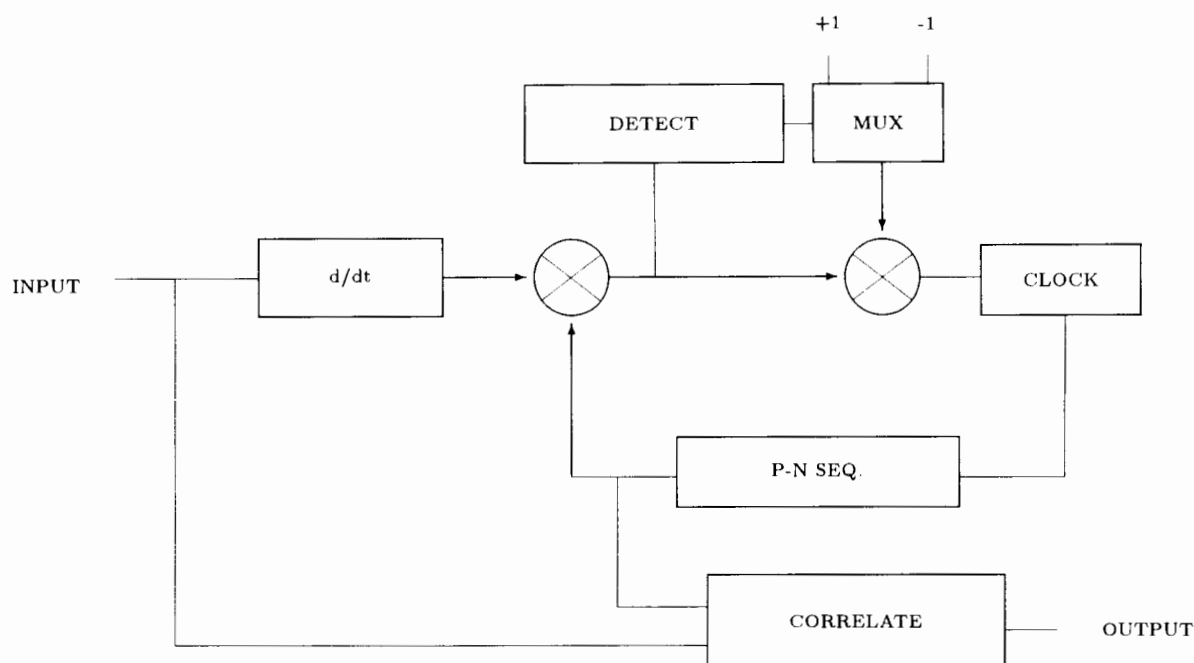
**Figure 4.35** Graph of the data and despread waveforms at a signal-to-noise ratio of -1dB. The loop fails to achieve lock.

Two types of detection circuitry were designed. The first used the spikes generated by the maximal likelihood phase discriminator to determine the loop conditions while the second used the product of the transmitted waveform and the receiver's replica of the spreading waveform.

### *Data Estimation Detection Circuitry Using the Spikes at the Output of the Maximal Likelihood Phase Discriminator.*

This implementation of the detection circuitry makes use of the spike at the output of the maximal likelihood phase discriminator to resolve the issues raised by data modulation. The detection circuit ( Figure 4.39 ) consists of two arms, one to deal with the negative going spike and the other with the positive going spike. Each arm consists of a comparator, to determine the sign of the spike, a binary counter ( CNTR ), to detect for a lock condition, and a memory element ( MEM ), to save the polarity of the locking condition.

The input spikes are compared to set positive ( POS ) and negative ( NEG ) levels to determine locking direction. The narrow pulses generated by the comparators trigger mono-stables ( M-S ) with pulse widths set to half that of a chipping period so as to allow the digital circuitry time to settle and time to change the state of the tracking loop. These pulses are then counted using binary counters ( CNTR ). In simulation the loop was said to be in lock when the most significant bit ( M ) of the counter was high. With a four bit counter and a fifteen bit length pseudo-noise sequence this means that half the sequence needed to have multiplied either positive or negative. Due to the correlation properties of pseudo-noise sequences this is a sufficient length over which to determine lock. The counters are inter-locked as a negative going pulse will reset ( RST ) the positive counter and visa-versa as a spike of opposite polarity is an indication that lock still needs to be achieved. The counters are gated with their own outputs so that they do not continue counting once lock has been detected and wrap around back to zero.



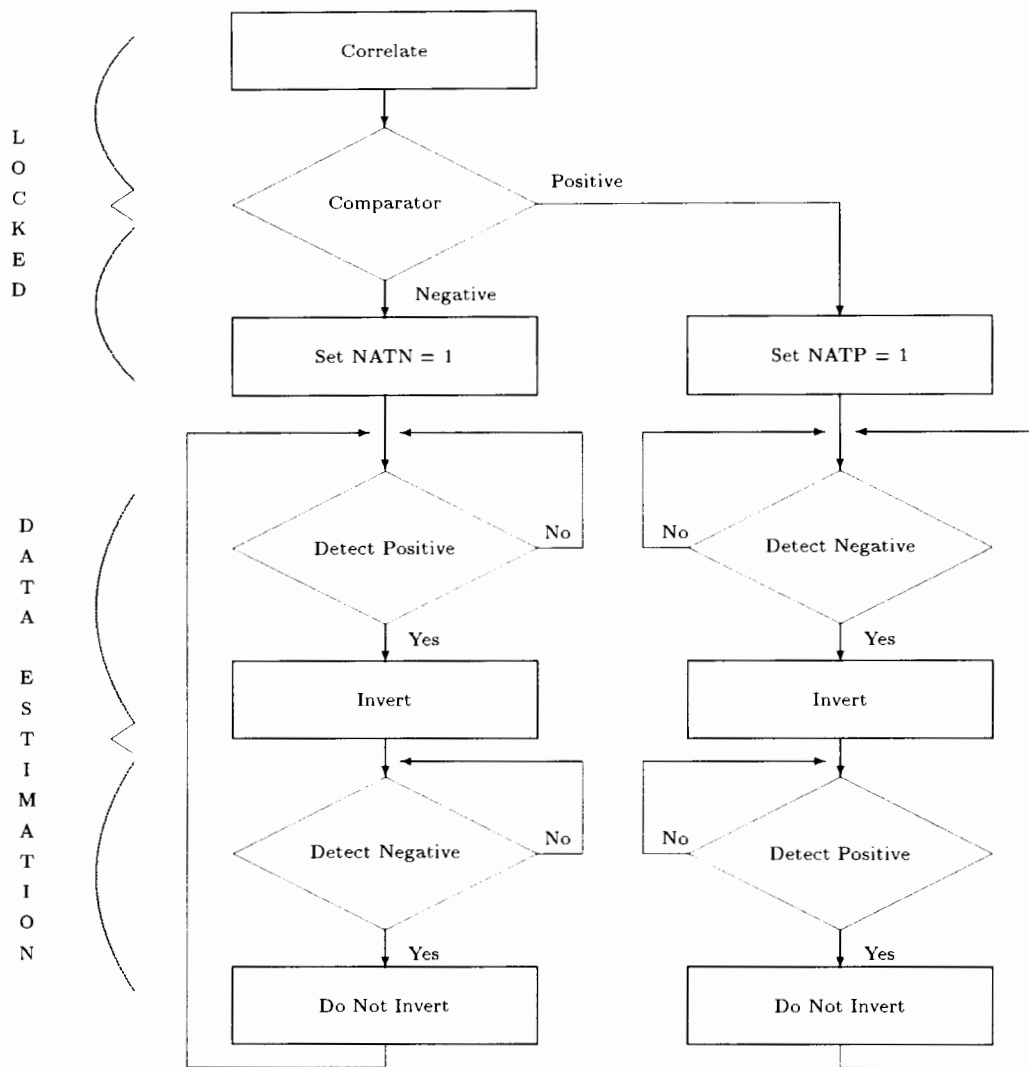
**Figure 4.36** Block diagram of the novel maximal-length sequence synchronisation adapted to the data environment through data estimation. The detection circuitry swaps the sign of the spike to keep the spike unipolar.

Once the tracking loop has locked, as evidenced by either the negative or positive counter's most significant bit going high, then that polarity is declared the natural locking direction. This information is then stored in a memory element ( NAT POS or NAT NEG ). One then AND-gates the natural locking direction with the least significant bit ( L ) of the opposite direction counter to detect when the loop should generate the SWAP signal. ( See Figure 4.38 ). The SWAP signal is the control flag deciding whether one must rectify the spikes or not. This keeps the input spikes to the clock circuitry unipolar and the loop remains in lock.

Figure 4.40 depicts the spike at the output of the maximal likelihood phase discriminator. The spike undergoes phase inversions due to the effects of data modulation on the transmitted waveform. However, the detection circuitry rectifies the spike correctly ( Figure 4.41 ) to generate unipolar spikes. These spikes should allow the loop to track the input code and thereby maintain lock.

The optimum receiver for data recovery is formed multiplying the input with the receiver's replica of the pseudo-noise sequence followed by an integrate and dump circuit. However, this assumes a higher level of synchronisation than that of the spreading sequence so that the integrate and dump circuit is triggered at the edge of a data bit period. There are circuits to achieve this, but for the purposes of the simulation the clock for the integrate and dump circuit was fed through from the transmitter. The data was recovered from the integrate and dump circuit ( Figure 4.42 ). Data estimation is therefore a viable method of adapting the novel synchronisation loop to the data driven environment.

However, the detection circuitry failed under even a moderate amount of noise. This is due to its inability to differentiate between an incorrect spike due to noise or an incorrect spike due to data inversion. Although the detection circuitry using the spikes at the output of the maximal likelihood phase detector has proved that data estimation is a viable technique in theory, a more robust detection circuit is needed.



**Figure 4.37** Flowchart of the operation of the detection circuitry for data estimation. The first block detects the type of lock, either positive or negative going. The second block determines when to invert the spike due to data modulation.

### *Data Estimation Detection Circuitry Using the Despread Input.*

As the overall effects of white Gaussian noise tend to diminish when integrated with respect to time, a detection circuit using the despread input waveform should be more robust than a circuit using the spikes at the output of the maximal likelihood phase detector.

Figure 4.43 depicts the detection circuitry ( Appendix 1 contains the source code ). The circuit uses the despread input waveform and the receiver clocking waveform to determine the loop conditions. Lock is detected by comparing the correlated input waveform to a set level. If despreading occurs due to lock the level will be exceeded. The comparator then triggers a mono-stable ( MS1 ) set at a period half that of the transmitter's bit period so as to allow the loop time to settle yet time to react before a change in

	NEGS.POSS	NEGS.POSS	NEGS.POSS	NEGS.POSS
NATN.NATP	X	X	X	X
NATN.NATP	X	1	X	0
NATN.NATP	X	X	X	X
NATN.NATP	X	0	X	1

$$\text{SWAP} = (\text{NATP AND NEGS}) \text{ OR } (\text{NATN AND POSS})$$

NEGS = NEGATIVE SPIKE  
 NATN = NATURAL NEGATIVE

POSS = POSITIVE SPIKE  
 NATP = NATURAL POSITIVE

**Figure 4.38** Logic table for the data detection circuitry.

input takes place. The output of mono-stable MS1 is stored in memory element DFF1. A 'one' at node Q thus means that lock has been detected.

The memory element, DFF1, has been designed to clock itself when the first  $0 \rightarrow 1$  transition takes place on the input and then to reject all other changes in input. The negative going edge needed to clock the D type flip-flop is generated by NOR gating the input with the  $\bar{Q}$  output. As  $\bar{Q} = 1$  before lock the first  $0 \rightarrow 1$  transition will generate the required negative going edge. AND gating the input to the  $\bar{Q}$  output means that when  $\bar{Q} = 0$  after lock no change in the input to the D type flip-flop will be possible.

Once lock has been detected it is necessary to determine the polarity of the input relative to the receiver. The despread waveform is integrated by an integrate and dump circuit. The reset signal for the integrate and dump circuit comes from mono-stable MS2. This mono-stable is set to provide a narrow pulse on every negative going edge of the receiver clock. Integration thus takes place over a complete bit period of the receiver's maximal-length sequence. If the received waveform is of opposite polarity to the receiver's replica of the spreading waveform this integration will generate a negative output, while if the two waveforms are of the same polarity a positive output will be generated. The output of the integrate and dump circuit is thus compared to a negative level to determine if the polarity of the input is different to that of the output. The output of the comparator is a digital level which is AND-gated with the Q output of DFF1. This is to detect if lock has been achieved and whether the data estimation block has been entered into. If lock has been achieved the output of the comparator is stored in memory element MS2 which is clocked at the receiver's clocking frequency. This memory element supplies the invert signal.

As was the case with the synchronisation network being adapted to the data environment through edge detection this detection circuitry only detects for sequences precisely aligned as an offset of one bit will not despread the input. Again however it is conceptually easy to include this case by using the same circuit but having as the input the input waveform despread by the receiver code delayed by one bit.



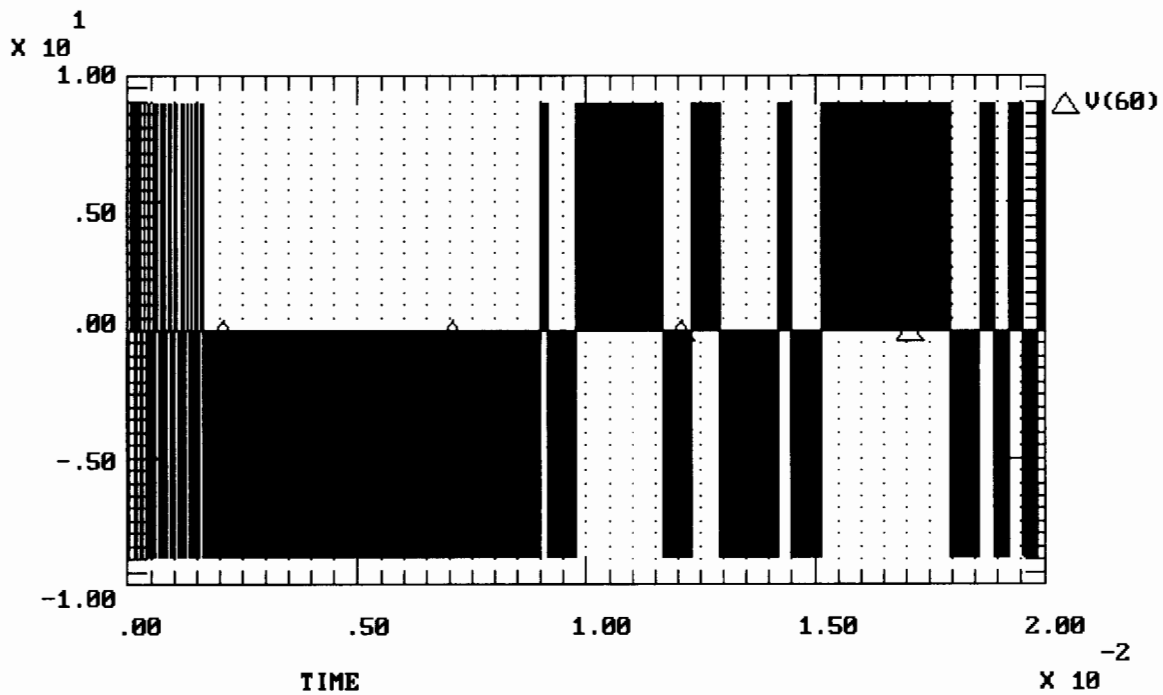


Figure 4.40 Graph of the spikes before inversion by the detection circuitry. The spikes are not unipolar due to the effect of data modulation.

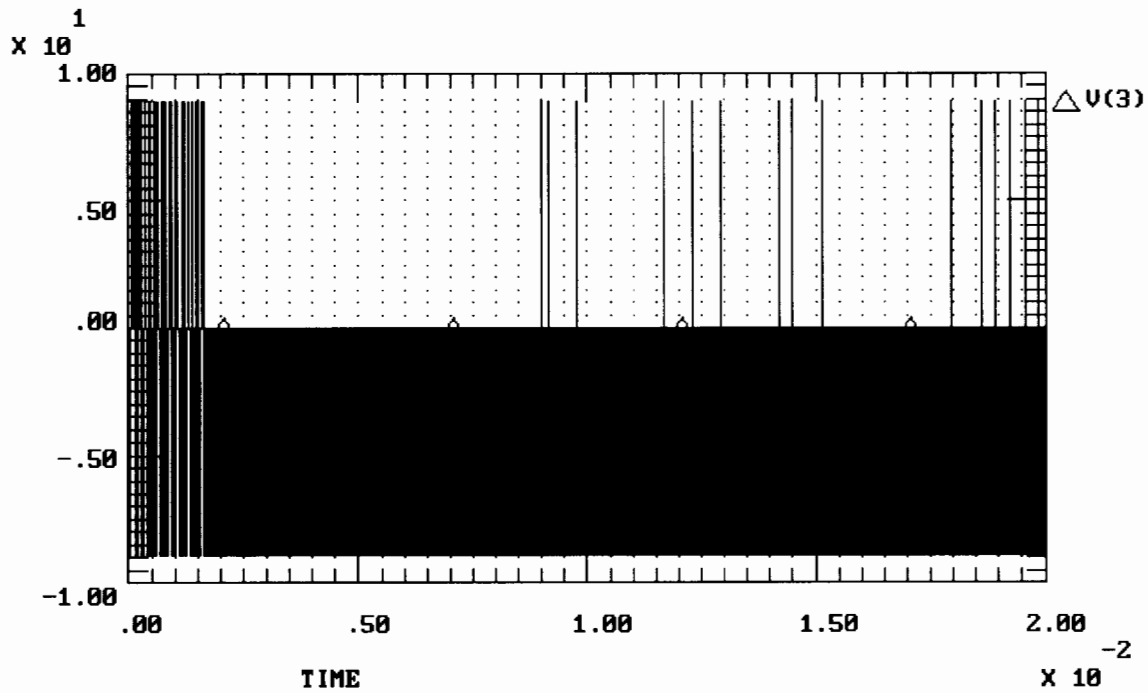


Figure 4.41 Graph of the spikes after inversion by the detection circuitry. The effect of data modulation has been removed as evidenced by the unipolar spikes.

the despread input waveform, and the comparator level a design decision. However, the loop is stable and accurately despreads the transmitted data.

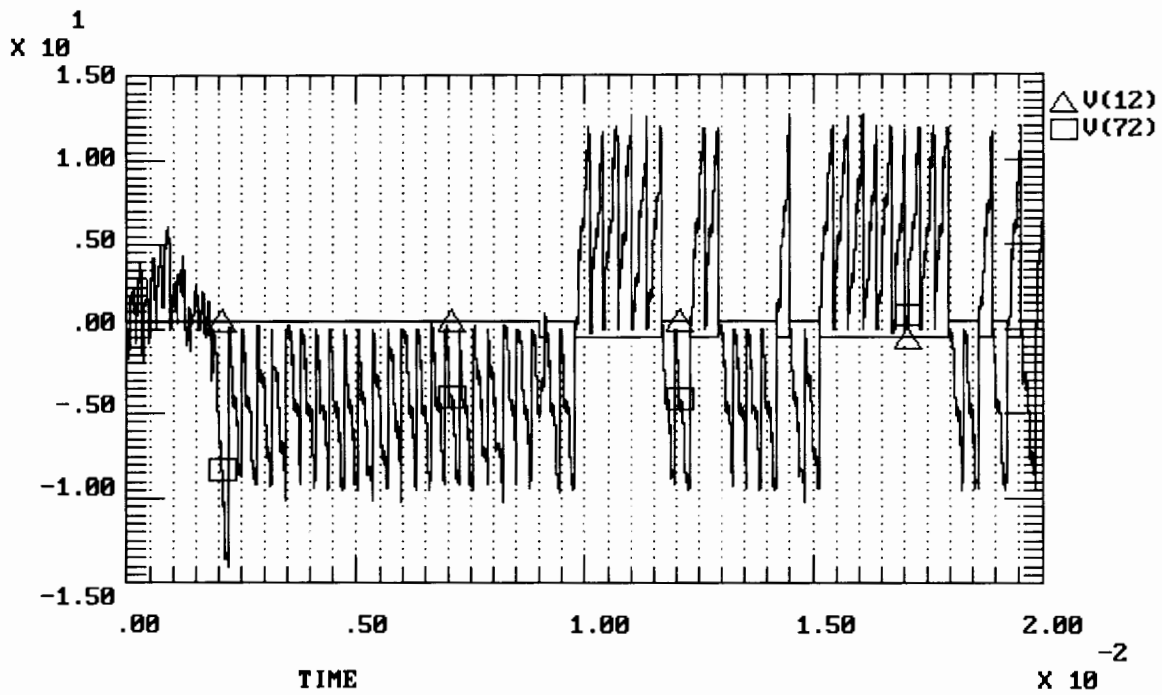


Figure 4.42 Plot of the data and despread waveform. The detection circuitry has thus accurately rectified the spikes so as to allow the novel synchronisation circuitry to maintain lock.

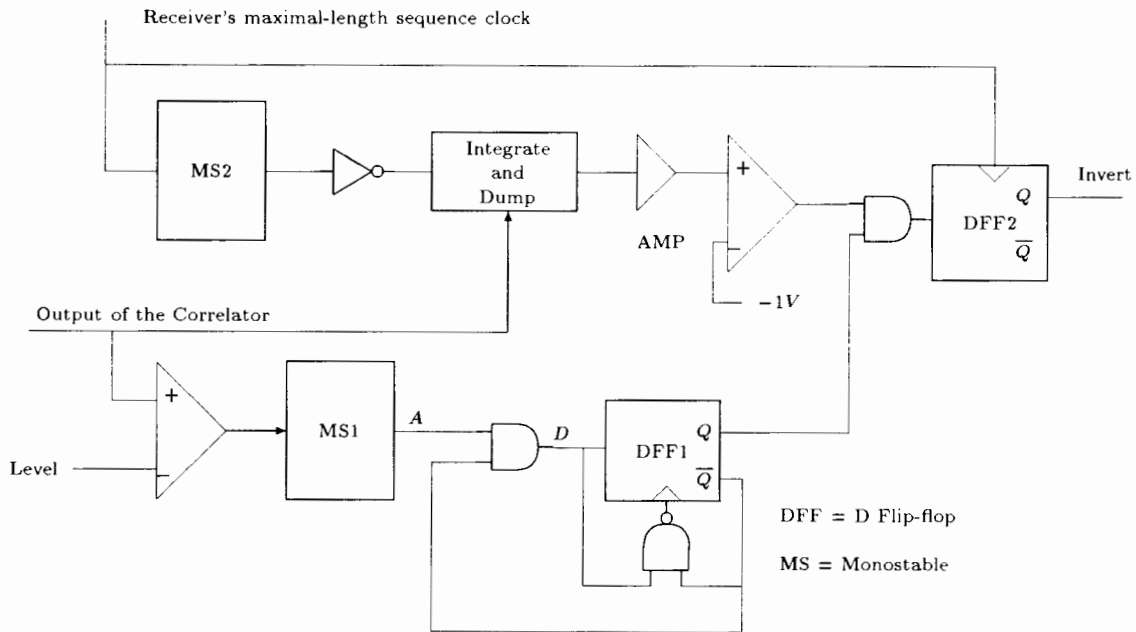


Figure 4.43 Block diagram of the detection circuitry using the despread input and receiver clocking waveform. The memory element DFF1 indicates that lock has been detected while memory element DFF2 determines whether the input waveform should be inverted so as to counter act the effects of noise.

The detection circuitry using the despread input waveform is more stable than the detection circuitry using the spikes at the output of the maximal likelihood phase detector. Figure 4.47 depicts the data

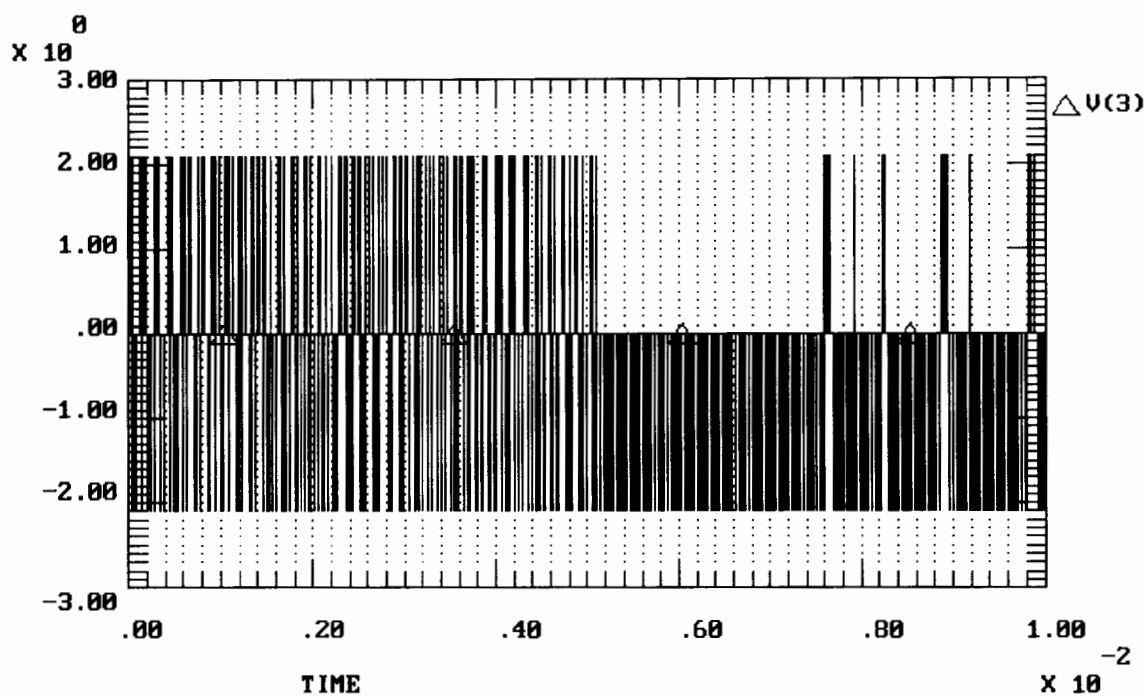


Figure 4.44 Graph of the spikes for the noiseless case. The detection circuitry keeps the spikes unipolar except for those short periods over which integration must take place.

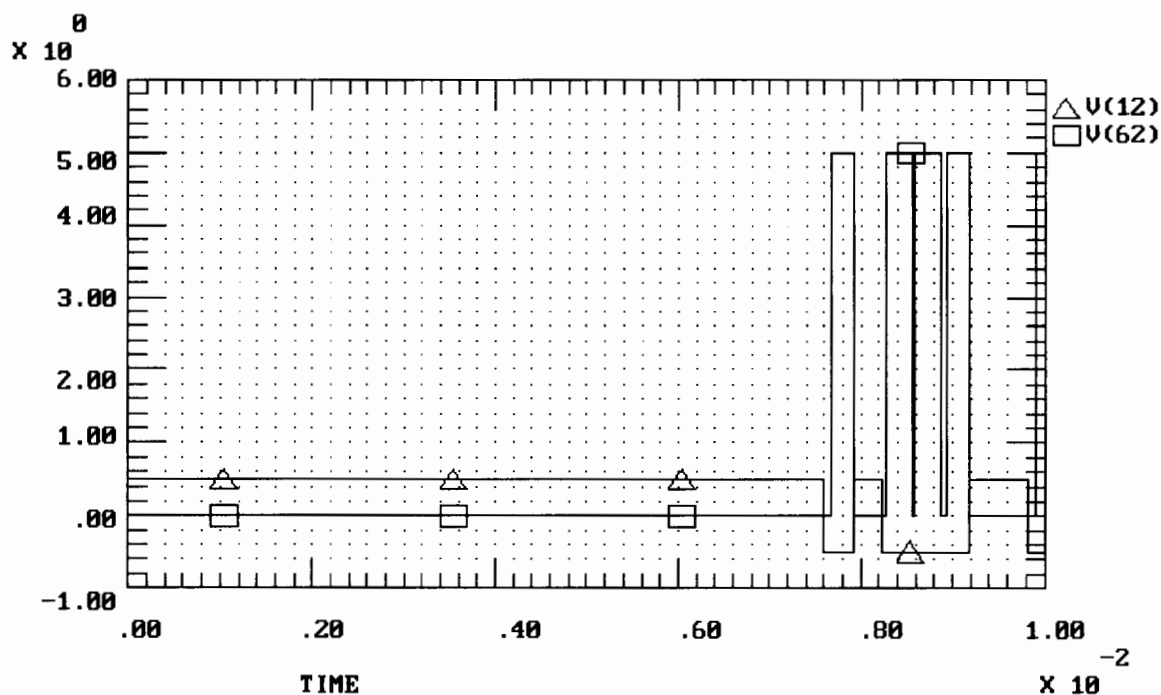


Figure 4.45 Graph of the data ( V(12) ) and invert ( V(62) ) signals for the noiseless case. The selfnoise is generated as the invert signal is not the inverse of the data signal over the complete data bit period.

and invert signals for a signal-to-noise ratio of 8.4 dB. The data signal is correctly estimated at this

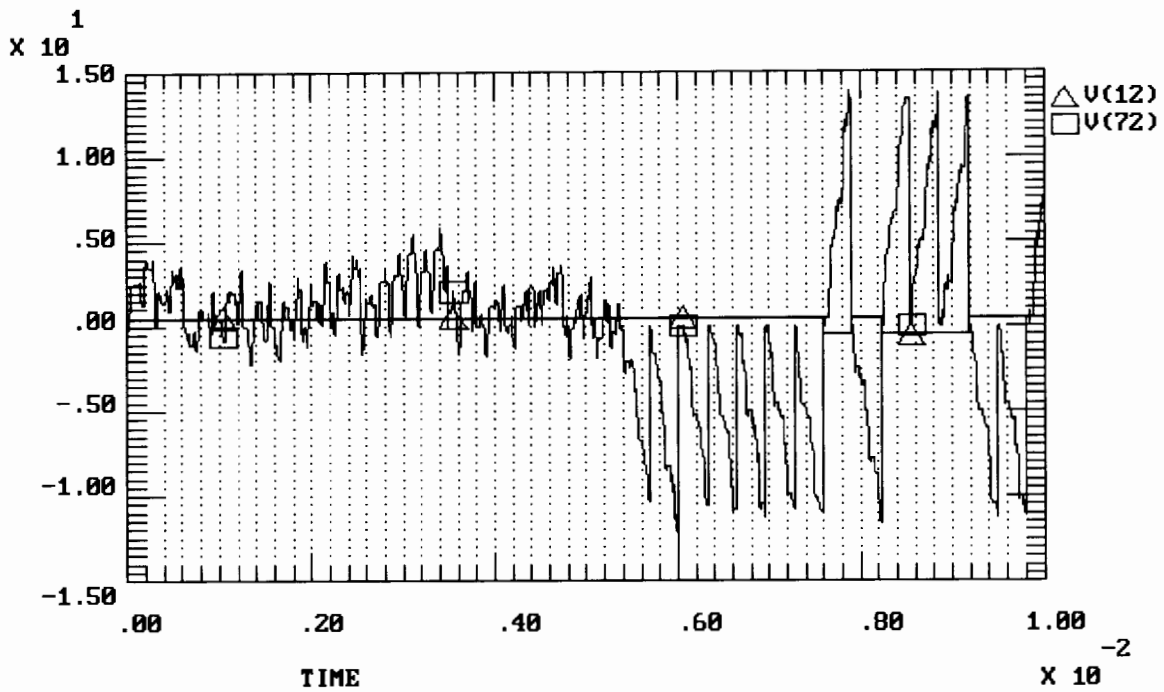


Figure 4.46 Graph of the output for the noiseless case.

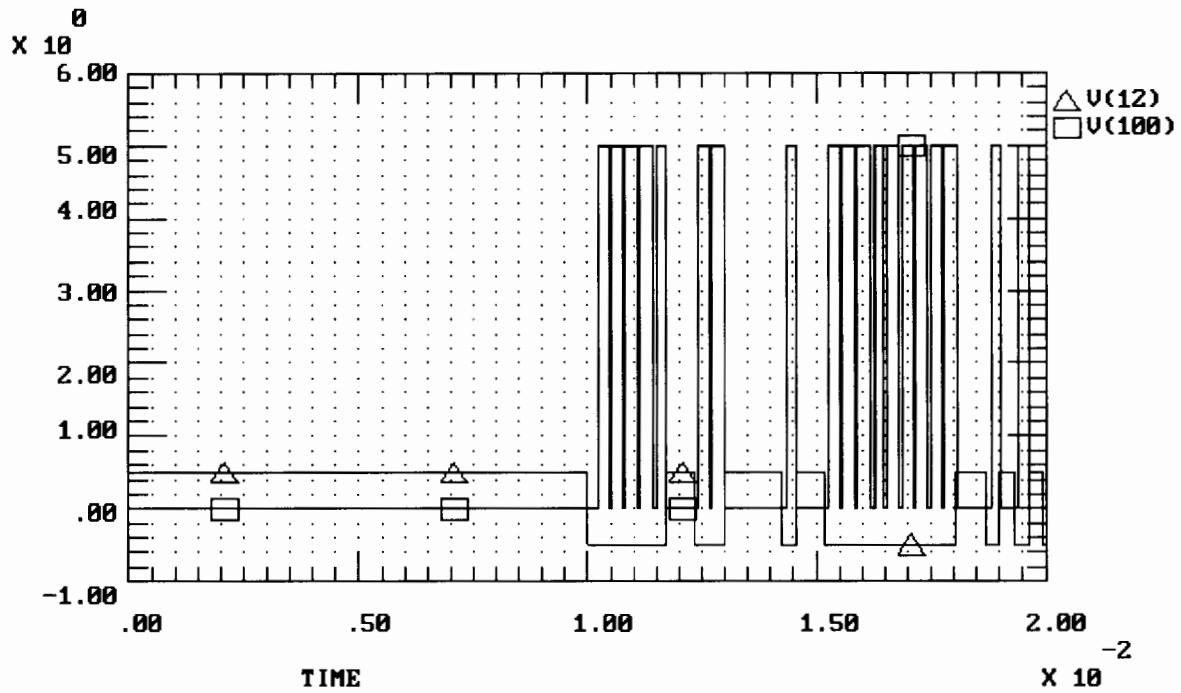
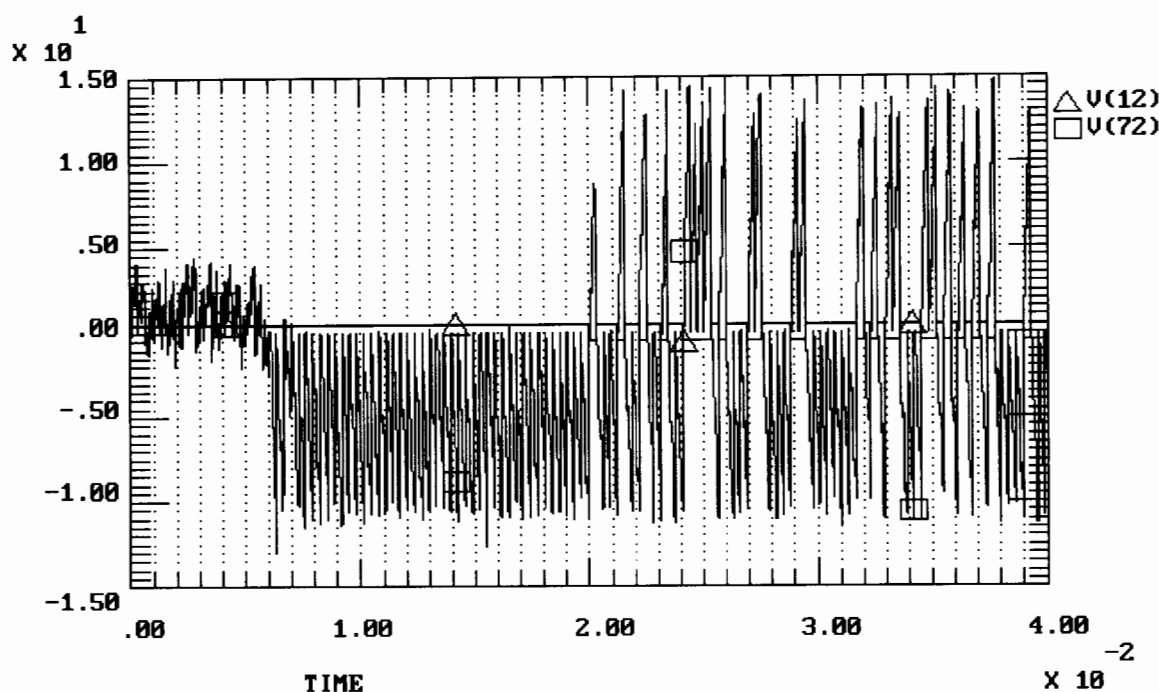


Figure 4.47 Graph of the data ( V(12) ) and invert ( V(100) ) signals at a signal-to-noise ratio of 8.4 dB. The invert signal correctly inverts the data signal except for the self noise generated at data bit edges.

noise level except for the characteristic self noise. The spike at the input to the adjustable clock should therefore not cause it to lose lock.



**Figure 4.48** Graph of the data and despread transmitted waveform at a signal-to-noise ratio of 8.4 dB. The detection circuitry has kept the loop stable as lock is not lost.

The data is correctly despread at the output of the correlator at this signal-to-noise ratio ( Figure 4.48 ). The detection circuitry using the despread waveform and the receiver clocking waveform to monitor the synchronisation loop's condition therefore keeps the loop stable. The loop is therefore more robust than that using the detection circuitry based on the spikes at the output of the maximal likelihood phase detector which failed at this signal-to-noise ratio.

The signal-to-noise ratio was further decreased to the level at which the adaptation of the loop to the data environment through edge detection failed. Investigation of the data and invert signals ( Figure 4.49 ) show that the the detection circuit is still correctly estimating the incoming data except at the bit edges. Figure 4.50, depicting the data and despread transmitted waveform, shows that the loop is still able to track a data modulated waveform at this signal-to-noise ratio.

The loop adapted to the data environment through data estimation by using the despread waveform to monitor the loop's condition is therefore a more robust method of adaptation than edge detection. This is due to the differing effects noise will have on the resulting spike at the output. In the edge detection environment every error will produce a positive going spike at the input to the adjustable clock. This effectively skews the statistical noise distribution and results in the noise having a D.C. value at the input to the adjustable clock. The long term effects of noise in this configuration are therefore cumulative. The data estimation circuit however just inverts the spike. The errors due to noise are thus not skewed as the spike may still go either positive or negative. As the errors are caused by Gaussian noise they should have a Gaussian distribution. The long term effects of errors should therefore be small as they cancel each other out over time. The implementation losses due to the data estimation technique are therefore smaller than those due to the adaptation of the loop through edge detection.

The signal-to-noise ratio was further decreased to 0 dB. Figure 4.51 depicts the data and despread transmitted wave. Lock is achieved at 10 milliseconds. However, it is lost then regained. The loop is thus at the edge of its tracking range in this configuration.

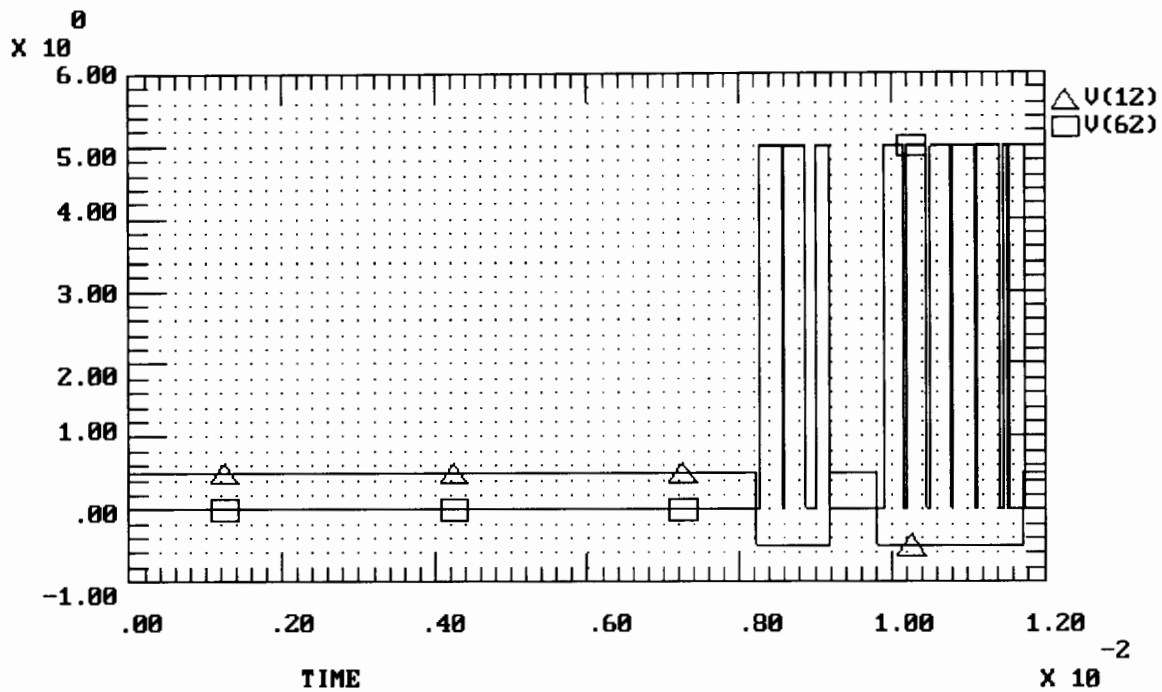


Figure 4.49 Graph of the data ( V(12) ) and invert ( V(62) ) signals at a signal-to-noise ratio of 3 dB. The invert signal correctly inverts the data signal except for the self noise generated at data bit edges.

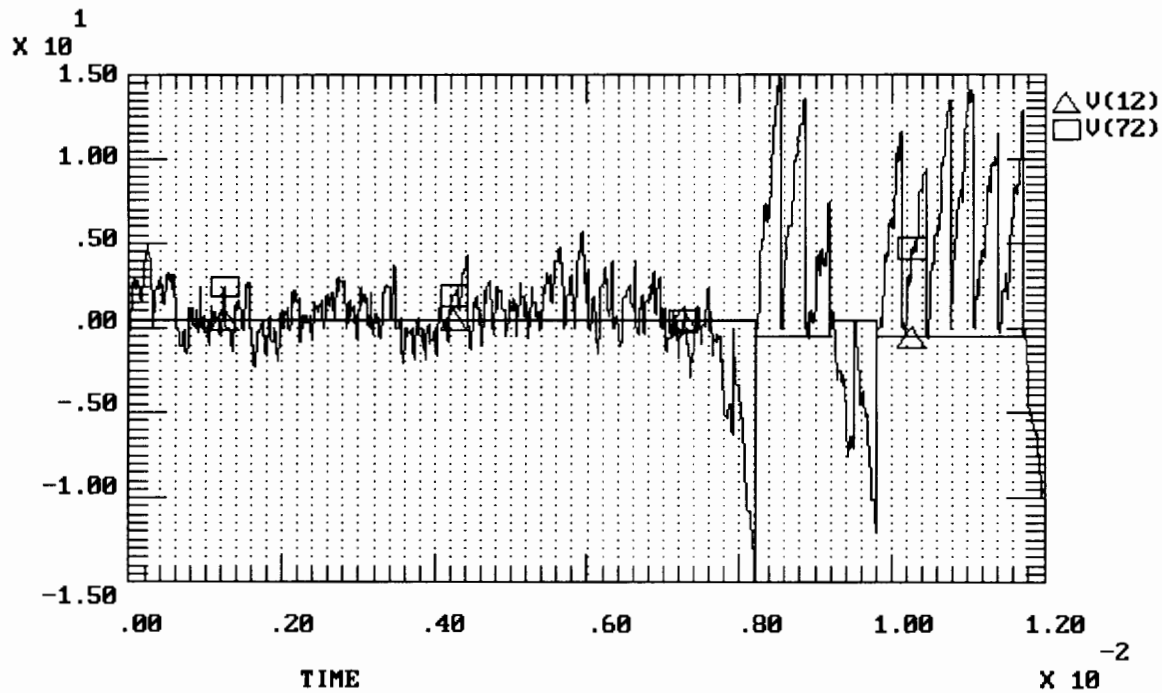
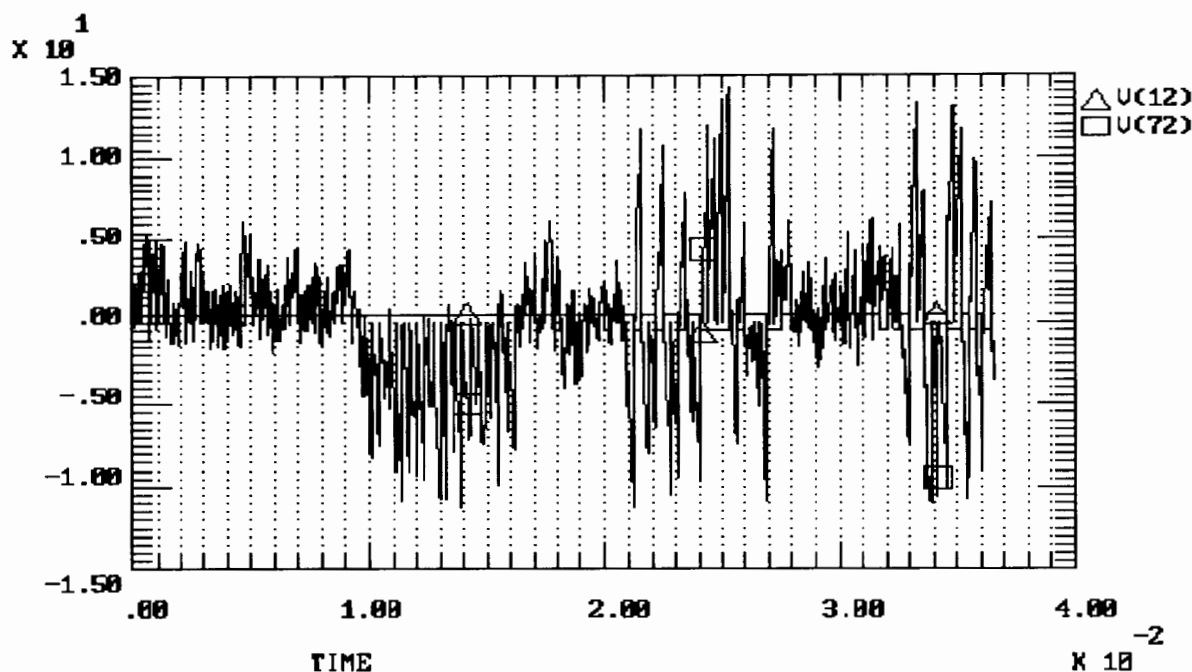


Figure 4.50 Graph of the data and despread transmitted waveform at a signal-to-noise ratio of 2.4 dB. The detection circuitry has kept the loop stable as lock is not lost.

The loop thus has an implementation loss of approximately 2 dB when compared to the loop under conditions of no data modulation of the transmitted waveform. This is better than the 4 dB loss suffered by the loop adapted through edge detection. Using data estimation with the detection circuitry using



**Figure 4.51** Graph of the data and despread transmitted waveform at a signal-to-noise ratio of 0 dB. The loop in this configuration is at the lower bound of the signal-to-noise ratios at which the novel synchronisation loop can operate as lock is achieved and then lost.

the despread waveform and receiver clocking waveform therefore appears to be good solution to the data problem.

#### 4.5 COMPARISON OF THE NOVEL SYNCHRONISATION NETWORK WITH THE DIFFERENTIATOR IN THE INPUT PATH WITH OTHER COMMON SYNCHRONISATION CIRCUITS.

Placing the differentiator in the novel synchronisation network's frontend resulted in a loop which offered a complete maximal-length sequence synchronisation solution in one simple configuration. However, it should not be considered a replacement for all synchronisation circuits. The criterion that need to be considered include:

1. **LOOP COMPLEXITY.** The novel synchronisation loop provides complete code acquisition and tracking in a simple circuit. The common methods of synchronisation presented in this text such as RASE and serial search are more complex in that they need a further level of synchronisation in order to allow for code tracking. This adds to the complexity of these synchronisation solutions.

As two stable points exist on the baseband Spread Spectrum search grid as expressed in terms of integrated phase at either a one or no bit offset, the possible code phase available at the transmitter is halved. The other methods do not suffer from this ambiguity. However, the complexity of the novel synchronisation circuit is not increased as a single extra register will allow despreading of the input to be effected.

2. **LOCKING TIMES.** An accurate comparison of locking times is not strictly possible for two reasons. Firstly, the times quoted for the common code-acquisition circuits are the average times taken to resolve the correct code-phase, not to synchronisation. The times would have to be lengthened to include the synchronisation times taken by the tracking loop. Secondly, the common methods are deterministic in their approach to achieving lock while the novel synchronisation loop follows a chaotic path through the baseband Spread Spectrum search grid.  
As show earlier the novel synchronisation loop's locking time is approximately three times longer than those of RASE and serial search techniques for the noiseless case.
3. **NOISE PERFORMANCE.** The novel synchronisation loop is able to achieve lock at signal-to-noise ratios in the order of -2 dB. It is however a coherent synchronisation scheme as the received waveform must be demodulated before lock can be effected. This means that the loop should only be considered in relation to systems where the RASE code acquisition loop and the full-time early-late tracking loop might be used. At lower signal-to-noise ratios a non-coherent synchronisation scheme must be used. It should be noted that Spread Spectrum systems do tend to be used at signal-to-noise ratios lower than -2 dB so as not to interfere with adjacent users. The noise performance of the novel maximal-length sequence synchronisation network is therefore a major disadvantage.
4. **DATA TRANSMISSION.** The novel synchronisation loop suffers from some implementation loss when adapted to the data environment. Furthermore, the adaptations required do add a little complexity to the loop.

The novel synchronisation loop should provide a synchronisation solution for a coherent direct sequence Spread Spectrum system. However, as such systems require a high signal-to-noise ratio they tend to interfere with adjacent bandwidths. For this reason non-coherent systems tend to be used. The novel maximal-length sequence synchronisation network is not a viable alternative in such systems.

Furthermore, the novel synchronisation loop has not been tested at typical sequence lengths. It seems unlikely that the results can simply be extrapolated to include longer maximal-length sequences. However, it is felt that the loop could be adapted in case of failure at longer run lengths.

---

# DESCRIPTION OF THE PRACTICAL IMPLEMENTATIONS OF THE VARIOUS COMPONENTS IN THE SYNCHRONISATION LOOP.

Insight into the loop's characteristics and dynamics has been given by the simulation of the novel synchronisation network. However, the system must be validated on the bench. During the simulation of the novel synchronisation loop one was limited in the implementations of the different loop components by the models that TESLA provided as part of its simulation environment. No such restrictions apply when proto-typing.

Various practical implementations were investigated for both the differentiator block and the adjustable clock. The various circuits offered certain advantages and disadvantages as regards their applications in the novel maximal-length sequence synchronisation network.

## 5.1 SIGNED EDGE DETECTION AS AN ALTERNATIVE TO DIFFERENTIATION.

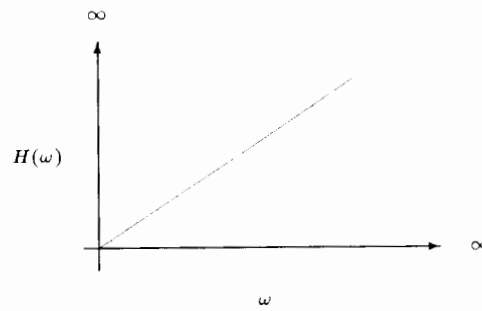
Differentiation tends to be avoided in electronic circuits as this operation may increase noise levels, especially those components at high frequencies. This is illustrated by the transfer function of the ideal differentiator ( Figure 5.1 ). The ideal differentiator has an infinite bandwidth with a large gain at higher frequencies. Any high frequency noise is therefore amplified by this circuit. This may make the noise levels unacceptable in the rest of the circuitry and lead to errors. For this reason one does not implement an ideal differentiator, but includes an integrator component in the transfer function to reduce the output bandwidth ( Figure 5.2 ). This implementation limits the output bandwidth of the differentiator and therefore also suppresses high frequency noise.

The most common practical implementation of a differentiator uses an active device, as shown in figure 5.3. The differentiator's time constant is given by  $\tau_1 = R_1C_2$ . The integrator's time constant is given by  $\tau_2 = R_2C_1$ . The circuit will therefore approximate a differentiator between the frequencies given by the time constants. At frequencies above that of the time constant of the integrator the frequency response flattens, therefore suppressing the noise at these frequencies.

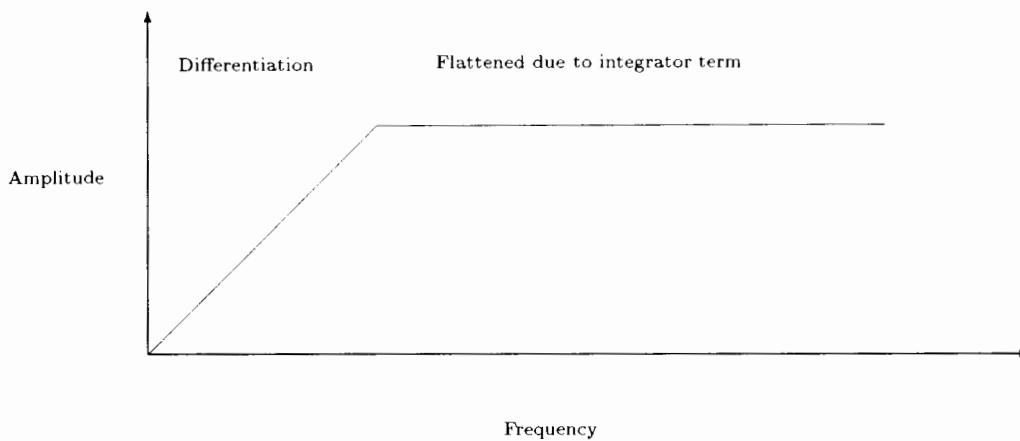
In the practical differentiation of a wide band signal therefore, the bandwidth over which differentiation occurs is traded off against noise suppression at higher frequencies. In some cases however one may wish to differentiate a wideband signal. This is the case in the novel synchronisation loop where the input is a pseudorandom sequence. For the purposes of demonstration however a square wave input is assumed.

Mathematically, the ideal differentiation of a square pulse  $u(t - T)$ , can be shown to be

$$\frac{d}{dt}u(t - T) = \delta(t) - \delta(t - T)$$



**Figure 5.1** Graph of the frequency response of the ideal differentiator. The response indicates that high frequencies will be amplified.

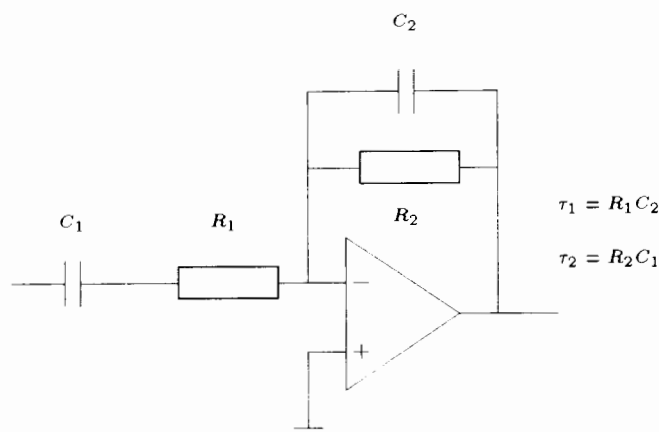


**Figure 5.2** Graph of the frequency response of a practical differentiator. The inclusion of an integrator term in the transfer function results in a narrowing of the output bandwidth. High frequency noise is therefore no longer amplified.

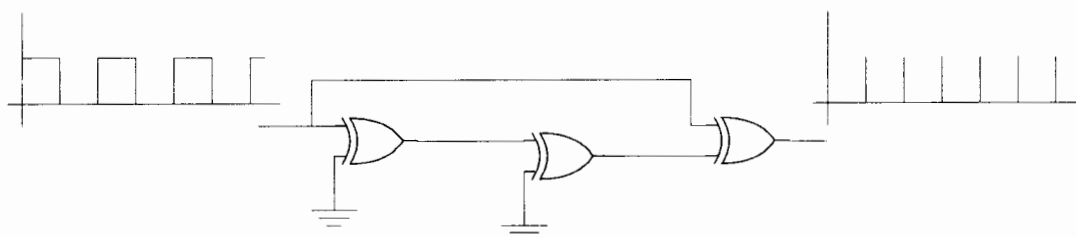
Therefore, to truly differentiate a square wave one needs to generate diracs  $\delta(t)$ , an operation which requires infinite bandwidth. The practical differentiation circuit is therefore not ideal for this application. Possible alternatives to differentiation were therefore investigated.

Edge detection provides a positive going spike for every edge of a square wave ( Figure 5.4 ). Assuming that the input unipolar square wave swings between 0 and 5 volts, the delays introduced by the exclusive OR gates tied to ground cause a glitch at the output. A positive going spike is therefore generated for every edge.

As the circuit is based on level comparison the edge detector is a low noise implementation. The output only changes when a large change in the input occurs. Furthermore digital circuits tend to have fast switching times. This circuit therefore does not suffer from the slew rate problems that the opamp implementation of the differentiator might have. The most important advantage of this circuit is that the delay value can be set according to the application. This allows the designer to reduce the effects of the harmonics by making the delay longer. This is due to the relationship between the time and the frequency domain. A periodic train of square pulses in the time domain translates to a train of impulses in the frequency domain that are weighted by a  $\frac{\sin(x)}{x}$  function. As the pulses are widened in the time



**Figure 5.3** Circuit diagram of a practical implementation of a differentiator. The design includes an integrator to suppress high frequencies.

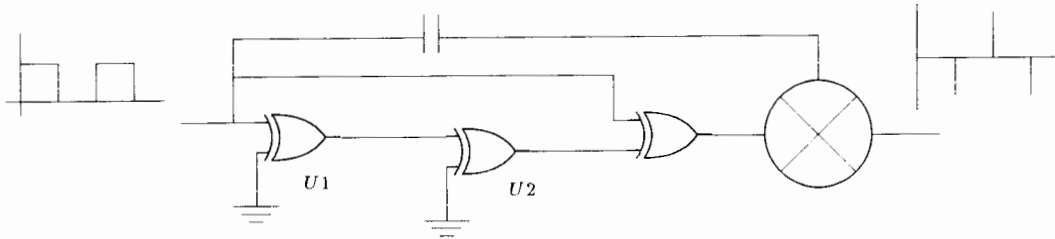


**Figure 5.4** Circuit diagram of an edge detector. A change in the input causes a glitch at the output thereby generating a spike for every edge.

domain so the  $\frac{\sin(x)}{x}$  function narrows in the frequency domain. This performs a ‘filtering’ operation by reducing the higher order harmonics.

Edge detection does not, however, truly differentiate the input square wave as the polarity of the output spikes do not match the polarity of the bit edges, either positive or negative going. This led to the implementation of the signed edge detector by J.G. van de Groenendaal [14]. It is possible to modify the edge detector ( Figure 5.4 ) by adding a few analog components ( Figure 5.5 ) to generate signed pulses for every edge. The output pulses in the signed edge detector are generated using the same method as in the edge detector. However, the pulses are further multiplied by the input signal. This results in them having the correct polarity.

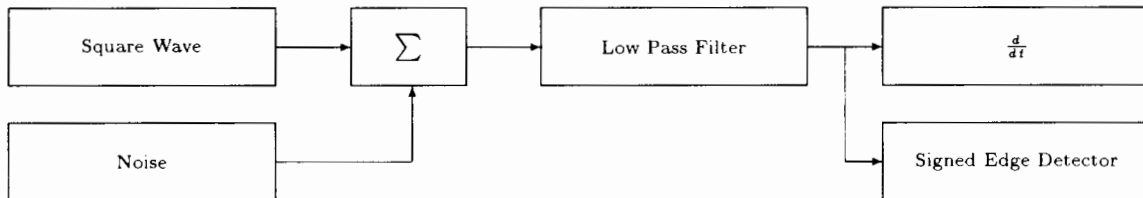
The edge detector retains the low noise characteristics of the edge detector as the input is only sampled when a large change in the input occurs. At all other times during the bit period the output is zero. However, the signed edge detector truly differentiates an input square wave as it generates signed pulses for every bit edge. The circuit therefore implements a low noise, wideband square wave differentiator. The only limits on the bandwidth are the speed of the digital circuitry and the bandwidth of the multiplier.



**Figure 5.5** Circuit diagram of the signed edge detector. Multiplying the spikes at the output of the edge detector results in them having the correct polarity to make this circuit an alternative to differentiation.

### 5.1.1 Simulation of the Signed Edge Detector and Differentiator.

Simulation used to compare the noise performance of the differentiator to the signed edge detector when differentiating a square wave. Figure 5.6 depicts the block diagram used for the simulation. White additive Gaussian noise generated by the TESLA noise model is added to the square wave. Bit estimation is done by resistor-capacitor low pass filtering. A more sophisticated bit estimation method was not used as the simulation was performed to compare of the methods of differentiation in a noisy environment. The filtered output stimulates both the TESLA model of a differentiator, including the lead-lag filter, and the signed edge detector.



**Figure 5.6** Block diagram of the simulation of the TESLA differentiator and the signed edge detector.

Figure 5.7 depicts the output of the low pass filter. Bit edges are discernible while high frequency noise is suppressed but not removed. Figure 5.8 depicts the output of the differentiator. The superimposed noise on the input waveform is amplified thereby degrading the output. The pulses due to the bit edges are not discernible amongst the noise spikes. Figure 5.9 depicts the output of the signed edge detector. Pulses are generated for every bit edge. The only effect of the noise is on the amplitude of the pulses. The high frequency noise is therefore suppressed by this configuration while the output remains wideband.

It should be noted that introducing a Schmitt trigger before the differentiator would improve its noise performance. However, the signed edge detector will still retain the advantages in not suffering from slew rate problems and a variable wide output bandwidth.

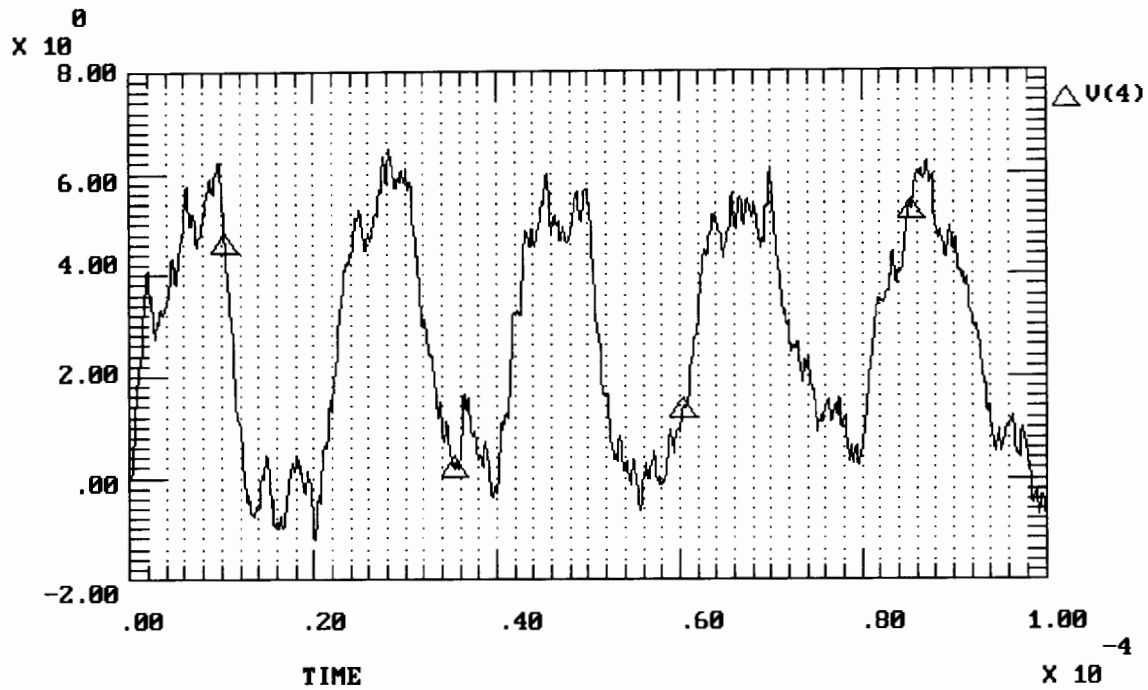


Figure 5.7 Plot of the output of the low pass filter. Bit edges are discernible while high frequency noise is suppressed but not removed.

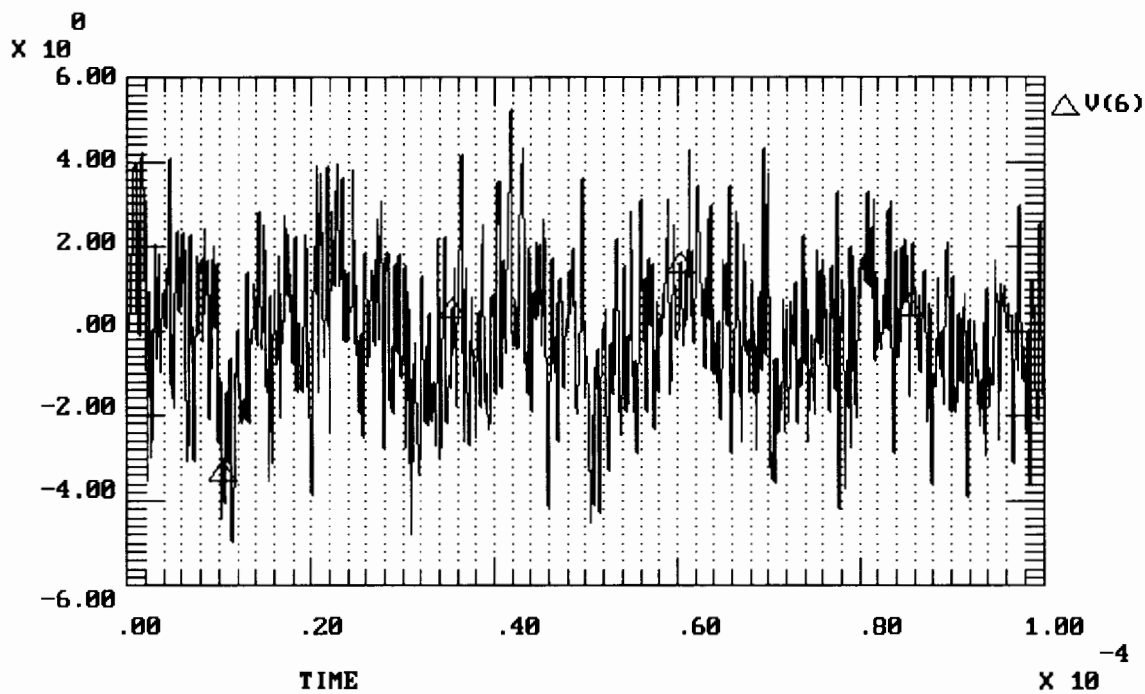


Figure 5.8 Plot of the output of the TESLA differentiator model. High frequency noise is amplified.

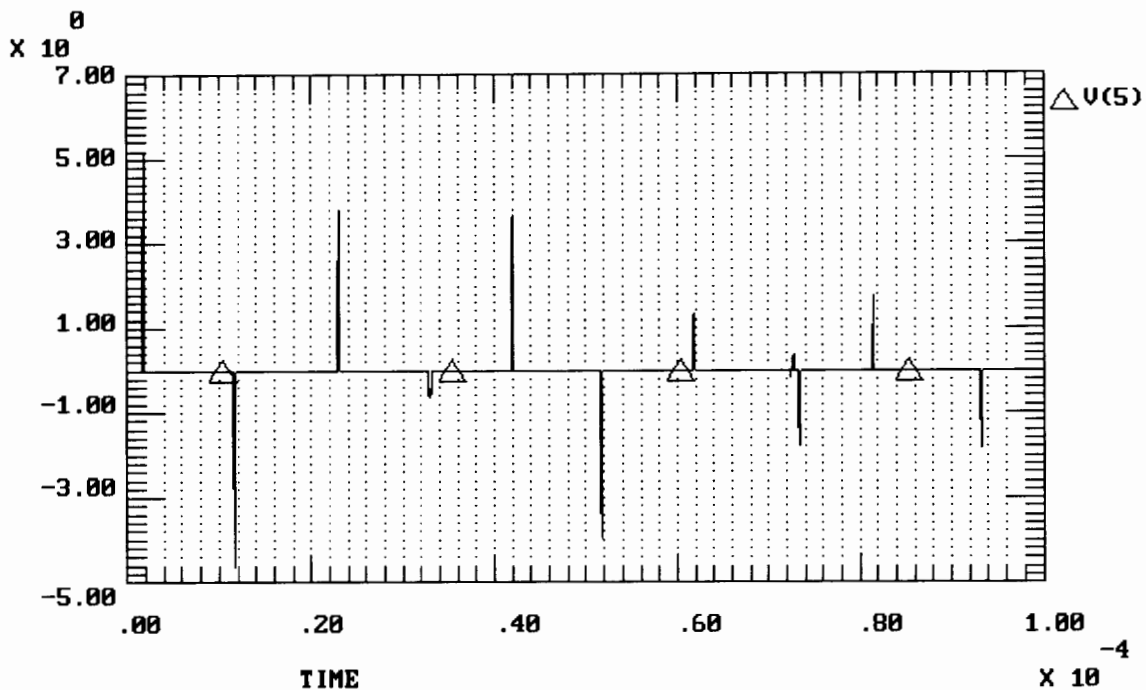


Figure 5.9 Plot of the output of the signed edge detector. The noise only effects the amplitude of the output pulses.

### 5.1.2 Results of the Hardware Implementations of the Differentiator and the Signed Edge Detector.

The differentiator and the signed edge detector were built and tested in a noiseless environment. Under these conditions both the signed edge detector as well as the differentiator perform equally well. The practical findings matched the noiseless simulations.

The signed edge detector was built on double sided pc-board. The input was Schmitt triggered using an HC14 chip so as to sharpen the bit edges and remove ripple. The edge detector was designed using the HC86 XOR-gate chip package. All digital circuitry was decoupled by placing a 100 nF capacitor between the supply rail and ground. The mixer used was a passive double balanced diode bridge mixer from mini-circuits, the SBL-1. This is a wideband mixer with an output bandwidth from D.C. to 500 Megahertz. The mixer did have maximum power limits on its inputs. It was therefore necessary to attenuate the digital input. A resistor network matched to 50  $\Omega$  was therefore used to attenuate the digital input and drive the mixer. However, the digital circuitry could not source enough current to drive a 50  $\Omega$  load. To interface the digital circuitry to the analog components it was therefore necessary to use a line driver. The AD8001, an opamp from Analog Devices, with a 800 MHz bandwidth and a 70 mA output current designed to operate on a  $\pm 5$  volt supply was used for this.

The input to the signed edge detection is shown in figure 5.10. The waveform is not noisy and contains little ripple. The measured results of the signed edge detector are shown in figure 5.11. The output spikes display little overshoot or ringing. The output is clearly wideband due to the narrow width of the spikes.

The active device used in the differentiator was the AD844 opamp. This is a wideband opamp with a slew rate of 2000 Volts per microsecond. The capacitor  $C_1$  was set to 6.8 pF. As the input was to be a noiseless signal the integrator term in the transfer function as set by the capacitor  $C_2$  was not

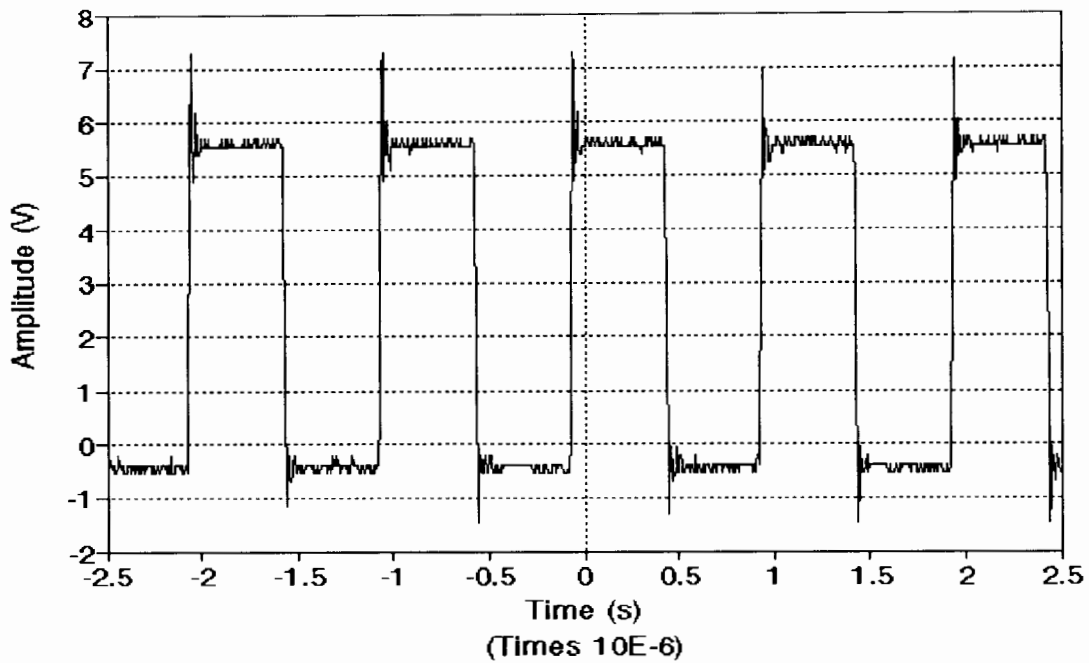


Figure 5.10 Input square wave to the signed edge detector at 1 Megahertz.

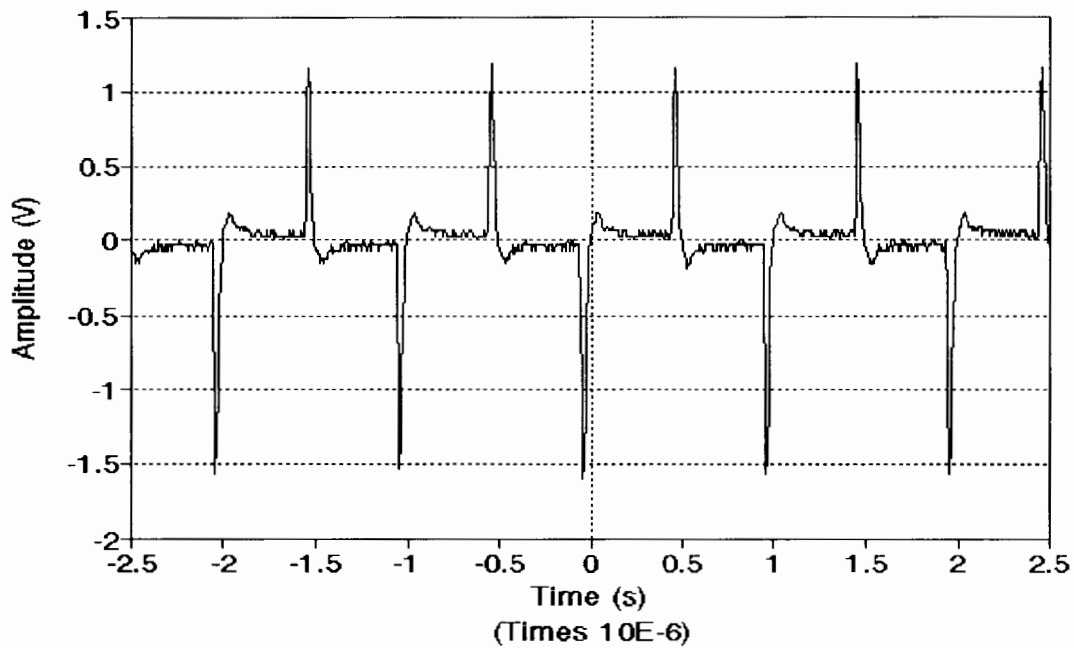


Figure 5.11 Plot of the output of the signed edge detector. The narrow spikes indicate that it is a wideband output.

implemented. This means that the resistors will set the differentiator time constant and the loop gain. The resistors  $R_1$  and  $R_2$  were 10 k $\Omega$  and 50 k $\Omega$  tunable potentiometers. Both resistors were adjusted till the spike were as sharp as possible.

The input to the differentiator is shown in figure 5.12. The waveform has little noise and ripple. The measured results of the differentiator are shown in figure 5.13. Due to the integration component the spikes are not as sharp as those of the signed edge detector and approach zero volts exponentially.

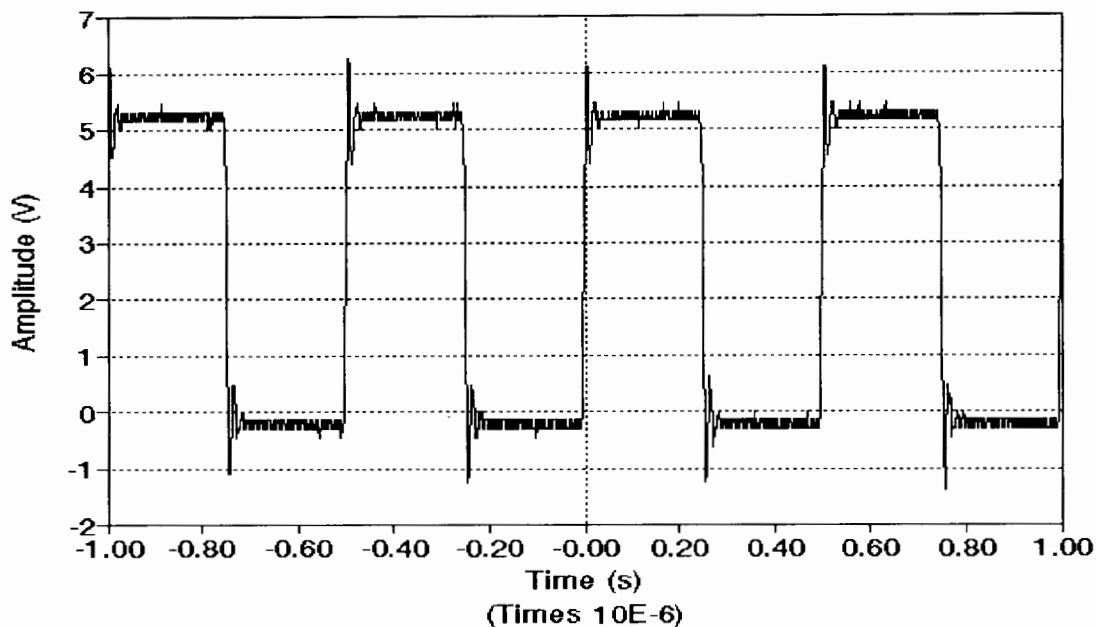


Figure 5.12 Input square wave to the differentiator at 1 Megahertz.

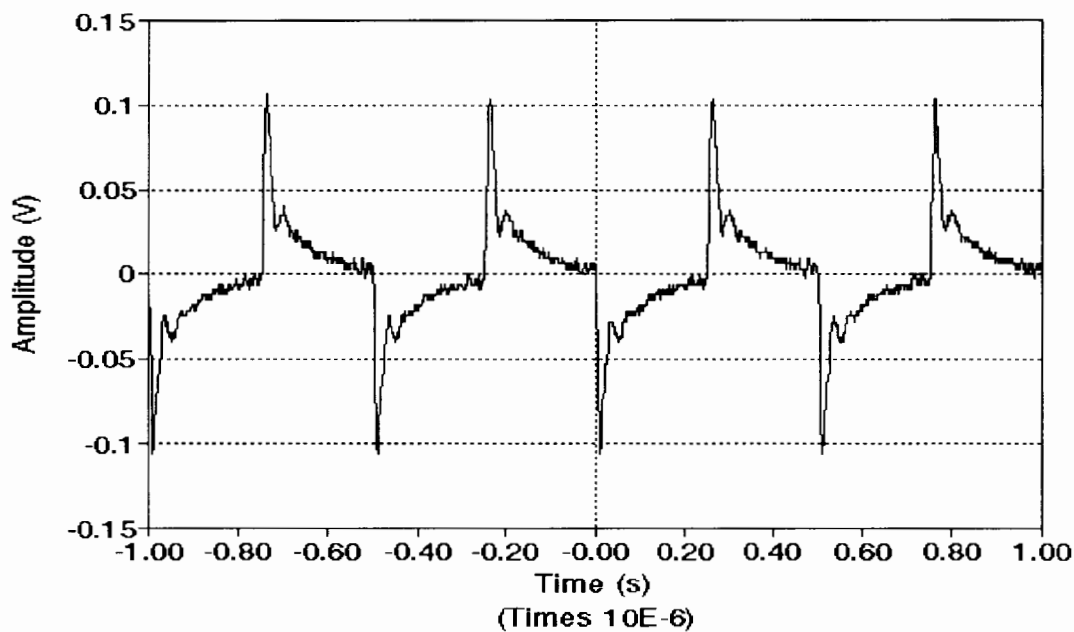


Figure 5.13 Plot of the output of the analog differentiator. The integrator component results in the spike returning to zero exponentially.

It is therefore possible to do signed edge detection in the presence of noise and reduce the noise contributed by the implementation. The advantages of wide bandwidth are not traded off against noise rejection in the signed edge detector.

## 5.2 IMPLEMENTATION OF THE ADJUSTABLE CLOCK.

Two possible implementations of the adjustable clock were considered. They are the phase-locked loop and the injection locked oscillator. Both circuits will acquire the frequency of an input waveform, but differ in both the mechanism of acquisition and the final steady states reached.

### 5.2.1 Discussion of the Phase-locked loop.

The phase-locked loop is a common method of achieving synchronisation. It is a feedback system ( Figure 5.14 ). It consists of a voltage controlled oscillator ( V.C.O. ) driven by a phase detector. The phase detector compares the output of the V.C.O. with the input signal. If there is any phase difference between the two signals the phase detector will adjust the frequency of the V.C.O. to drive the error to zero. As a wide body of theory on the phase-locked loop exists only the fundamental relationship between the acquisition time and noise bandwidth will be highlighted in this text.

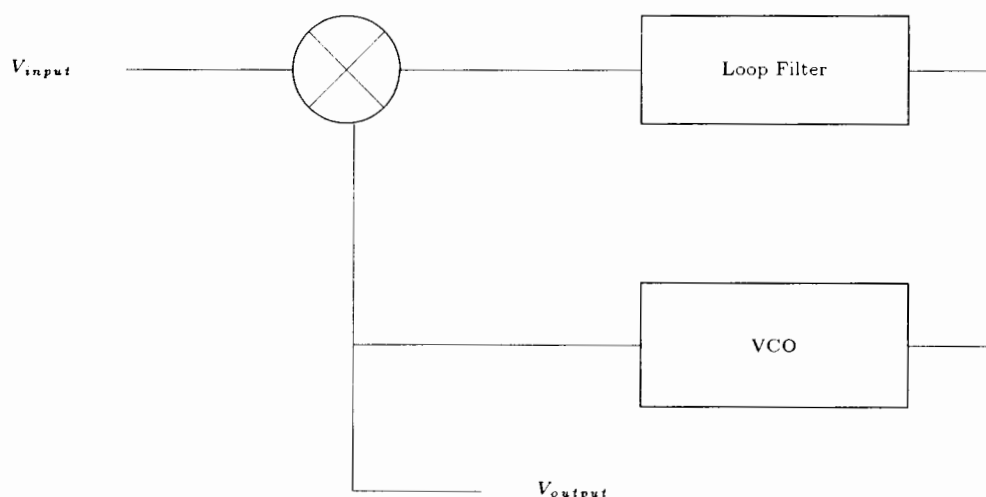


Figure 5.14 Block diagram of the phase-locked loop

As the phase lock loop consists of linear elements it lends itself to analysis in the Laplace domain. Let the input have phase  $\theta_{in}(t)$  and the V.C.O. have phase  $\theta_{out}(t)$ . For the purposes of analysis it is assumed that the phase detector is linearly proportional to the phase difference between its two inputs. One may therefore write

$$V_d(t) = K_d(\theta_{in}(t) - \theta_{out}(t))$$

where  $K_d$  is the phase detector gain factor.

This phase error voltage is filtered by the loop filter. Let the filter have the transfer function  $\mathcal{F}(s)$ . Only those frequencies within the filter's passband will effect the V.C.O. The noise bandwidth is therefore a

fixed constant dependent on the width of the loop filter. The output of the filter is therefore

$$V_c(s) = K_d(\theta_{in}(s) - \theta_{out}(s))\mathcal{F}(s)$$

This voltage then drives the V.C.O.

For the purposes of analysis the output frequency of a V.C.O. is proportional to the input voltage. This may be written as

$$\Delta\omega = K_o V_c$$

where  $\Delta\omega$  is the frequency deviation and  $K_o$  is the V.C.O. gain factor. As frequency is the derivative of phase one may write

$$\theta_{out}(s) = \frac{K_o V_c(s)}{s}$$

The V.C.O. phase is therefore proportional to the integral of the control voltage. It is this property that allows the feedback loop to drive the phase difference between the input and output to zero and therefore achieve frequency synchronisation.

In order to easily consider the loop filters effect on the phase lock loop it is instructive to look at the open loop transfer function.

$$\mathcal{G}(s) = \frac{K_o K_d \mathcal{F}(s)}{s}$$

From this equation it can be seen that the loop filter, characterised by the transfer function  $\mathcal{F}(s)$ , determines the response of the phase-locked loop. If the filter is narrow the loop will have good noise rejection capabilities but the slow poles of the transfer function will lead to a poor acquisition time. The equation further shows that this balancing of noise rejection and acquisition time is a fundamental feature of all phase-locked loops.

## 5.2.2 Discussion of the Injection Locked Oscillator.

Injection locking is a further method of obtaining synchronisation between a local oscillator and an external source. An oscillator will acquire the frequency of an external signal provided that the frequency difference between the impressed signal and the centre frequency of the oscillator is not too large. This method combines a fast frequency locking time with an independent noise and tracking bandwidth. However, the oscillator will introduce a constant phase offset in its output with respect to the input.

### *Derivation of an Injection Locked Oscillator's Locking Mechanism and Final Steady State Error.*

An analysis [1] requires that the output be only a function of instantaneous phase and amplitudes in the circuit. For past conditions to have a negligible affect the circuit must be able to respond quickly to changes. This will be true if the bandwidth of the circuit is wide and if the feedback and bias circuits have no slow poles. Therefore limits on minimum bandwidth and maximum time constants are needed.

In the derivations the following symbols will be used

Angular frequencies

$\omega_o$  = free-running frequency

$\omega_i$  = injected frequency

$\Delta\omega = \omega_o - \omega_i$  = "undisturbed" beat frequency

$\omega$  = instantaneous frequency

$\Delta\omega = \omega - \omega_i$  = instantaneous beat frequency

A further assumption is that the impressed signal is not strong enough to cause lock. This will allow the beat frequencies to exist.

Voltages

$V_c$  = voltage at the collector

$V_t$  = feedback voltage from the tank circuit

$V_i$  = input signal

$V_b$  = voltage on the base

$Q$  = figure of merit of oscillator

If the oscillator is to vary its frequency with little delay, its pass band must be wide in comparison to the 'undisturbed' beat frequency. If this is not the case the tuned circuit will 'memorise' the previous phase and amplitude. One therefore requires

$$\frac{\omega_o}{2Q} \gg \Delta\omega_o$$

As the 'undisturbed' beat frequency is the greatest difference between the free-running and injected frequency this condition implies that the injected signal's frequency should be near the center frequency of the oscillator.

All regenerative oscillators need a feedback loop. This loop will have a form of amplitude control so as to limit the output amplitudes. If this mechanism is too slow then previous variations of amplitude in the beat frequency will affect the locking process. As the shortest beat cycle is the 'undisturbed' beat frequency one requires

$$T \ll \frac{1}{\omega_o}$$

where  $T$  is the time constant of the feedback circuit.

If an oscillator fulfills both these conditions, and many do, then the output amplitude is solely dependent on the gain characteristics of the active devices and the strength of the impressed signal. As the active device operates in a linear part of its curve one should keep the injected signal small.

$$V_i \ll V_t$$

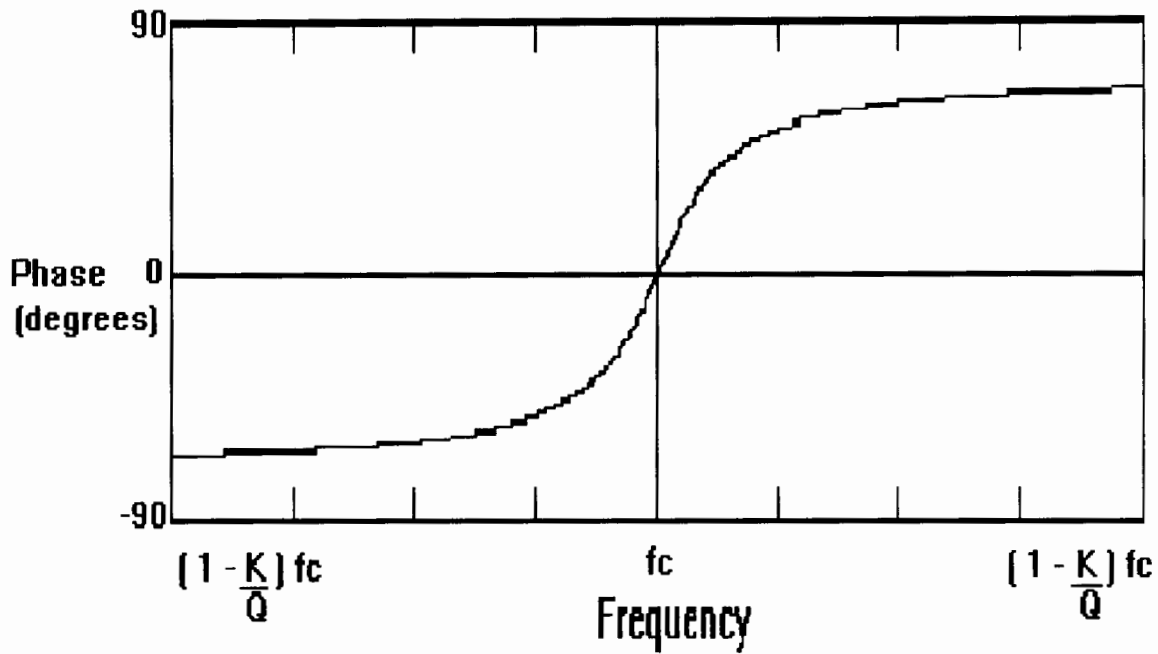
This means that output amplitude variations due to the amplitude of the input will be negligible.

A bandpass filter introduces a degree of phase shift between its input and output ( Figure 5.15 ). As an oscillator consists of a tuned circuit with feedback, any external signal will introduce a phase shift.

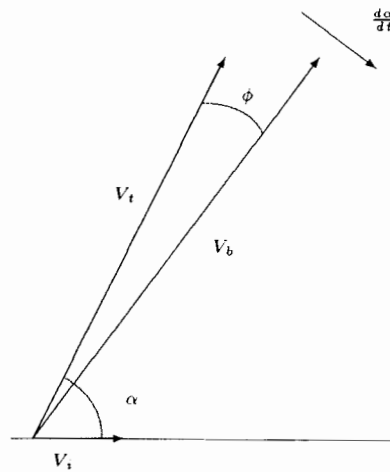
The frequencies present in the injection locked oscillator can be represented in the form of a phasor diagram. The injected signal  $V_i$  sums in the base of the transistor with the feedback from the tank circuit  $V_t$  to form  $V_b$ .

In figure 5.16 the injected signal is used as the reference. The beat frequency is represented by  $d\alpha/dt$ . The impressed signal will now cause  $V_t$  to lag  $V_b$  by a phase angle  $\phi$ . One may therefore write that

$$\phi = -\frac{V_i \sin \alpha}{V_t}$$



**Figure 5.15** Graph of the phase response of a tuned circuit. The phase is only zero at the centre frequency.



**Figure 5.16** Phasor diagram of the waveforms present in an injection locked oscillator.

As the impressed signal is close to the center frequency of the oscillator one can linearise the relationship between frequency and phase for small frequency offsets ( Figure 5.15 ). One may therefore write

$$A = \frac{d\phi}{d\omega}$$

Therefore the phase angle for a frequency close to the free-running frequency is

$$\phi = A(\omega - \omega_o) = A(\Delta\omega - \Delta\omega_o)$$

Now substituting gives

$$\frac{-V_i \sin \alpha}{V_t} = A \left( \frac{d\alpha}{dt} - \Delta\omega_o \right)$$

Letting  $B = V_i/V_t A$  and adding  $\omega_i$  to both sides one obtains

$$\omega = -B \sin \alpha + \omega_o$$

This means that the instantaneous frequency is shifted from the free-running frequency by an amount proportional to the sine of the phase angle between the oscillator and the injected signal. The shift is also proportional to the impressed signal, but inversely proportional to the current in the tank circuit  $V_t$  and the phase verses frequency slope,  $A$ , of the tuned circuit.

In a single tuned circuit with small deviations in frequency from the free-running one may approximate phase by

$$\phi = 2Q \frac{\omega - \omega_o}{\omega_o}$$

This gives the phase verses frequency slope as

$$A = \frac{2Q}{\omega_o}$$

Substituting this into the equation one derives

$$\frac{d\alpha}{dt} = \frac{-V_i \omega_o \sin \alpha}{2V_t Q} + \Delta\omega_o$$

In a steady state there is no change in phase error. This means that  $d\alpha/dt = 0$  and one obtains

$$\sin \alpha = 2Q \frac{V_t \Delta\omega_o}{V_i \omega_o}$$

This gives the stationary phase angle between the oscillator and impressed signal. As the absolute value of  $\sin \alpha$  must be less than or equal to one, this equation also gives the synchronisation condition.

$$\frac{V_i}{V_t} = 2Q \left| \frac{\Delta\omega_o}{\omega_o} \right|$$

This synchronisation condition is of practical importance and can be reworked. As the voltage at the collector, which is normally the output, is of more import than the current in the base this should be included in the synchronisation condition. As the tuned circuit is the collector load one may write

$$V_c = V_b g_m Q \sqrt{\frac{L}{C}}$$

Where  $g_m$  is the transconductance of the transistor. Which allows the synchronisation condition to be written as

$$\frac{V_c}{V_i} = \left| \frac{\omega_o}{2\Delta\omega_o} \right| h_{fe} \sqrt{\frac{L}{C}}$$

Outside the locking range defined by the above condition one notices the beat frequency phenomenon. To simplify the mathematics one can make the following substitution. Let

$$K = \frac{\Delta\omega_o}{B} = 2Q \frac{V_t \Delta\omega_o}{V_i \omega_o}$$

One can now write

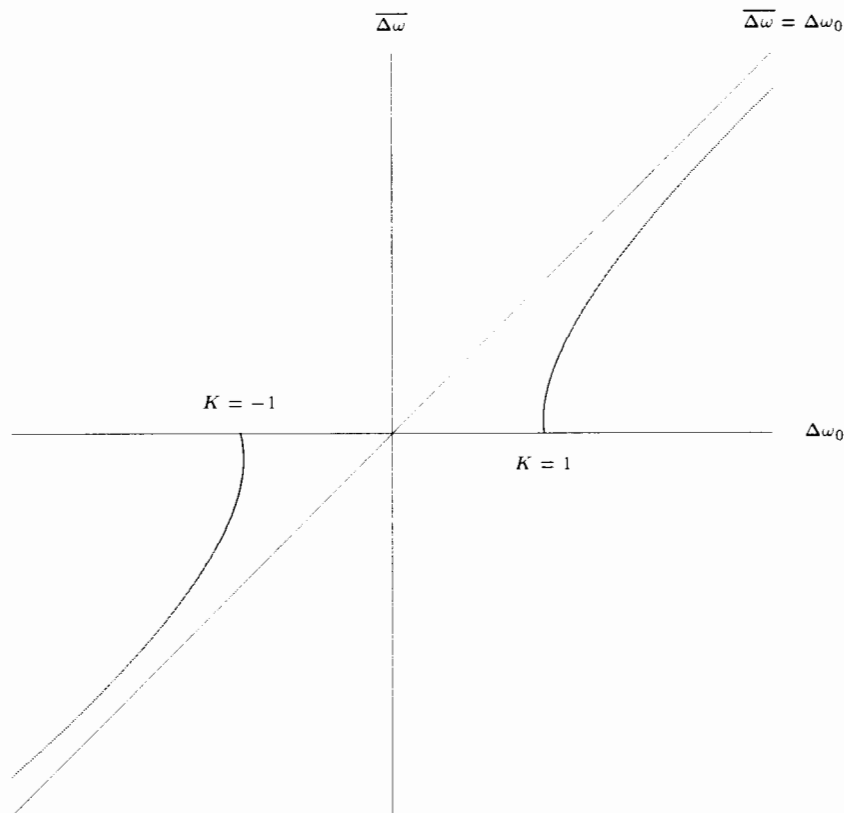
$$\frac{d\alpha}{dt} = -B(\sin \alpha - K)$$

Integration of this equation will give phase as a function of time. Integration gives

$$\alpha = 2 \arctan\left(\frac{1}{K} + \frac{\sqrt{K^2 - 1}}{K} \tan \frac{B(t - t_0)\sqrt{K^2 - 1}}{2}\right)$$

where  $t_0$  is an integration constant.

This equation has two states. If the synchronisation conditions are not met, then  $|K| > 1$  and  $\sqrt{K^2 - 1}$  is therefore real. Under these conditions beat frequencies will be present. Let  $\overline{\Delta\omega}$  be the average beat frequency. This is different from the ‘undisturbed’ beat frequency as the instantaneous frequency is being continuously modulated by the impressed signal.( Figure 5.17 ).



**Figure 5.17** Graph depicting the reduction of beat frequencies due to locking.

As the impressed signal moves further from the center frequency one can see that its effect on the oscillator decreases. The other state is  $|K| < 1$  in which case lock will occur.

Firstly the simple case where  $\Delta\omega_o = 0$  is considered. This will allow the synchronisation requirements to be met under any conditions. This reduces the differential equation to

$$\frac{d\alpha}{dt} = -B \sin \alpha$$

As the initial lag  $\alpha_i$  is reduced one may apply the linear approximation of sine to give

$$\frac{d\alpha}{dt} = -B\alpha$$

The solution to this equation is

$$\alpha = \alpha_i \exp(-Bt)$$

This means that, in the simple case, the oscillator phase approaches exponentially that of the impressed signal. The speed of this approach will depend on the strength of the injected signal and the bandwidth of the tuned circuit.

If  $\Delta\omega_o$  is not equal to zero one must use the accurate equation of phase as a function of time. The synchronisation requirements are fulfilled so  $|K| < 1$ . This means that  $\sqrt{K^2 - 1}$  must be replaced by  $j\sqrt{1 - K^2}$ . The equation reduces to

$$\tan \frac{\alpha}{2} = \frac{1}{K} - \frac{\sqrt{1 - K^2}}{K} \tanh \frac{B(t - t_o)\sqrt{1 - K^2}}{2}$$

The tanh function maintains an exponential characteristic. The integration constant allows the equation to be fitted to any initial conditions.

As time increases the tanh function tends to unity. The steady state is therefore given by

$$\tan \frac{\alpha}{2} = \frac{1 - \sqrt{1 - K^2}}{K}$$

The synchronisation condition makes  $K$  equal to  $\sin \alpha$  in the steady state. The above equation can be therefore be proved true using trigonometric identities.

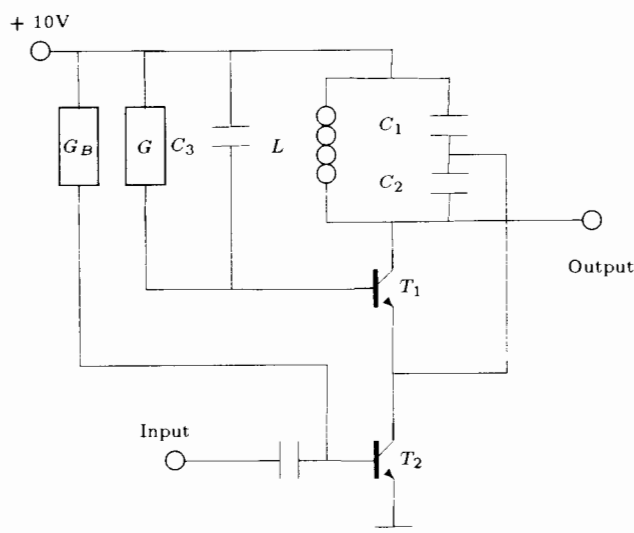
The analysis therefore shows that an oscillator is able to acquire the frequency of an impressed signal and can therefore be used as a synchronisation circuit. It oscillator achieves this by modulating its output phase to cancel any beat frequencies that might exist. In so doing a degree of steady state phase error is generated between the input and output of the injection locked oscillator.

### *The Synchronous Oscillator.*

The synchronous oscillator was originally developed by Uzunoglu and White [12] ( Figure 5.18 ). It can be viewed as a Colpitts oscillator with a current tail. It differs from other injection locked oscillators in that it has a near constant magnitude over its tracking range ( Figure 5.22 ) unlike other circuits which display the characteristic tuned circuit magnitude response.

Hickman [8] highlighted design flaws in Uzunoglu and White's original circuit. Hickman's investigation involved improving the bias conditions of the transistors in the network while maintaining the gain-synchronisation curve as described by Uzunoglu and White. Hickman designed a split supply network that maintains the characteristics of the synchronous oscillator ( Figure 5.19 ).

The characterisation of the synchronous oscillator reveals some interesting behavior ( Figures 5.20 through 5.23 ). As the input level is lowered the locking range, the flat part of the magnitude response, narrows therefore increasing the noise rejection capabilities. The tracking bandwidth is narrowed by this characteristic under conditions of low input. This improves the synchronous oscillator's ability to resolve input frequencies at low input levels as adjacent frequencies in the frequency domain will be rejected. The synchronous oscillator's noise bandwidth however is determined by the magnitude response of its output. As the output consists of a sine wave generated by the oscillator's tank circuit the noise bandwidth is a function of the lock conditions. The synchronous oscillator's noise and tracking bandwidth are therefore largely independent of each other.



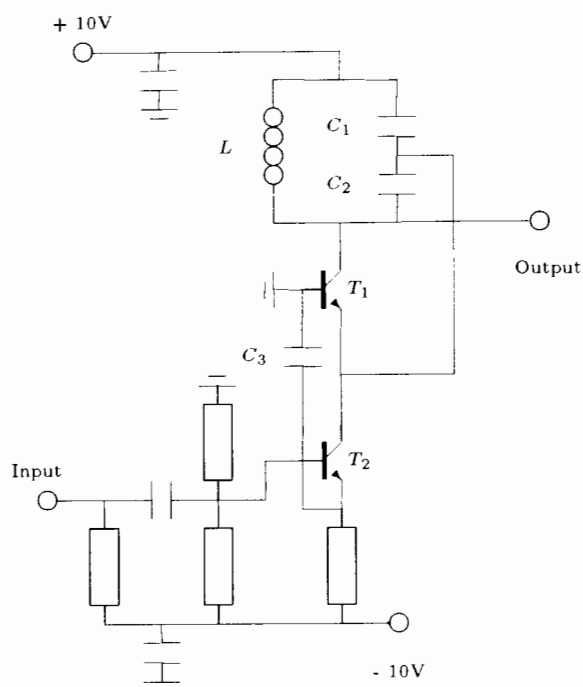
**Figure 5.18** Circuit diagram of the synchronous oscillator as designed by Uzunoglu and White. It can be viewed as a Colpitts oscillator with a current tail.

The near constant output across the locking range as mentioned by Uzunoglu and White is not wholly independent of the input level. One can detect a slope across the tracking range which varies with the input level.

The centre frequency of the synchronous oscillator can be adjusted by varying the D.C. bias voltage on the base of the transistor  $T_1$  in figure 5.18. Varying the voltage on the base alters the width of the depletion layer in the transistor. This in turn effects the value of the depletion capacitance. As this capacitance forms part of the tank circuit, varying this capacitance alters the centre frequency of the oscillator. This effect has two applications in the synchronous oscillator.

Firstly, the storage time of the synchronous oscillator can be designed independently of the tracking range. Storage time is the time taken by the synchronous oscillator to return to its free running frequency after the input has been removed. The length of this time is determined by the capacitor  $C_3$  in figure 5.18. As this capacitor stores charge it acts as a 'memory' element in the loop. When the input is removed the capacitor  $C_3$  maintains the voltage on the base of transistor  $T_1$  therefore maintaining the output frequency. As the charge drains from the capacitor  $C_3$  the oscillator's output frequency slowly returns to its free running frequency.

The effect of the feedback capacitor on storage time was investigated. An edge detected maximal-length sequence was used to supply a series of unipolar spikes to synchronise a synchronous oscillator as depicted in figure 5.18. However, the pseudorandom sequence would also supply periods when no edge would be detected and therefore no input to the synchronous oscillator would be generated. Figure 5.24 depicts the synchronous oscillator output and pseudorandom sequence used as input with a 100 nF feedback capacitor. During the long run without edges the synchronous oscillator output amplitude decreases. However, the frequency is maintained. Only when the output disappears will the frequency start to return to the free running frequency. Figure 5.25 depicts the synchronous oscillator output and pseudorandom sequence used as input with the sequence length shortened and the feedback capacitor decreased to 47 nF. The synchronous oscillator was unable to maintain the output frequency using a longer maximal length sequence. This is due to the shortening of the storage time due to the decrease in the value of the feedback capacitor  $C_3$ . Figure 5.26 depicts the synchronous oscillator output and pseudorandom



**Figure 5.19** Circuit diagram of the synchronous oscillator as designed by Hickman. The bias conditions of the transistors are improved by the split supply design.

sequence used as input with the feedback capacitor decreased to 107 nF. The output amplitude drops off faster than it did with larger feedback capacitors. The feedback capacitor  $C_3$  is therefore responsible for the storage time of the synchronous oscillator.

Secondly, use of the depletion capacitor can be made in removing the steady state phase error introduced by the synchronous oscillator. As the phase difference between the input and output of a tuned circuit is zero at the centre frequency of the tuned circuit altering the free running frequency of the synchronous oscillator to that of the input could drive the phase error to zero. Uzunoglu and White [11] further refined the synchronous oscillator to produce the coherent phase-locked synchronous oscillator using this phenomenon ( Figure 5.27 ). In this network the input phase is compared to the output phase. This phase error voltage is fed to the base of the transistor  $T_1$  therefore modulating the depletion capacitance in the transistor. This in turn varies the free running frequency of the synchronous oscillator and reduces the steady state phase error. However, phase comparison requires a degree of integration with respect to time. The time required therefore to achieve phase coherency is of the order of that of a phase-locked loop. However, the time taken to achieve frequency lock is unchanged from the normal synchronous oscillator.

The modulation of output phase through the depletion capacitance of the tank circuit transistor  $T_1$  in figure 5.18 was investigated by adding an extra power supply. This power supply was connected to the base of transistor  $T_1$  through a 10 k $\Omega$  resistor. Figure 5.28 depicts the Lissajous figure when lock has been achieved by the synchronous oscillator. The figure is a skewed oval showing that a phase difference, but no frequency difference, exists between input and output. Applying a voltage to the base of transistor  $T_1$  however allowed one to correct this phase difference ( Figure 5.29 ). The Lissajous figure is now a

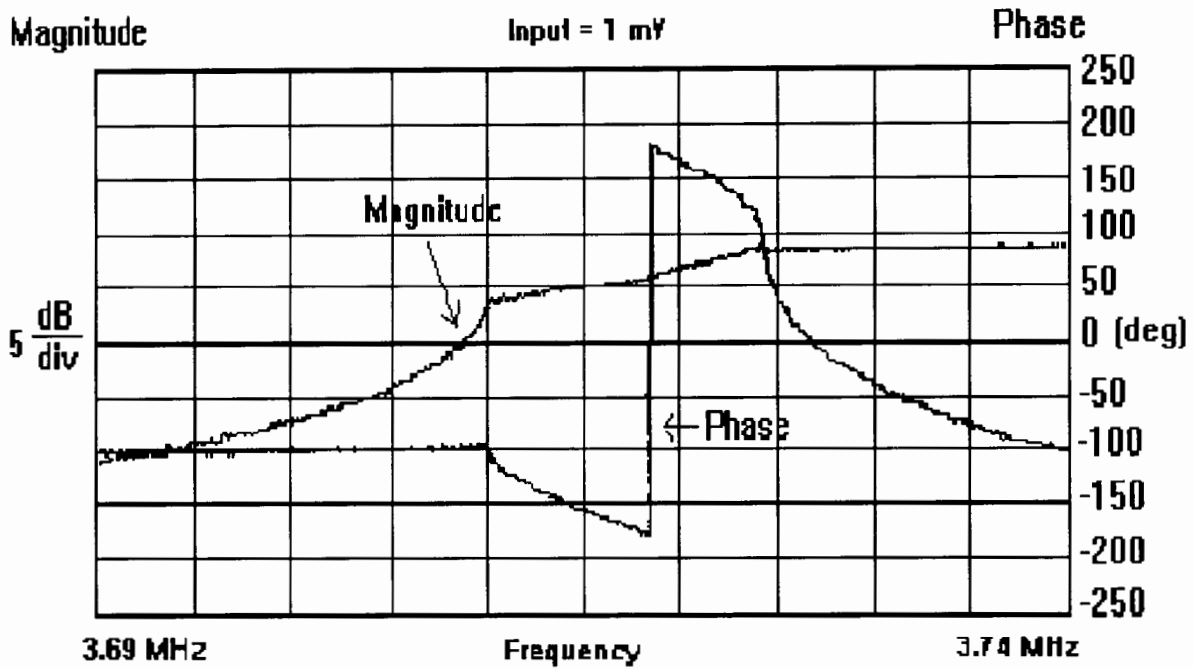


Figure 5.20 Plot of the magnitude and phase response of the synchronous oscillator at 1mV input.

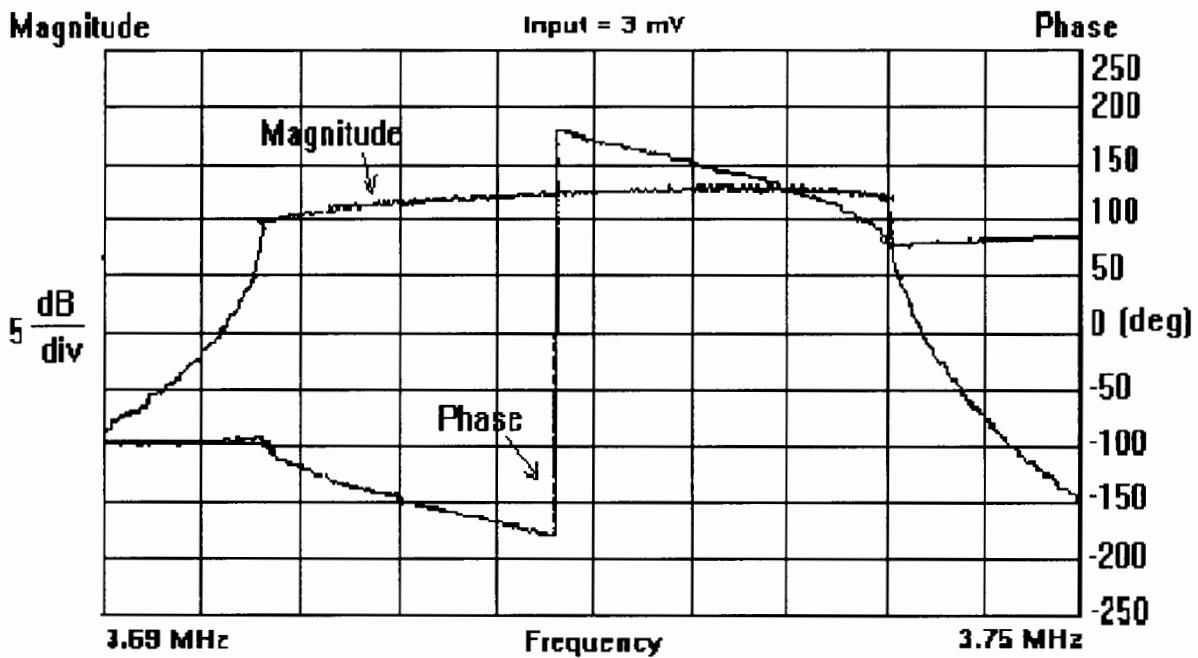


Figure 5.21 Plot of the magnitude and phase response of the synchronous oscillator at 3mV input.

circle indicating that the input and output are in phase. This was achieved by applying 25 Volts to the base [3].

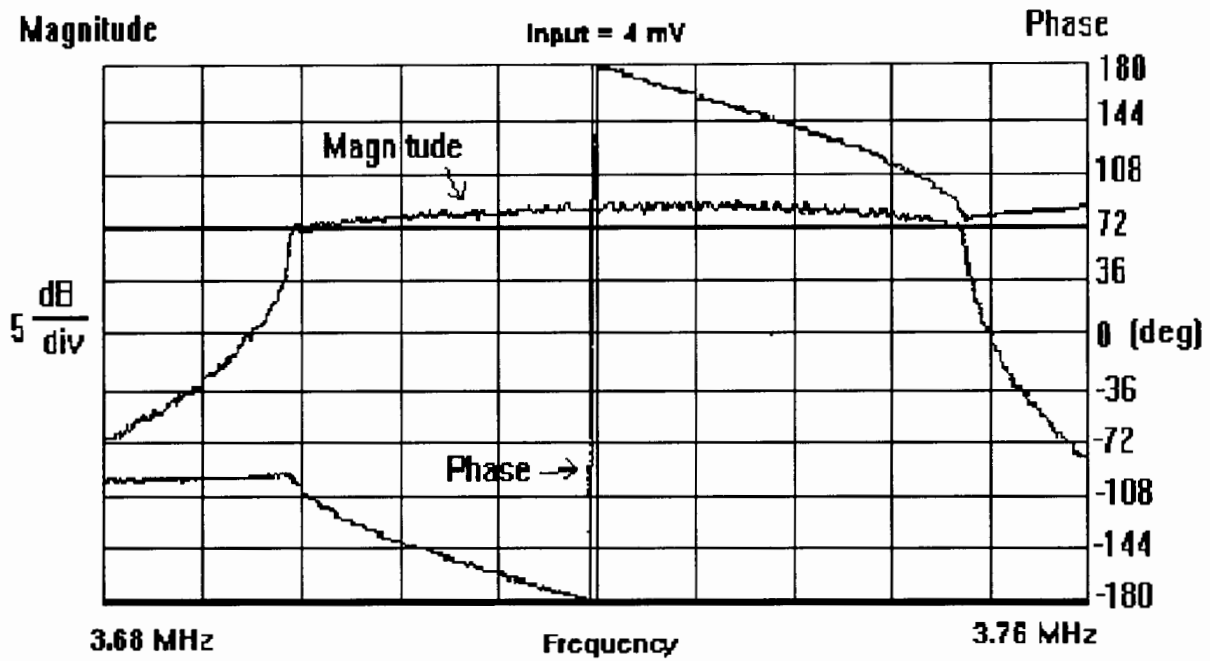


Figure 5.22 Plot of the magnitude and phase response of the synchronous oscillator at 4mV input.

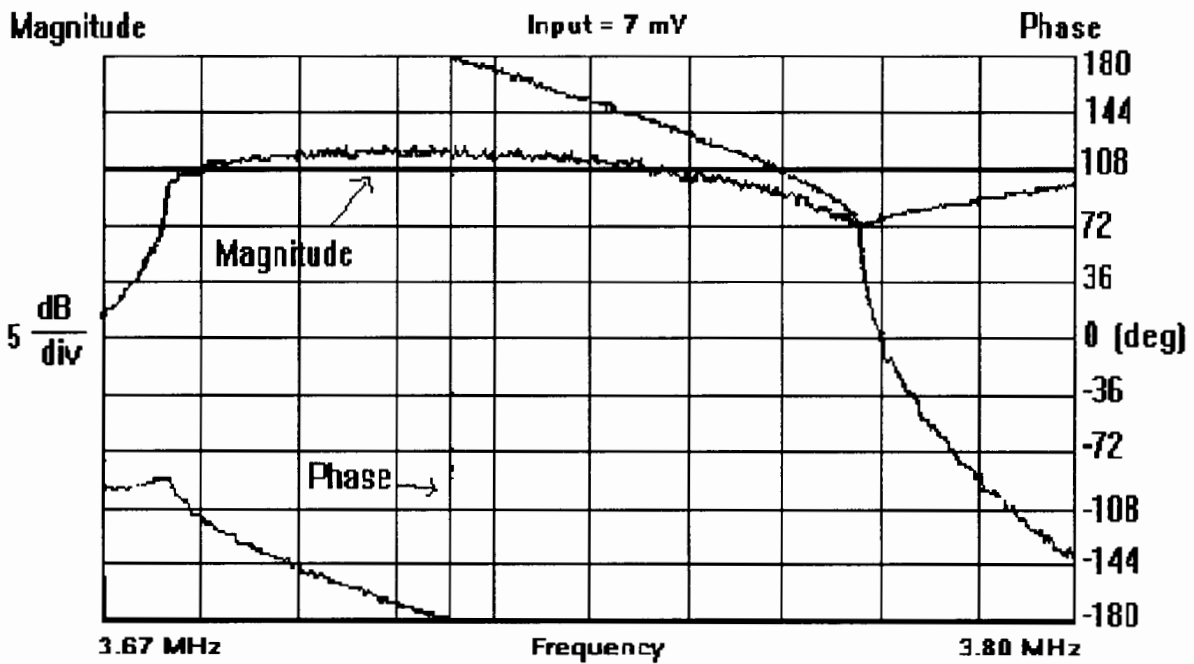
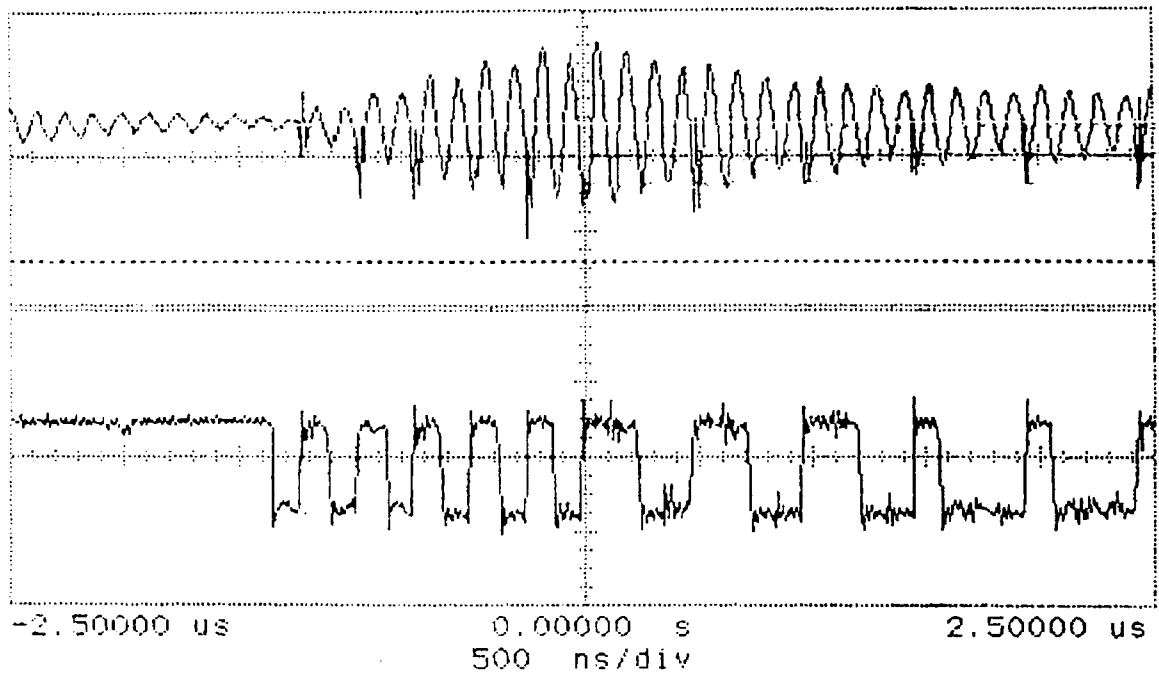


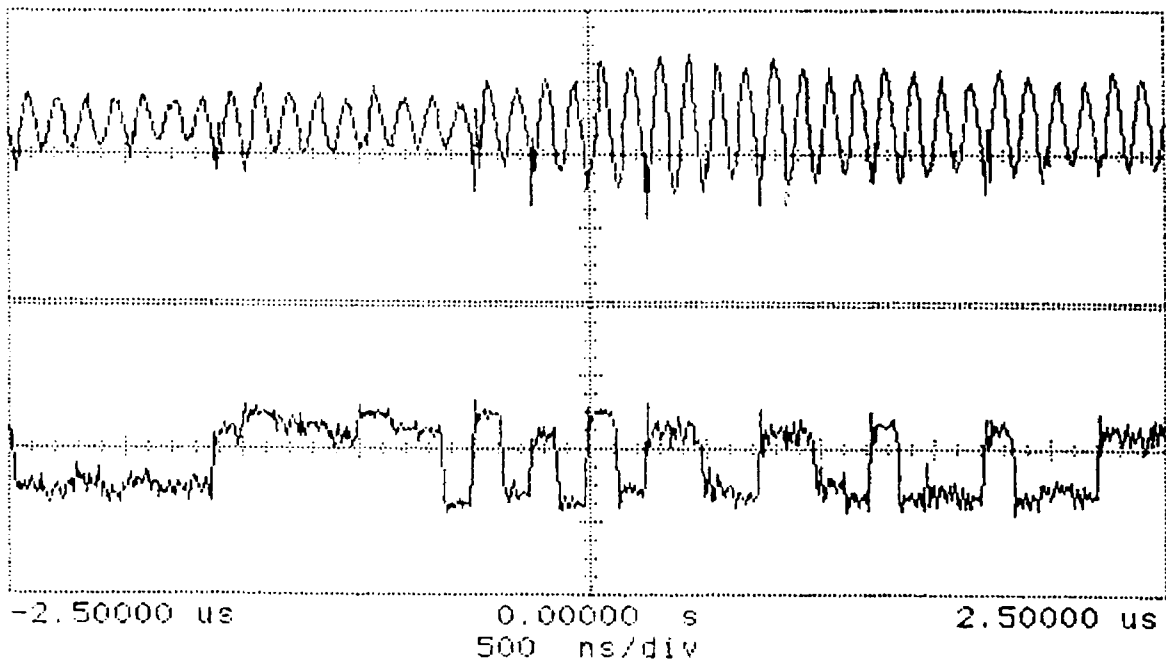
Figure 5.23 Plot of the magnitude and phase response of the synchronous oscillator at 7mV input.

*The ‘Current Hump’ in Injection Locked Oscillators.*

The injection locked oscillator exhibits an interesting phenomenon called the ‘current hump’ [4] [2]. At the centre or resonant frequency of the injection locked oscillator, the power supply current is a maximum. It decreases as the injected frequency moves away from the resonant frequency. Braun used

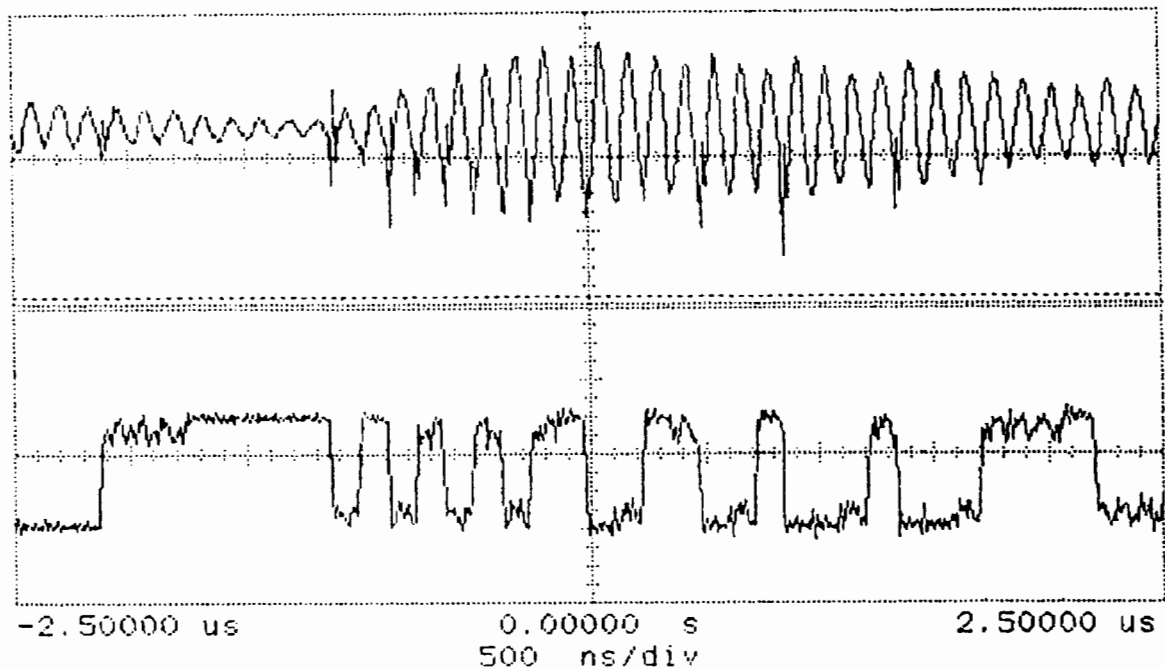


**Figure 5.24** Plot displaying the effects of the feedback capacitor  $C_3$  on storage time. The sequence length was  $2^{12} - 1$  and the feedback capacitor was 100 nF.



**Figure 5.25** Plot displaying the effects of the feedback capacitor  $C_3$  on storage time. The sequence length was  $2^8 - 1$  and the feedback capacitor was 47 nF.

this phenomenon to produce a phase coherent injection locked oscillator at microwave frequencies. The ‘current hump’ was therefore investigated in order to determine if it existed, and could therefore be applied in order to cancel the phase error, in other injection locked oscillators



**Figure 5.26** Plot displaying the effects of the feedback capacitor  $C_3$  on storage time. The sequence length was  $2^8 - 1$  and the feedback capacitor was 10 nF.

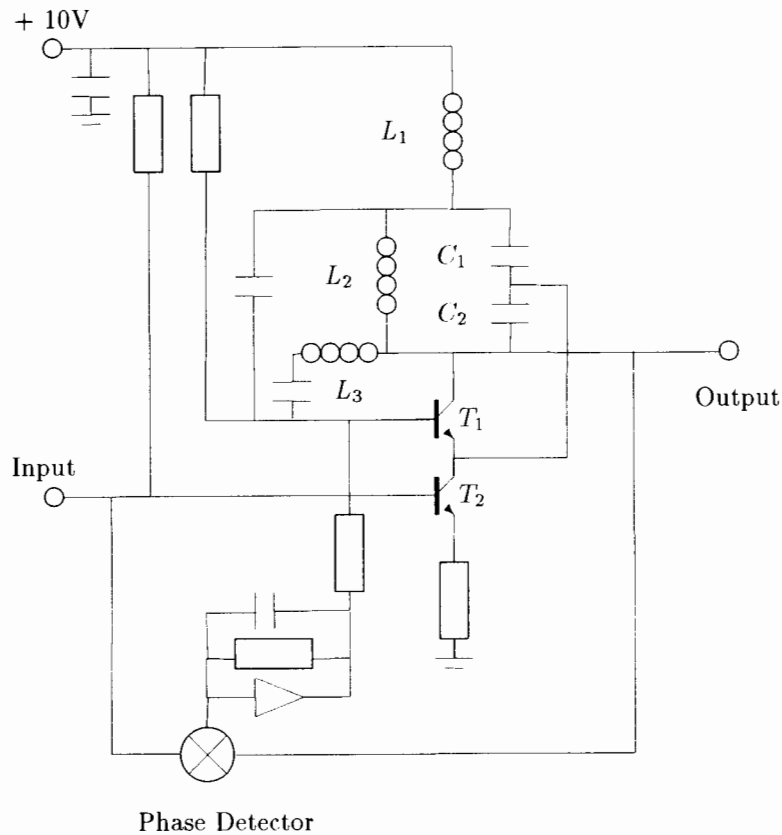
Investigation of the ‘current hump’ phenomenon, involved three different oscillators, namely a simple Colpitts oscillator, an optimised Colpitts oscillator and the synchronous oscillator as described by Hickman. The optimised Colpitts oscillator consists of a standard colpitts oscillator but displays a similar ‘flat’ tracking range as the synchronous oscillator. This is accomplished by tuning the resistor between the emitter and ground. Figures 5.30, 5.31 and 5.22 show the magnitude and phase frequency responses of the three circuits respectively.

The plots were obtained on a network analyser by sweeping the frequency across the range of interest and measuring the magnitude and phase of the output. The simple Colpitts oscillator displays the classic tuned circuit response ( Figure 5.30 ), while the optimised Colpitts oscillator and the synchronous oscillator, due to their bias conditions, display a near constant amplitude response over the tracking range ( Figures 5.31 and 5.22 ). All the oscillators were tested at various injection levels while sweeping the injection signal’s frequency across the tracking bandwidth. The ‘current hump’ was obtained in all cases.

Although the output amplitude of a tuned circuit ( Figure 5.30 ) is a maximum at the resonant frequency, and decreases as the injected frequency is swept away from this maximum the ‘current hump’ is not related to this phenomenon as the optimised Colpitts oscillator and the synchronous oscillator have a near constant output amplitude across the tracking range. ( Figures 5.31 and 5.22 ). However, all the circuits display a steady-state phase error between the injected and the output waveform. This phase error is zero at the resonant frequency, while tending to  $|90|$  degrees at either side of the tracking range. Examination of the power supply current versus phase plots display a raised cosine relationship ( Figure 5.32 ). It is therefore possible that the ‘current hump’ is a function of the steady-state phase error.

The following equation was fitted to the measured data, making use of only the maximum and minimum values of the supply current in the locking range.

$$I = I_{|90 \text{ deg}|} + I_{\Delta} \cos(\phi) \quad (5.1)$$



**Figure 5.27** Circuit diagram of the coherent phase-locked synchronous oscillator as described by Uzunoglu and White.

where  $I_{\Delta} = I_o - I_{|90 \text{ deg}|}$ ,  $I_o$  is the current at the centre frequency (i.e. zero phase) and  $I_{|90 \text{ deg}|}$  is the current at the edge of the locking range. Figures 5.33 and 5.34 show the measured and fitted curves across the tracking range for two of the injection locked oscillators. The close fit suggests that equation 5.1 is a good approximation.

Equation 5.1 suggests a definite link between the current drawn from the power supply and the phase shift introduced by an injection locked oscillator. Adler's derivation of the locking mechanism in oscillators clearly shows that all injection locked oscillators introduce a phase offset between the input and output signals in order to achieve lock. It is therefore proposed that the detection of a 'current hump' in a network under injection conditions is a possible condition for identifying an injection locked oscillator as described by Adler.

It is claimed by Uzunoglu [7] that the synchronous oscillator is not an injection locked oscillator due to the oscillator's magnitude response. ( Figure 5.22 ). The synchronous oscillator displays a near constant output amplitude over the tracking range, as opposed to the classic bandpass response of a tuned circuit. Although the original design by Uzunoglu and White failed to produce the 'current hump' the design proposed by Hickman displayed this phenomenon as described by equation 5.1. This suggests that the synchronous oscillator could be an injection locked oscillator as described by Adler as the 'current hump' is a possible condition for identifying an injection locked oscillator.

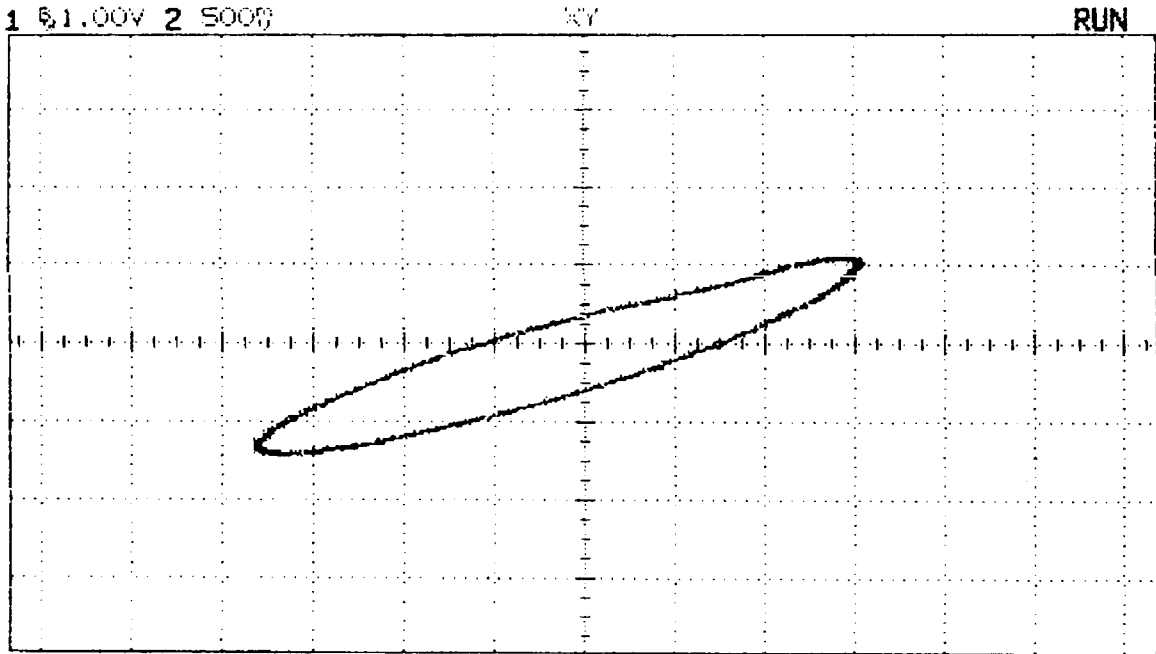


Figure 5.28 Plot of the Lissajous figure when the input and output are not in phase. This is indicated by the skewed oval.

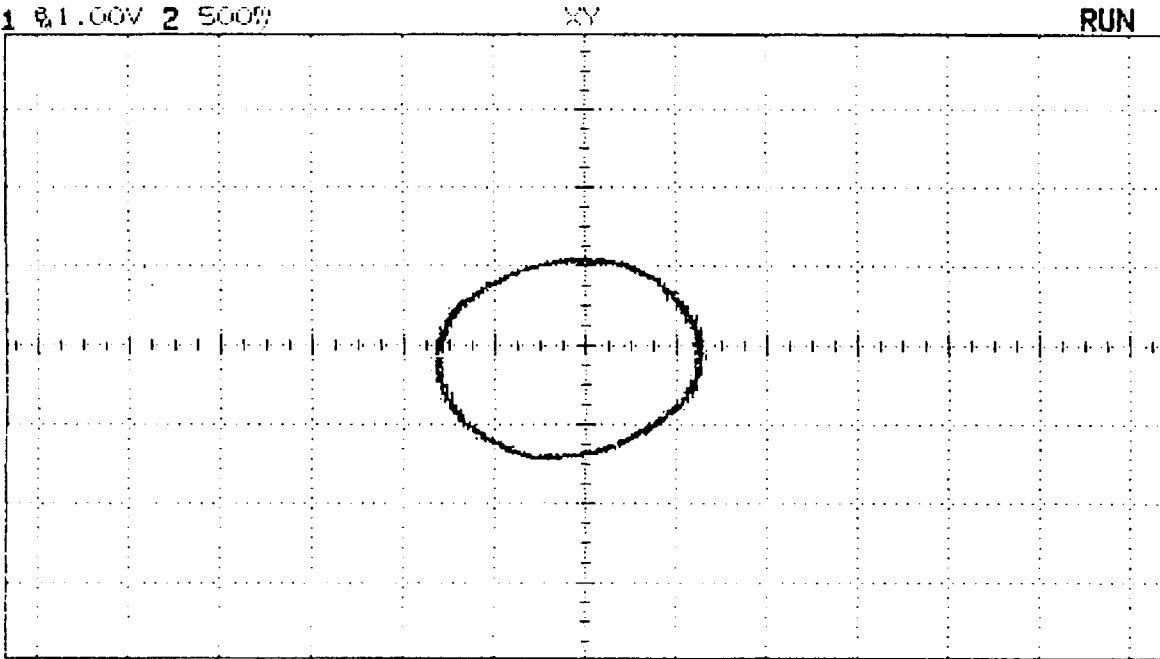


Figure 5.29 Plot of the Lissajous figure when the input and output are in phase. This is indicated by the circle.

The clear link between the power supply current and the steady state phase error indicates that the 'current hump' may well be able to be used in generating a phase coherent injection locked oscillator using the methods as detailed by Braun.

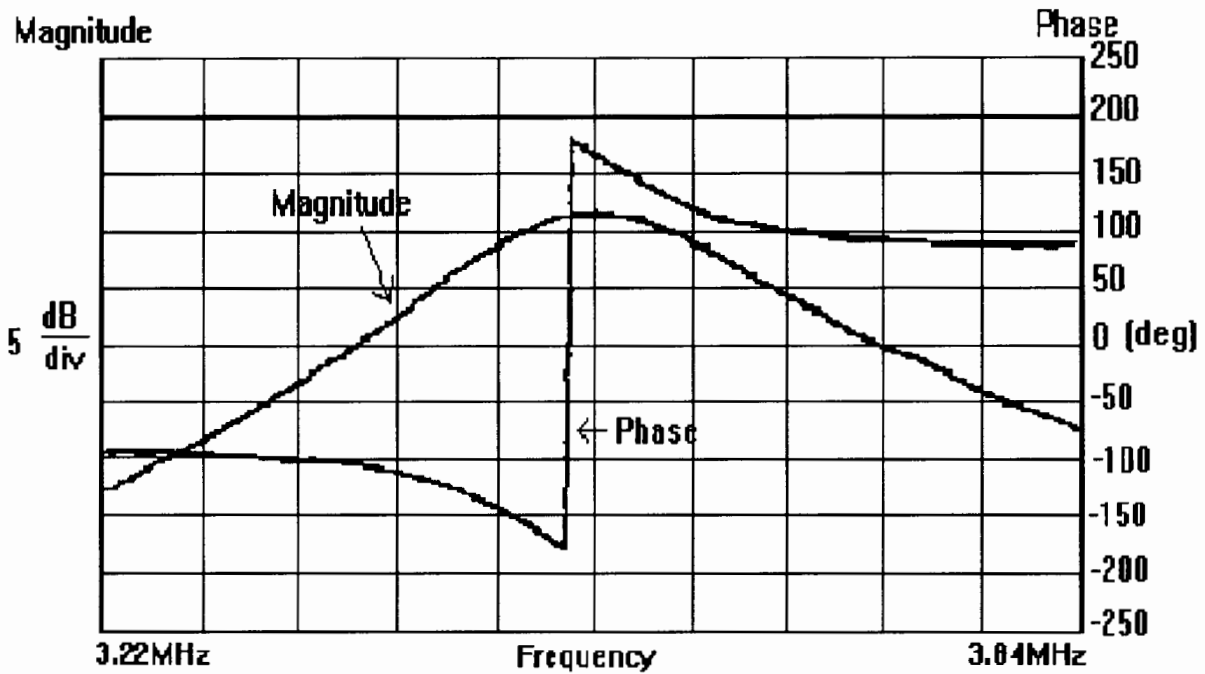


Figure 5.30 Plot of the magnitude and phase response of the Colpitts oscillator.

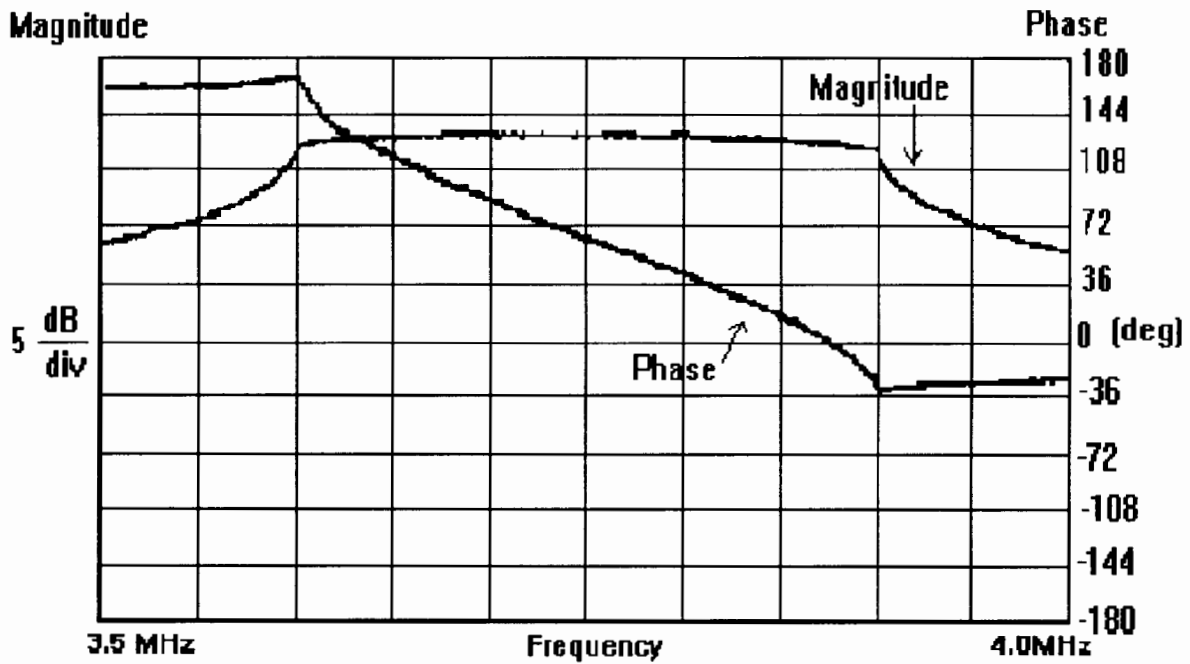


Figure 5.31 Plot of the magnitude and phase response of the optimised Colpitts oscillator.

### 5.2.3 Comparison of the Injection locked Oscillator and the Phase-locked Loop

The use of any synchronisation network is dependent on the application. By choosing the correct circuit one can improve performance and more importantly reduce costs. Possible criteria for comparison of the synchronisation networks include:

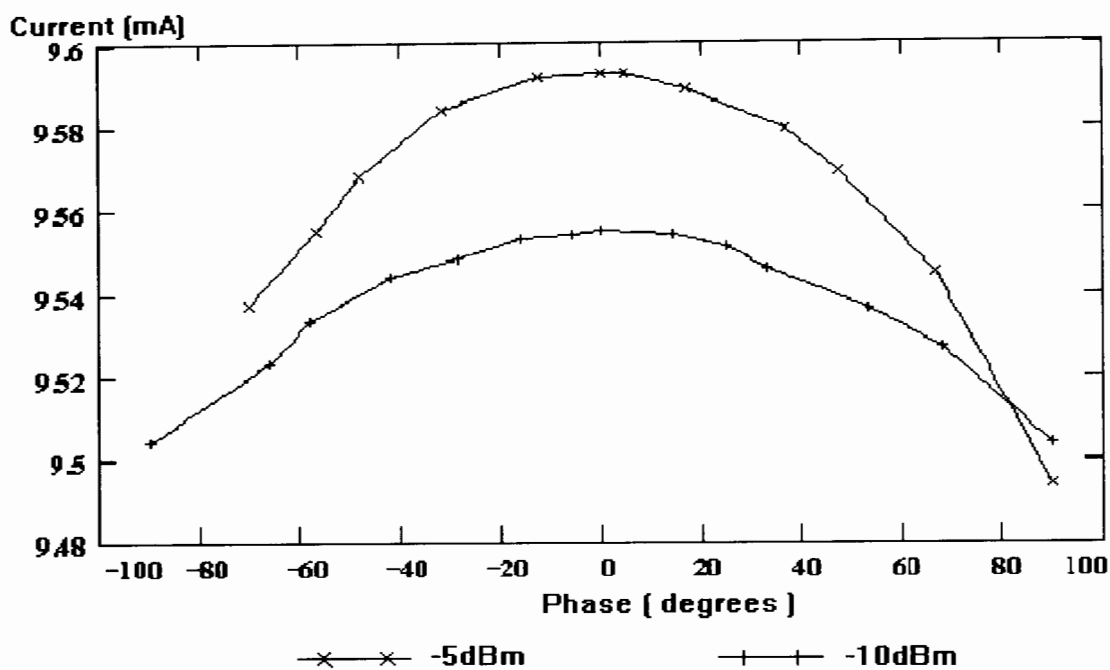


Figure 5.32 Graph of the 'current hump' at different injection levels.

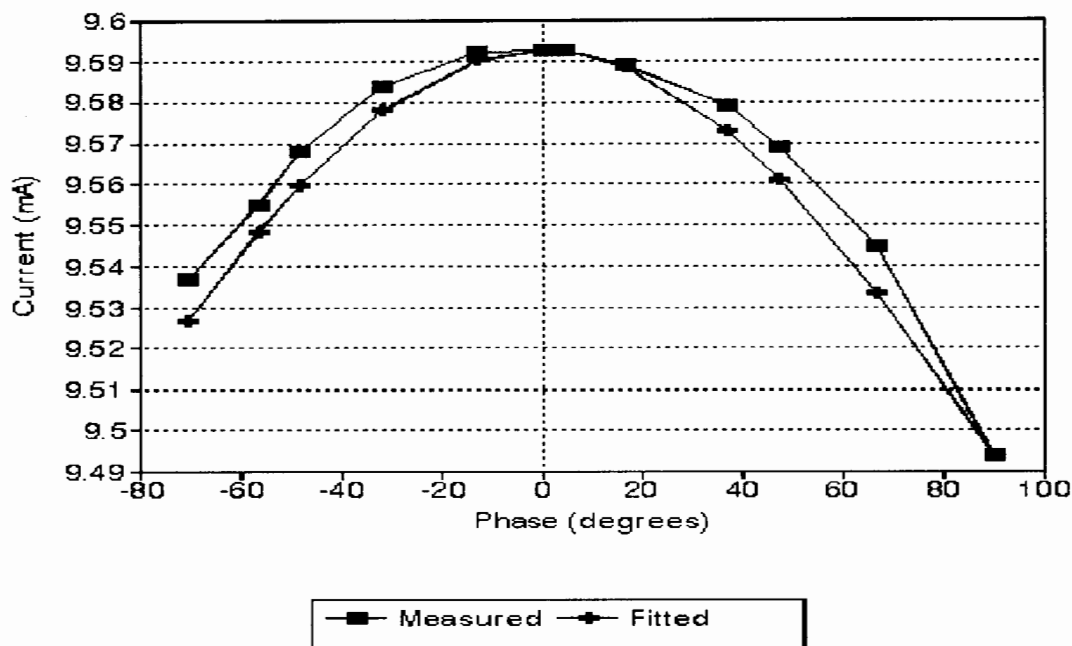
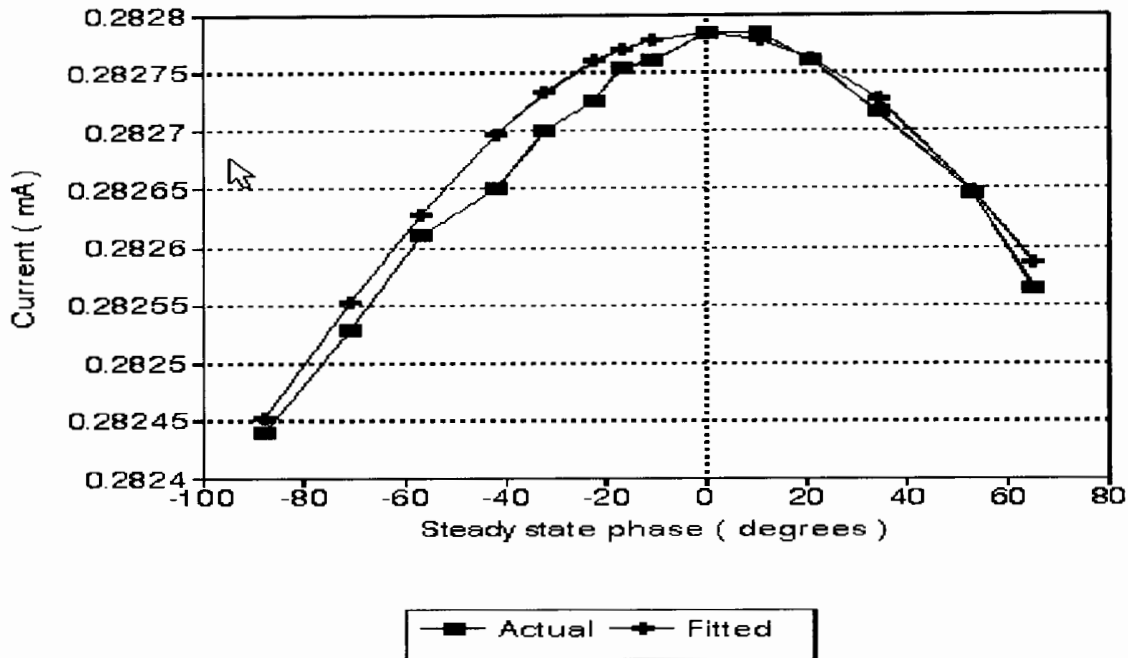


Figure 5.33 Graph of the actual and fitted 'current hump' curves for the optimised colpitts oscillator. The close fit suggests that the equation could be correct.

1. LOCKING TIMES. As the output frequency of an injection locked oscillator is a function of only the instantaneous input frequency its frequency synchronisation times are faster than those of a phase-locked loop which has to integrate phase with respect to time to achieve frequency lock. Furthermore, attempts to reduce the locking times in a phase-locked loop can



**Figure 5.34** Graph of the actual and fitted 'current hump' curves for the synchronous oscillator as designed by Hickman. The close fit suggests that the equation could be correct.

lead to degradation of the noise performance of the circuit. This is due to a phase-locked loop's response being governed by the loop filter. A filter which will allow for quick frequency acquisition will also do little to suppress the noise at the loop input. By comparison however an injection locked oscillator's noise bandwidth is to a large extent independent of its acquisition time.

2. **STORAGE TIME.** The storage time of a synchronous oscillator can be designed independently of the noise and tracking bandwidths. This is due to its being a function of a feedback capacitor. In the phase-locked loop the output frequency will be maintained only by the loop filter. Therefore to have a good storage time in a phase-locked loop one requires a narrowband loop filter. This in turn requires a trade off in locking times.
3. **PHASE COHERENCY.** The injection locked oscillator will introduce a measure of phase shift between its input and output dependent on the frequency offset of the input from the free running frequency of the oscillator. Theoretical phase-locked loops, on the other hand, by the nature of their locking processes lock with either no phase offset or with a phase offset that is a multiple of 90 degrees. Practical circuits may not adhere to these phase differences due to circuit D.C. offsets, but with good design such errors are small.

As phase coherency is important in many synchronisation networks this is an major consideration when choosing a synchronisation network. Although an injection locked oscillator will always introduce a steady-state phase error, if the frequency differences are kept small the phase error generated by the injection locked oscillator will remain small.

Use of the coherent phase-locked synchronous oscillator can remove the phase error between the input and output. The time taken to achieve phase coherency is however similar to that of the phase-locked loop. One does still retain the advantages of a noise bandwidth independent of the tracking range. This will allow one to design for a wider tracking range than for a phase-locked loop at a specific signal-to-noise ratio. However, the cost of a coherent phase-locked synchronous oscillator approaches that of a phase-locked loop due to the incorporation of a phase detector. Furthermore, care must be taken in the design of the phase detector so as to

supply the correct bias voltage to the base of the transistor. This makes the design difficult. Due to the wide spread use of phase-locked loops however one can obtain chip sets which are both effective and inexpensive.

During the prototyping of the novel synchronisation circuits both synchronisation networks were used.

---

# DESCRIPTION OF THE PROTOTYPING OF THE NOVEL MAXIMAL-LENGTH SEQUENCE SYNCHRONISATION NETWORKS.

Both the novel synchronisation loop with the differentiator in the feedback path and the novel loop with the differentiator in the input path were built and tested.

## 6.1 RESULTS FROM THE NOVEL SYNCHRONISATION NETWORK WITH THE DIFFERENTIATOR IN THE FEEDBACK PATH.

The novel synchronisation loop with the differentiator in the feedback path was built. The spikes were generated by differentiation and the adjustable clock was implemented by the synchronous oscillator as designed by Uzunoglu and White ( Figure 5.18 ).

### 6.1.1 Design of the Prototype Circuit for the Synchronisation Loop with the Differentiator in the Feedback Path.

The use of the synchronous oscillator raises some design criterion. A centre frequency for prototyping must be chosen. This was set at 8 Megahertz as the function generator used could only generate the square waves used to drive the transmitter's maximal-length sequence generator up to a frequency of 10 Megahertz. The component values used in the design of the synchronous oscillator were:

$$\begin{aligned}L &= 0.134 \mu\text{H} \\C_1 &= C_2 = 6.8 \text{ nF} \\G &= 53 \text{ k}\Omega \\G_B &= 160 \text{ k}\Omega\end{aligned}$$

As the synchronous oscillator will be required to maintain the correct clocking frequency under conditions of no input, the storage time should be as large as possible. The feedback capacitor  $C_3$  was therefore set to 100nF.

The transistors,  $T_1$  and  $T_2$ , used were the BFR90A. These are high speed surface mount transistors with a bandwidth of 3 Gigahertz and a collector current of 25 milliamp. The large bandwidth is required to ensure that the transistors operate in a linear region of their gain curves. The transistors have a large gain  $h_{fe}$ . As the synchronisation condition for injection locked oscillators is proportional to this

parameter this will extend the locking range of the synchronous oscillator and thereby the locking range of the synchronisation loop.

The multiplier was implemented using a SBL-1. As was the case with the signed edge detector the input to the multiplier was attenuated by a resistor  $\pi$  network matched to  $50 \Omega$ . The output of the synchronous oscillator was hard-limited using the LM360 high speed comparator in order to drive the receiver maximal-length sequence generator.

Each loop component was built on a separate board. They were connected using co-axial cable in order to try and shield the signals and thereby reduce the noise levels. Noise was further reduced by decoupling the supply line on every component by placing a  $100 \text{ nF}$  capacitor to ground.

### 6.1.2 Results from the Prototype Loop with the Differentiator in the Feedback Path.

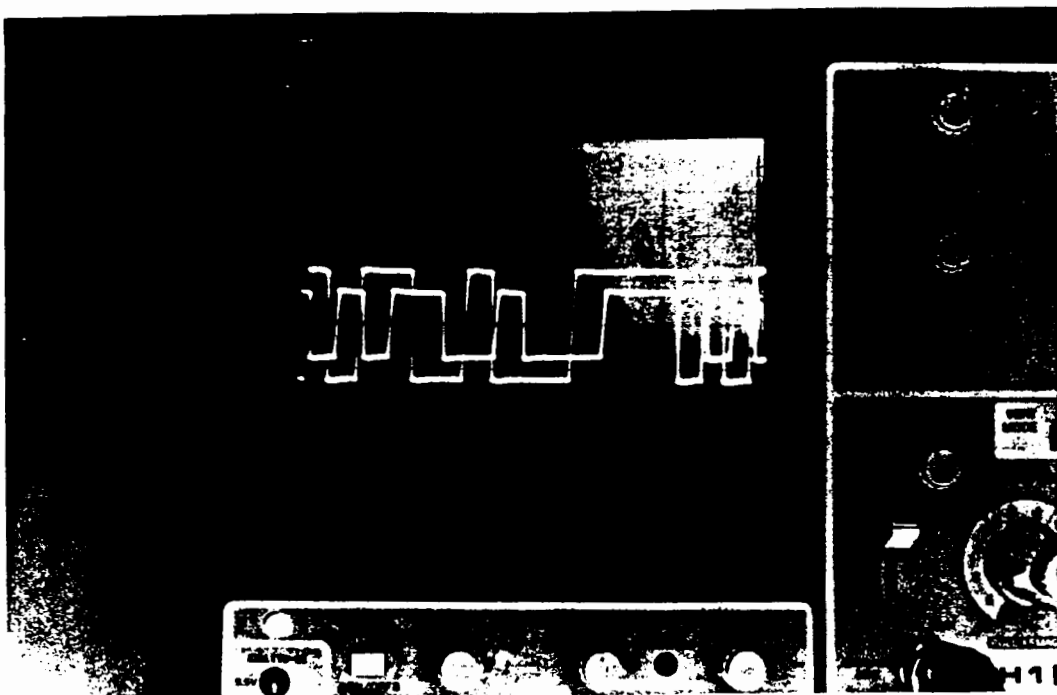


Figure 6.1 Photograph of the two maximal-length sequences in lock at a one bit code-phase offset. The oscilloscope was set to trigger on the input waveform.

Figure 6.1 depicts the two  $2^4 - 1$  bit pseudorandom sequences. The synchronisation loop effected lock at a one bit code-phase offset. Figure 6.2 depicts the corresponding spike. The spike are negative going, although there is some overshoot. The best results were obtained when the clock frequency of the input was higher than the free running frequency of the synchronous oscillator yet still within the locking range of the loop.

The loop was stable, once lock was achieved, and was able to track the input frequency across the range of a few hundred kilohertz. However, the loop failed in two regions. Firstly, at frequencies outside the locking range of the synchronous oscillator. At these frequencies the synchronisation condition is violated and the synchronous oscillator cannot supply the correct driving signal to the maximal-length sequence generator. Secondly, the loop failed to achieve synchronisation at frequencies around the centre

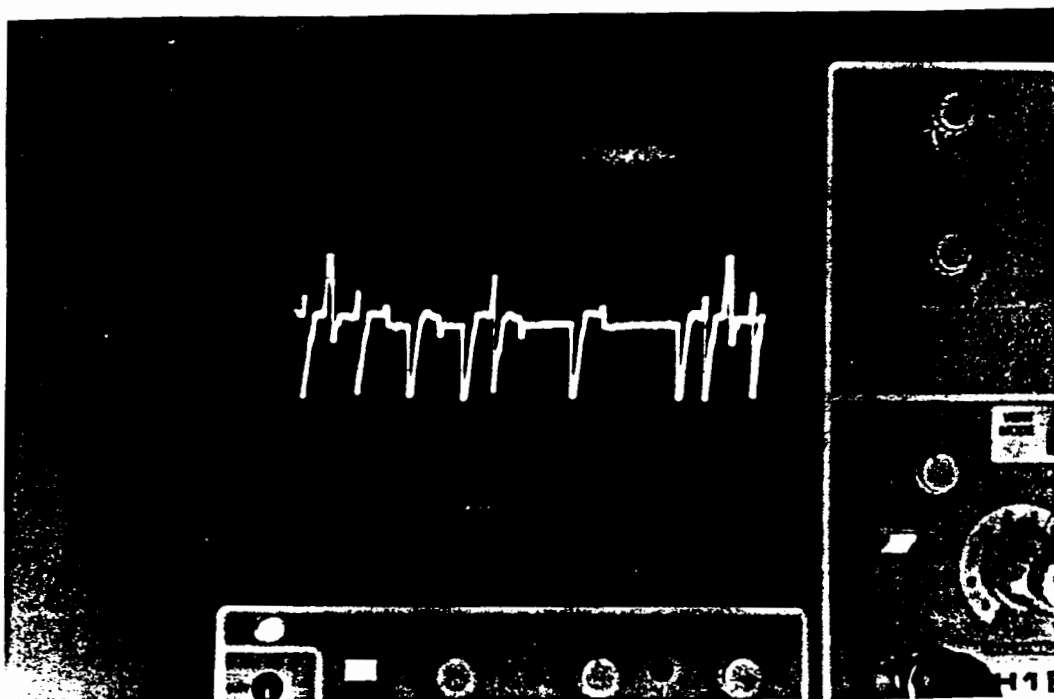


Figure 6.2 Photograph of the spikes at the output of the maximal likelihood phase detector. The spikes are negative going although some overshoot exists.

frequency of the synchronous oscillator. In this region false lock was achieved in some cases. False lock is characterised by the loop resolving the input frequency correctly, but not acquiring the input code-phase.

This is due to the power limits of the SBL-1 mixer. The output power is limited which results in spikes that are small. A single incorrect spike does not therefore cause a large perturbation of frequency at the output of the synchronous oscillator.

However, the results of the simulation of the novel synchronisation loop with the differentiator in the feedback loop were confirmed. The loop did acquire and maintain lock in the noiseless case [3].

## 6.2 PROTOTYPING OF THE NOVEL SYNCHRONISATION NETWORK WITH THE DIFFERENTIATOR IN THE INPUT PATH.

The novel synchronisation loop with the differentiator in the receiver input path, as it was found to be more stable than the loop with the differentiator in the feedback path during the simulation of the novel loops, was built and further adapted to the data environment.

### 6.2.1 Design of the Prototype Circuit for the Synchronisation Loop with the Differentiator in the Input.

The novel synchronisation loop with the differentiator in the input path was built on the bench. The adjustable clock was implemented as a phase-locked loop, preceded by a bandpass filter, and the differentiator as the signed edge detection circuit.

As phase-locked loops are widely used there are a wide variety of phase-locked loop chips to choose from. The one chosen for the prototyping was the XR-215 due to its large frequency bandwidth and dynamic range. According to the specifications given in the data sheet it can track frequencies from 0.5 Hz to 35 MHz and at signal strengths between 300  $\mu$ V and 3 V. Furthermore its output is claimed to be compatible with the DTL, TTL and ECL logic families and so could be used to drive the receiver's pseudo-noise sequences without the need for a comparator. This however was not found to be the case and the output needed to be hard-limited. The LM360 high speed comparator was used for this.

The phase-locked loop chip is easily configurable with the free running frequency set by one external capacitor  $C_o$ . In this case the free running frequency was set at 6 Megahertz by setting  $C_o = 39$  pF. The loop filter is also configurable, but as the phase discriminator is internally connected to the V.C.O. the signals cannot be extracted for comparison to the simulation. Another draw back is the failure of the data sheet to mention the type of phase discriminator used, beyond the fact that it is balanced. One is therefore unable to do a full comparison with the results of the simulations.

The loop gain  $K$  of the XR-215 is adjustable. As the loop should have a high gain in order that changes in the polarity of the spikes at the input to the adjustable clock have a large effect this was the first design parameter that was set. The gain  $K$  is the product of the gain of the phase discriminator,  $K_d$ , which is given in the application notes for the device as 2 Volt per radian and the gain of the voltage controlled oscillator,  $K_o$ , which is given in the notes as

$$K_o = \frac{700}{C_o R_o}$$

where  $C_o$  is the timing capacitor which sets the natural frequency of the phase-locked loop in microfarad and  $R_o$  is a resistor which modifies the gain in k $\Omega$ . As the timing capacitor  $C_o$  is used to set the free running frequency only  $R_o$  is adjustable. This resistor was chosen as 5 k $\Omega$  which is in the middle of the recommended range of 1 to 10 k $\Omega$ . Only the damping factor  $\eta$  must still be chosen in order to calculate the loop filter component values. It was decided to critically damp the loop so as to approach the required output frequency as quickly as possible but not to allow for overshoot. The damping factor was therefore set to 0.5.

The phase-locked loop was designed with passive lead-lag loop filter of the form

$$H(s) = \frac{s\tau_1 + 1}{s(\tau_1 + \tau_2) + 1}$$

where  $\tau_1$  and  $\tau_2$  are the loop time constants. Letting  $\tau_T = \tau_1 + \tau_2$  one can write

$$\tau_T = \frac{K}{\omega_n}$$

where  $\omega_n$  is the natural frequency of  $12\pi \times 10^6$  radian per second. The time constant  $\tau_2$  can be calculated using the equation

$$\tau_2 = \frac{2\eta}{\omega_n} - \frac{1}{K}$$

This then allows one to calculate  $\tau_1$  as

$$\tau_1 = \tau_T - \tau_2$$

The application notes give the loop filter transfer function as

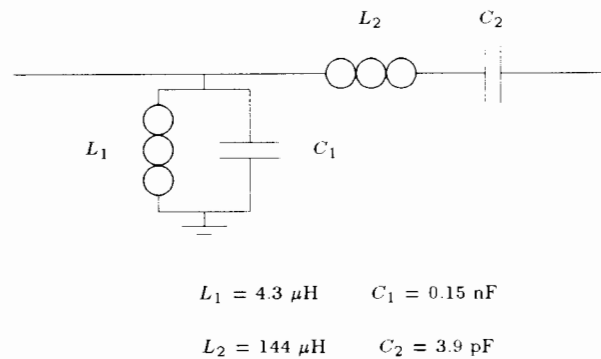
$$H(s) = \frac{sC_1R_2 + 1}{sC_1(R_1 + R_2) + 1}$$

where  $C_1$ ,  $R_1$  and  $R_2$  are the loop filter components. Setting  $C_1$  to 1 nF allows one to calculate the two resistors needed.

$$R_2 = \frac{\tau_2}{C_1} = 172\Omega$$

$$R_1 = \frac{\tau_T}{C_1} = 27\Omega$$

The band pass filter used at the input to the phase-locked loop was a passive Butterworth second order filter ( Figure 6.3 ). A second order loop was chosen as the simulation of the synchronisation loop showed that such a filter was sufficient in a noiseless environment. The component values were calculated using the shareware filter design program FILTRY. The values calculated were verified through a SPICE simulation of the circuit. The inductors used were variable so as to allow for a fine tuning of the center frequency of the filter. This is necessary in order to take into account tolerances in component values. The bandpass filter was adjusted so that its centre frequency was equal to that of the phase-locked loop's free running frequency.



**Figure 6.3** Circuit diagram of the passive Butterworth filter.

Care was taken to reduce the effects of noise in the circuit as much as possible. Due to the digital circuitry operating at high speeds this mostly involved the suppression of transients on the supply lines. Each loop component was built on a separate double sided printed circuit board with large ground plains. All chips and supply lines were decoupled by a 100 nF capacitor to ground. The supply lines were further all regulated by the LM79 and LM78 series of three pinned voltage regulators. The various boards were linked by twisted pairs with as short run lengths as possible. Appendix 2 shows the complete circuit diagram.

## 6.2.2 Results from the Baseband Nondata Modulated Novel Synchronisation Loop.

Figures 6.4 through 6.6 depict the two maximal-length sequences. The loop was able to lock the two sequences even at reasonably long sequence lengths of 132 bits. Figure 6.7 depicts the transmitted maximal-length sequence and the voltage spike at the output of the maximum likelihood phase detector. All the spike are negative going indicating that lock has been achieved. The time offset between the bit

edges and the spike is due to the delays introduced by the schmitt triggers on the input. The triggers were included to reduce the ripple. The loop was stable and was left running for 24 hours without losing lock. The loop was furthermore able to resolve a varying input clocking frequency.

However, the loop also did obtain false lock under some combinations of sequence length and frequency offset. This is most probably due to the prototype implementation. Due to the output and input power limitations of the SBL-1 mixer the spike had to be attenuated in order to drive mixer and were small at the output of the mixer. This coupled with some ringing on the lines resulted in voltage spikes that were not as well defined as they were in the simulation. A spike of incorrect polarity might therefore not have caused as large a perturbation in the phase-locked loop as was needed to drive the loop from this unstable point in the baseband Spread Spectrum search grid as expressed in terms of integrated phase. Further circuits should use multipliers capable of generating larger spike.

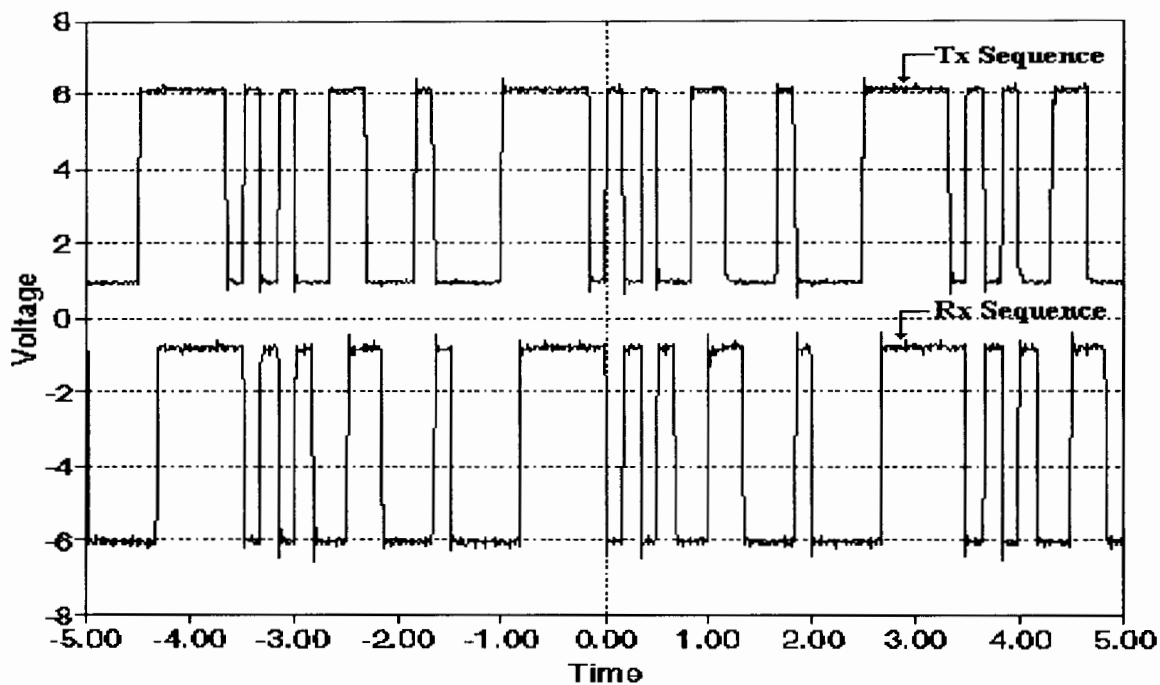


Figure 6.4 Plot of the two sixteen bit maximal-length sequences in lock.

It was attempted to incorporate the synchronous oscillator as designed by Hickman as the loop's adjustable clock. However, the voltage spike at the input although small were amplified by the input stage of the synchronous oscillator and were fed through to the output. The high speed comparator then caused glitches in the clocking waveform of the receiver's pseudorandom sequence generator causing its output to become zero. Possible solutions could include the placing of a narrower bandpass filter at the input to the synchronous oscillator in order to suppress the spike.

The prototyping did prove however that the loop is able to synchronise two maximal-length sequences accurately and maintain that lock over a large number of sequence lengths.

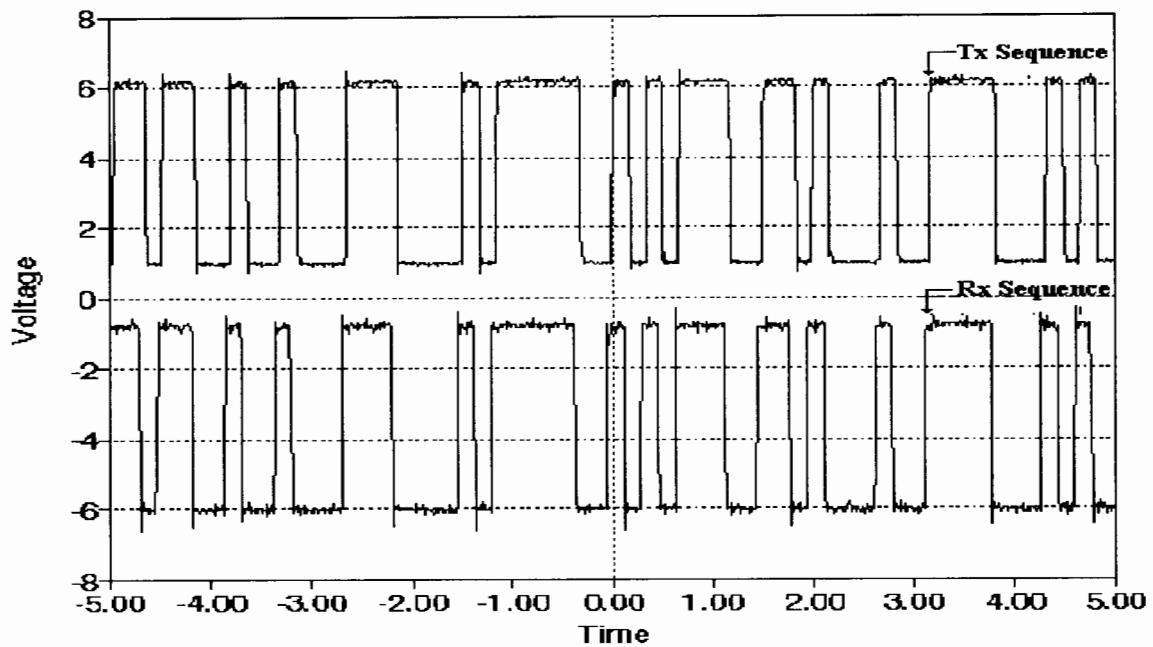


Figure 6.5 Plot of the two thirty-two bit maximal-length sequences in lock.

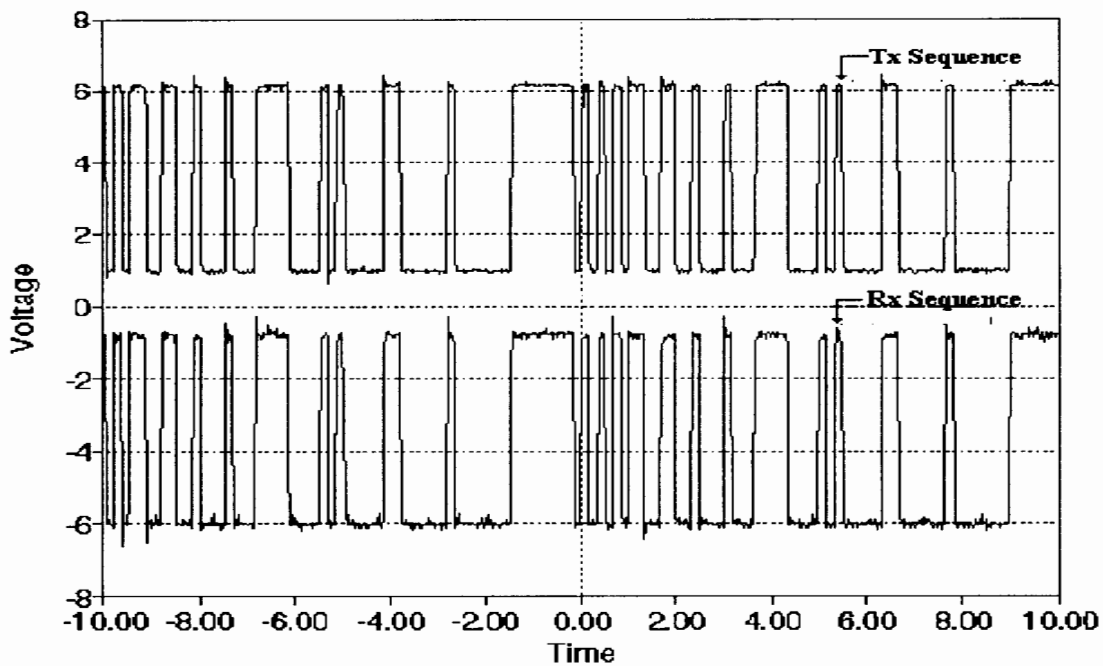
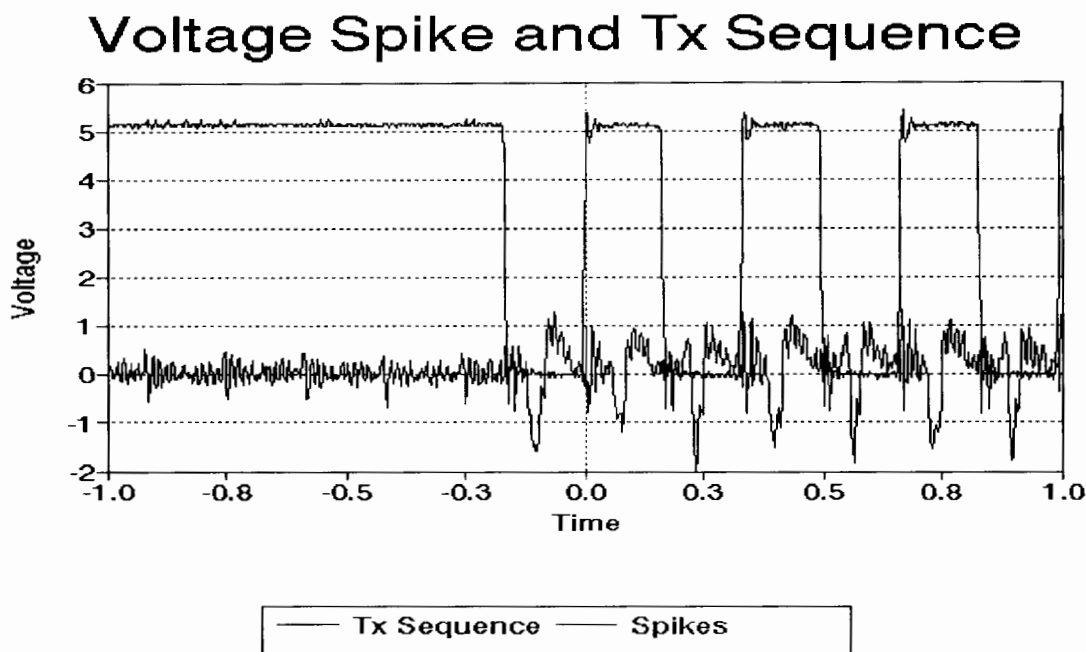


Figure 6.6 Plot of the two one hundred and thirty-two bit maximal-length sequences in lock.

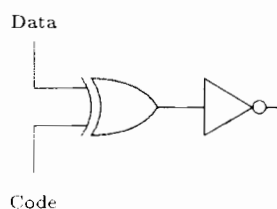
### 6.2.3 Results of the Novel Synchronisation Loop Adapted to the Data Environment.

The novel synchronisation loop with the differentiator in the input path was adapted to the data environment. However, unlike the simulated loops, the data modulation and detection circuitry invert signal were generated digitally.



**Figure 6.7** Plot of the transmitted maximal-length sequence and the voltage spike at the output of the maximal likelihood phase discriminator. The negative going spike indicate that the loop has achieved synchronisation.

Figure 6.8 depicts the digital data modulator. Rather than using a multiplier the transmitted maximal-length sequence is inverted digitally when a data ‘zero’ is sent. Table 6.1 shows the logic table for the digital data modulator. From the table it is clear that the transmitted pseudorandom sequence is inverted when the data bit is a ‘zero’.



**Figure 6.8** Circuit diagram of the digital data modulator. The output is inverted on every ‘zero’ data bit.

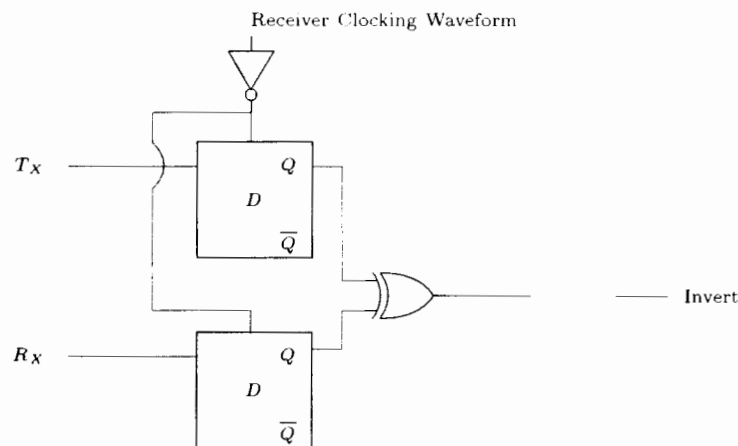
The detection circuitry proposed for the prototyping of the loop in the data environment will have the same characteristics as the detection circuitry based on the output of the maximal likelihood phase discriminator. This is because the decisions made by the circuit will not be done on integrated data, which averages out the effects of noise, but rather the instantaneous loop conditions which may contain errors due to noise. However, the circuit was decided on for three reasons. Firstly, as the detection circuitry is wholly digital it is easy to implement. Secondly, as the prototyping was carried out in a noiseless environment there are unlikely to be errors that preclude the use of circuitry that could be prone to these errors. Thirdly, the circuit was built only to confirm that switching is a practical solution to the data modulation problem. It was therefore further decided that the circuit need work only for

Data Bit	Code Bit	$Data \oplus Code$	$Data \oplus Code$	Action
1	1	0	1	Not Inverted
1	0	1	0	Not Inverted
0	1	1	0	Inverted
0	0	0	1	Inverted

**Table 6.1** Logic table for the digital data modulator.

sequences precisely aligned and not for sequences offset in code-phase by one bit. The detection of lock and the corresponding decision of switching in the invert signal would be done manually.

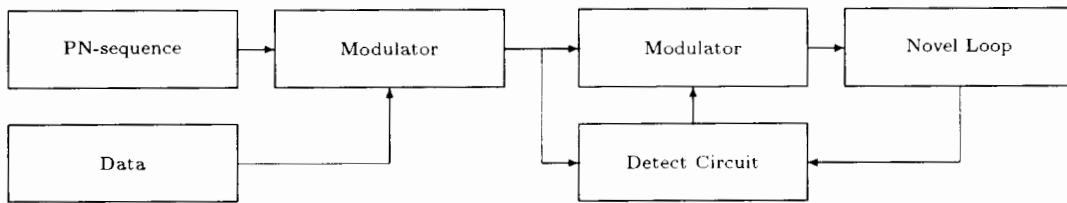
Figure 6.9 depicts the circuit diagram of the detection circuitry used to generate the invert signal. The receiver clock is inverted so as to clock the D-type flip-flops half way through a maximal-length sequence bit. This will allow the pseudorandom sequence output time to have settled. The flip-flops store the received maximal-length bit and the receiver's maximal-length sequence bit. An EXOR-gate detects whether the bits are the same or the inverse of each other. If they are the inverse an invert signal will be generated. The switch on the output allows the invert signal to be manually included once lock has been determined on an oscilloscope.



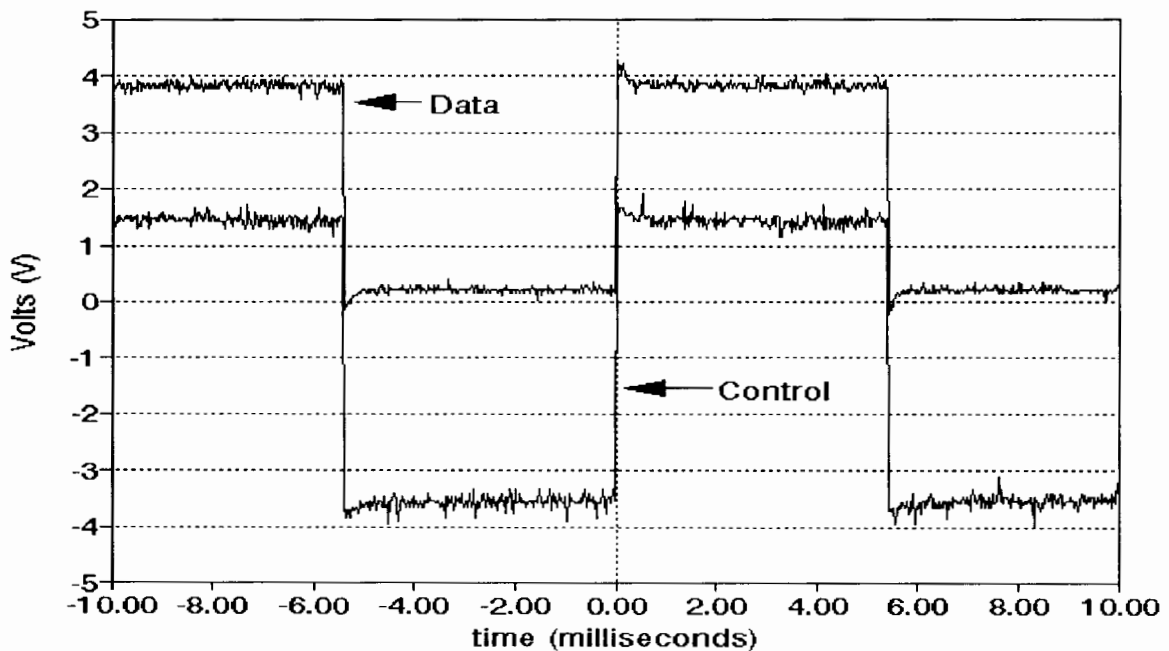
**Figure 6.9** Circuit diagram of the digital detection circuitry. The invert signal will be switched on manually once lock has been detected.

The detection circuitry is included in the circuit as shown in figure 6.10. The invert signal forms one of the inputs of a digital data modulator, the other input being the received sequence. This allowed the received sequence to be inverted before differentiation. It was therefore not necessary to alter the prototype novel synchronisation loop by including another multiplier. Furthermore the inclusion of yet another multiplier could have embedded the spikes in increased noise. This in turn could have increased the occurrence of false lock.

The data was a square wave at 100 kilohertz while the input maximal-length clocking frequency was 6 Megahertz. Figure 6.11 depicts the data and invert signals. The invert signal accurately inverts the data signal. Use of the invert signal to rectify the received maximal-length sequence was successful. The received sequence and receiver replica of the spreading sequence remained in lock ( Figure 6.12 ).



**Figure 6.10** Block diagram of the novel synchronisation loop adapted to the data environment during prototyping. The invert signal is used to invert the received data rather than the spikes at the output of the maximal likelihood phase discriminator. No further alterations to the loop were therefore needed.



**Figure 6.11** Plot of the data and invert signals. The detection circuitry accurately estimates the incoming data.

The results prove that data estimation is a practical solution to the problems data modulation raises. However, the implementation as shown with manual detection of lock and switching in of the invert waveform is not stable. The switching process creates transients in the invert signal. This noise causes glitches in the detection circuitry's invert signal which in turn causes glitches in the received sequence. These errors cause further errors in the invert signal resulting in the received sequence not being a maximal length sequence. This destabilises the system and lock is not achieved in every case. Electronic detection of lock and switching should lessen these problems as was done in the simulations of the loop.

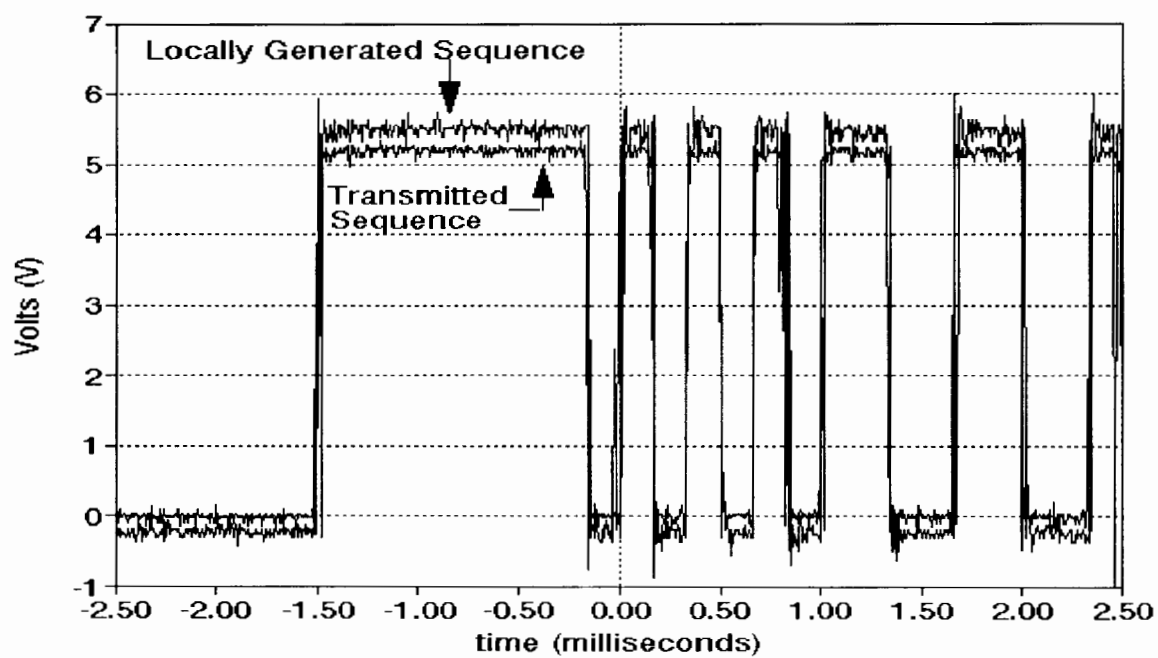


Figure 6.12 Plot of the two maximal-length sequences in lock in the data environment.

---

## CONCLUSION.

Spread Spectrum is a modulation scheme where the conservation of bandwidth principle is sacrificed in order to gain the benefits that a wideband signal can offer. These benefits include multi-path rejection, low probability of detection and multiple access. The most common method of implementation, and the method of interest in this thesis, consists of multiplying the data waveform by a maximal-length sequence which is a wideband, digitally generated signal. This is referred to as Direct Sequence Spread Spectrum.

As a Spread Spectrum system requires that the data be demodulated, at least in part, through the correlation of the received waveform and the receiver's replica of the spreading waveform, synchronisation is very important. Due to the properties of maximal-length sequences a code-phase offset of more than a bit will result in a low correlation value and therefore not allow for detection of the data waveform.

Most common synchronisation loops use a two pronged approach to achieve lock. Firstly the loop attempts to resolve the correct code-phase of the received waveform. This is referred to as code acquisition. Secondly the loop attempts to resolve the clocking frequency of the transmitter. This is referred to as code tracking. This two pronged approach, while effective, leads to circuits which are complex and therefore expensive.

This thesis described the investigation of two maximal-length sequence synchronisation loops proposed by J.G. van de Groenendaal. Both were developed to provide a simple yet robust method of achieving synchronisation by performing both code acquisition and code tracking in a single implementation. Both loops were based on variations of the maximal likelihood phase discriminator driving an adjustable clock.

The initial investigation consisted of expressing the baseband Spread Spectrum search grid, a grid where each possible code-phase and frequency offset represents a cell, in terms of the integral with respect to time of the phase information generated by the maximal likelihood phase detector. It was found that two stable cells existed. These are at no frequency offset, which means that the loop should track the received waveform, and either a one bit or no code-phase offset, which means that the loop should acquire the code-phase of the received waveform.

A simple model whereby the loop conditions are represented by a 'ball' resting on the surface of the search grid was described. The 'ball' will roll from unstable cell to unstable cell until one of the two stable cells is entered into. The 'ball' will not exit this cell and lock is achieved. This model gives insight into the chaotic manner in which a small variation in starting code-phase and frequency leads to widely different paths through the search grid till lock is achieved.

The first loop investigated consists of an adjustable clock driven by the product of the transmitted sequence and the first derivative with respect to time of the receiver's replica of the transmitted waveform. This loop was able to perform code acquisition but was unstable and parameter sensitive.

The loop was unstable due to the position of the differentiator in the synchronisation loop. As the differentiator took as input the receiver's pseudorandom sequence the clock frequency recovered was that of the loop's adjustable clock. Any jitter in the clock frequency was thus fed back resulting in instability.

The second loop moved the differentiator into the receiver input path. The frequency information supplied to the adjustable clock, being a function of the transmitter clock, now allows for the correct tracking of the received waveform. This loop therefore offered a complete synchronisation solution in one simple loop.

The novel synchronisation loop with the differentiator in the input path was investigated in detail. However, due to the difference in the synchronisation approach, comparisons to existing loops were difficult. The loop's locking times, in a noiseless environment, were found to be three times longer than those of the common code acquisition loops. Under noise, where the loop performed synchronisation at signal-to-noise ratios of -2 dB, the locking times of the novel loop were increased. However, the loop is restricted to coherent Direct Sequence Spread Spectrum systems. These systems however tend not to be used as they interfere with adjacent channels.

The loop was adapted to the data environment. A data stream would invert the phase discriminator's output and drive the loop out of lock. Two methods were investigated. The first disregarded all data information and performed edge detection after lock. This process however resulted in the errors due to noise having a cumulative effect and therefore suffered from a large degradation in noise performance from the loop without the adaptation. The second adaptation of the loop involved data estimation. This method estimated the data stream and then inverted the received waveform when needed to maintain the phase discriminator's slope. This method, through the generation of self noise, suffered only a small noise performance loss.

The loops were built and were able to achieve lock. However, they would sometimes enter a condition of false lock when only the frequency and not the code-phase of the received waveform would be acquired. This was felt to be due to the maximum power levels in the multipliers preventing the generation of spike that could cause a large perturbation at the input to the adjustable clock.

Investigation of alternative practical implementations of certain novel synchronisation loop components led to interesting findings. Firstly, in order to implement a wideband square wave differentiator yet not increase the noise levels in the circuit the signed edge detector was developed. This circuit, based on the edge detector, provides signed pulses for every bit edge yet is a low noise implementation as the input is only sampled when a marked change in input occurs. Secondly, the synchronous oscillator, a form of injection locked oscillator, was investigated as an alternative to the phase-locked loop. The synchronous oscillator exhibits many interesting features including a 'flat' magnitude frequency response across the tracking range, a steady state phase error between input and output, independent noise and tracking bandwidths, the ability to set storage times independently of bandwidths and the 'current hump'.

When the injection frequency is swept across an injection locked oscillator's tracking range the current drawn from the power supply forms a hump with its peak at the oscillator's center frequency. An equation linking this hump to the steady state phase error fitted the experimental data well. As the synchronous oscillator also displayed this 'current hump' it is felt that it too is possibly an injection locked oscillator.

The second of the two loops with an adjustable clock driven by the product of the first derivative with respect to time of the received waveform and the receiver's replica of the spreading waveform was found to offer a complete synchronisation of maximal-length sequences of short run lengths. The loop is robust and simple. However, the use of the loop is restricted by the signal-to-noise ratio at which lock can be achieved. This level is higher than those needed by non-coherent synchronisation schemes.

---

## RECOMMENDATIONS.

The novel synchronisation loops as proposed by J.G. van de Groenendaal have been shown to work through simulation and proto-typing. However, further tests could still be performed. These include:

1. EFFECTS OF NON GAUSSIAN NOISE. The loop works well in white additive Gaussian noise down to levels of -2 dB. Other noise sources such as multi-path interference should be investigated.
2. EFFECTS OF NOISE ON THE PRACTICAL LOOP. This would involve improvement of the proto-type loop through the use of a multiplier capable of producing larger voltage spike.

---

# TESLA SOURCE CODE.

This is the source code for the novel synchronisation loop with the differentiator in the input path adapted to the data environment through use of the despread output and the receiver's clocking waveform. As this was the most complex as well as the most successful loop it is the only one that will be presented.

\* Simulation of the tracking loop proposed by Johan van den  
\* Groenendaal. By A.M.Davidson on 25-11-94

\* Set reference levels.

```
pos      50      PWR V=1
neg      51      PWR V=-1
logicone 52      PWR V=5
dcllevel 53      PWR V=-2.5
level    54      PWR V=0.5
```

\* CIRCUIT DESCRIPTION

\* This is the input pn sequence

```
fclock    0 0 10 30    FCNGEN FCN=2 F=10.10E4 V=2.5 DC=2.5
pnsequence 10 11      PNSEQ      ;clock
```

\* The data is generated using the TESLA data generator. However, a  
\* header of all 'ones' must be sent. A pulse of the required width is  
\* generated. This forms the control signal to a multiplexor which will  
\* switch in the data once the header has been transmitted.

```
dclclock1 10 52 201 202 203 204  CNTR4 ;divide clock to
dclclock2 204 52 20 205 206 207  CNTR4 ;run at integral div.
data      20 22 25      BERTX EXTCLK ;TESLA data gen.
capacitor 22 54 26      SUM G1=.2 G2=-1
delay     21           PULSE PER=1 PW=3E-2 V=5 INV=0
swap      21 26 54 12  MUX      ;give time for
           ;synchronisation
adddata   12 11 7002    MULT
```

\* Amplify by two to get correct signal-to-noise as other simulation.

```
Amp        7002 0 1002    AMP G=2
```

\* Add some noise !!!!!

```
noise      1001          NOISE V=18
addin      1001 1002 1003 SUM G1=1 G2=1
```

\* Bit estimation is done by lowpass filtering and comparison.

```
lowpass    1003 1004      FILTER BUTR LP F=10.10e4 N=4
square     0 1004 1      COMP
```

\* The input is then differentiated.

```
differ     1 2          DIFF W=60E6
```

\* The maximal likelihood phase discriminator is followed by the the  
\* rectification multiplier. The detect circuitry determines whether the

\* multiplexor will multiply the spikes by a plus or minus one.

```
phase      2 5 60      MULT    ;Compare phase
datarec    100 50 51 23 MUX     ;mux to choose
                    ;sign
swapsign    23 60 3     MULT
lockdet     72 4 100    DETECT
```

\* Genertate a pulse to reset the integrate and dump circuit. The data  
\* clock is fed through from the transmitter.

```
reset       20 52 83    OSHOT T=1E-5
                    ;reset pulse
invert      83 82      INV     ;reset on low
correl      5 1 70     MULT
samplehold  82 70 72    INTEGR T=.1E-3
```

\* This is the adjustable clock. It will be shifted until phase and  
\* frequency is locked.

```
synch       3 4        SOS     ;receiver clock
```

\* This is the reciever pn sequence.

```
receiver     4 5        RECPN   ;receiver pn sequence
```

\*-----

\* LIBRARY OF SUBCIRCUITS USED.

\*-----

\* The input pn-sequence. Set to starting phase zero, is 16 bit maximal  
\* length.

\*  
\* CLOCK on 10  
\* OUTPUT on 11

```
.SUBCKT PNSEQ 10 11
```

\* Set reference levels

```
downshift   53          PWR V=-2.5
                    ;remove DC from logic
pngenin     10 1 14 15 16 17 PNGN4+0 ;see PNLIB.SUB
gatet       14 17 1     XOR     ;gate 1 and 4
capacitor    53 1 11    SUM G1=1 G2=1
.ENDS
```

\*-----

\*-----

\* This subcircuit interprets the data output. As the correlation value  
\* below -10V then the system is locked. The correlation value is also  
\* detected over one clock cycle to detect a sign change due to a data  
\* change.

\* INPUT on 1 ( correlator ) 2 ( receiver pn sequence clock )  
\* OUTPUTS on 62 (swap)

```
.SUBCKT DETECT 1 2 62
```

\* Set reference levels

```
poslevel    50          PWR V=0.8 ;positive above
neglevel    51          PWR V=-0.8 ;negative below
digitlevel  52          PWR V=5 ;logic level
posloclev   53          PWR V=10 ;positive lock
negloclev   54          PWR V=-10 ;negative lock
```

\* Calculate if the circuit is in lock

```
negdetect   54 1 8      COMP    ;negative spike
negpulse    8 52 199    OSHOT T=.5E-5
                    ;generate pulse
locked2     199 39 9    AND
```

\* Detect wheather the circuit is a negative or positive locking circuit

```
negativelking 9 32 39    KEEP   ;set register
```

\* Decide wheather to generate the invert signal.

```

genclk      2 52 10      OSHOT T=.5E-5
                ;pulse from clk
pulsechange 10 11      INV
corinpt     11 1 112    INTEGR T=.1E-3
                ;integrate 1 bit
amp2        112 0 12    AMP G=10

neggoing    12 51 13    COMP

swap?       13 32 3     AND

holdpos     3 2 52 52 62 21 DFF IC=0
                ;hold and output

```

ENDS

\*

\*

\* This implements a register that stores and locks itself

\*

\* INPUT on 1

\* OUTPUT on 2 and 3

.SUBCKT KEEP 1 2 3

\* Set levels

```
digitlevel  5          PWR V=5 ;logic level
```

```
highin      1 4 5 5 2 3 DFF IC=0
```

```
lockgate    3 1 4      NAND
```

ENDS

\*

\*

\* TESLA only supports NAND and XOR gates. This is an AND gate. Does this  
\* by inverting a NAND gate.

\*

\* INPUTS on 1 and 2

\* OUTPUT on 3

.SUBCKT AND 1 2 3

```
tesnand     1 2 4      NAND
```

```
andcon      4 3        INV
```

ENDS

\*

\*

\* TESLA only supports NAND and XOR gates. This is an OR gate. Does this  
\* by inverting a XOR gate.

\*

\* INPUTS on 1 and 2

\* OUTPUT on 3

.SUBCKT OR 1 2 3

```
tesxor      1 2 4      NOR
```

```
orcon       4 3        INV
```

ENDS

\*

\*

\* This is the adjustable clock. The spikes are fed into a bandpass  
\* filter. The output of this filter is used to drive a phase locked  
\* loop. The output of the VCO is then squared up in a comparator to  
\* give a digital clock output

\*

\* INPUT on 3

\* OUTPUT on 8

.SUBCKT SOS 3 8

```
fsos        3 4          FILTER BUTR BP N=2 F=1.00E5 B=7E3
```

\* Phase lock loop has a lead-lag loop filter

```
mpll        4 7 5      MULT
```

```
fppll       5 6          INTEG T=3E-6 FZ=1E3
```

```
vcopll      6 7          VCO V=5 F0=1E5 K0=3E3 FMIN=9.7E4 FMAX=10.3E4
```

```

comps      0 7 8      COMP
.ENDS
*-----
*
*-----
* This is the receiver pn-sequence. One can adjust its starting phase
* in relation to the input sequence. The output is shifted down so that
* it can be multiplied by the input to give positive and negative
* spikes.
*
* CLOCK on 8
* OUTPUT on 21

SUBCKT RECPN 8 21
* Set reference levels

downshift   53      PWR V=-2.5      ;remove DC from logic

pngenout    8 9 31 32 33 34 PNGN4+7      ;see PNLIB.SUB
gate        31 34 9      XOR
capacitor   53 9 21      SUM G1=1 G2=1
*rndcorners 21 22      RCLP F=1E5
.ENDS
*-----

INCLUDE C:"TESLA"PNLIB.SUB

```

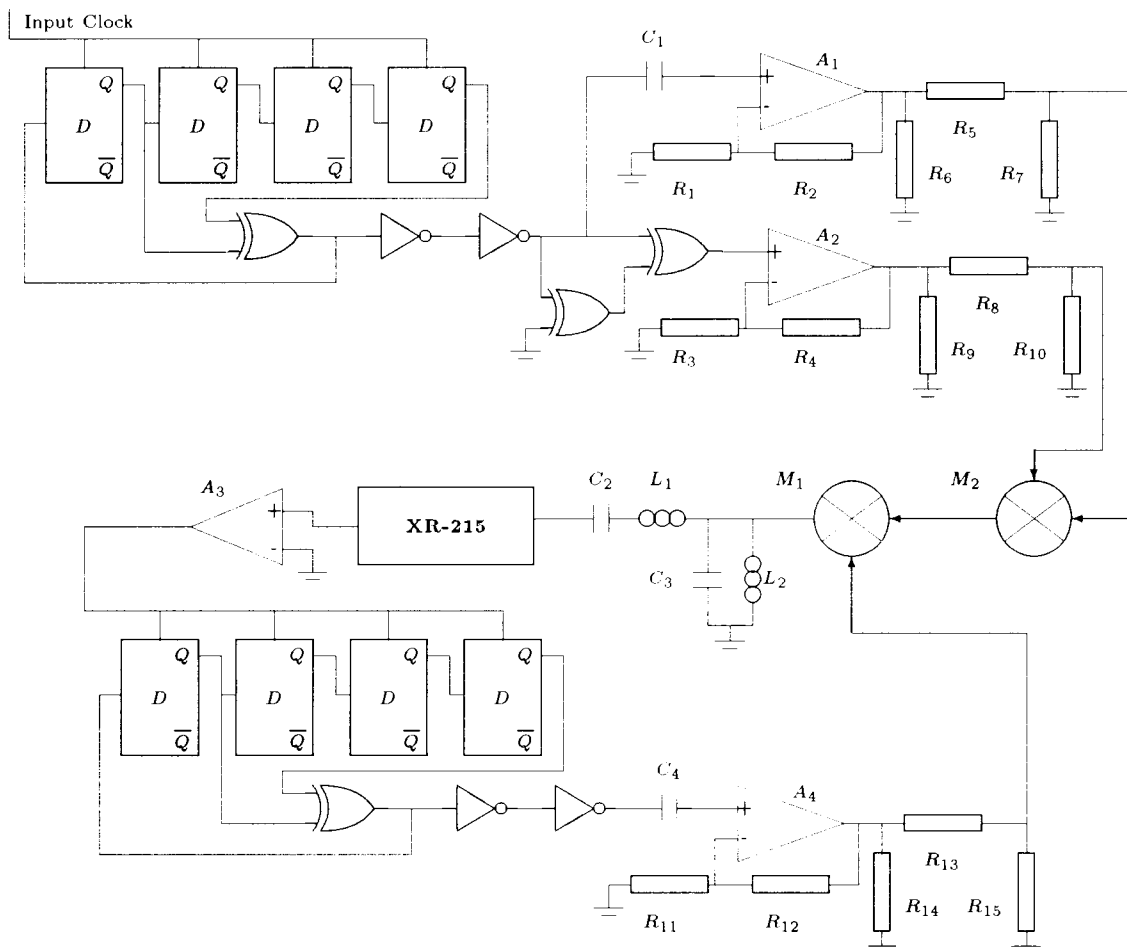
The following TESLA source code is an extract from the sub-library of maximal-length sequences of varied code-phase offsets. As these were commonly used through out all the simulations it was impractical to make them sub-circuits in each simulation.

```

*****
* Maximal length 16 bit pseudo-noise sequence with offset code phases.
*****
*
* Input 1 ( External clock ) 3 ( Input of the first shift register. )
* Outputs 4 7 9 11 ( Q output of the flip flops. Must be tapped and
*                    fed back.)
*
SUBCKT PNGN4+0 1 3 4 7 9 11
P1 5 PWR V=5
IN1 1 2 INV ;TO INVERT CLOCK
D1 3 2 5 5 4 6 DFF IC=5
D2 4 2 5 5 7 8 DFF IC=0
D3 7 2 5 5 9 10 DFF IC=5
D4 9 2 5 5 11 12 DFF IC=5
.ENDS
*
SUBCKT PNGN4+1 1 3 4 7 9 11
P1 5 PWR V=5
IN1 1 2 INV ;TO INVERT CLOCK
D1 3 2 5 5 4 6 DFF IC=0
D2 4 2 5 5 7 8 DFF IC=5
D3 7 2 5 5 9 10 DFF IC=0
D4 9 2 5 5 11 12 DFF IC=5
.ENDS
*
SUBCKT PNGN4+2 1 3 4 7 9 11
P1 5 PWR V=5
IN1 1 2 INV ;TO INVERT CLOCK
D1 3 2 5 5 4 6 DFF IC=5
D2 4 2 5 5 7 8 DFF IC=0
D3 7 2 5 5 9 10 DFF IC=5
D4 9 2 5 5 11 12 DFF IC=0
.ENDS

```

## CIRCUIT DIAGRAM OF PROTOTYPE LOOP.



**Figure 22.1** Detailed circuit diagram of prototype loop.

The maximal-length sequence generators consisted of a series of shift registers, the taps of which are EXOR-ed together. The output of the EXOR-gate forms the input to the registers. The shift registers were implemented using HC195 chips while the EXOR gate used was the HC86. The supply lines were decoupled by 100 nF capacitors to ground.

The signed edge detector consisted of the amplifiers  $A_1$  and  $A_2$ , the multiplier  $M_2$ , the resistors  $R_1$  through  $R_9$ , the capacitor  $C_1$ , two schmitt triggers and two EXOR-gates. The schmitt triggers formed the input stage to the signed edge detector. They reduced the ripple on the maximal-length sequence waveform. The edge detector was formed by the two EXOR-gates. In order to produce narrow spikes a one device delay was used. The capacitor  $C_1$  was set at 10 nF. This capacitor removes the D.C. value from the digital input waveform so as not to multiply by zero. The amplifiers  $A_1$  and  $A_2$  were used as line drivers to drive the resistor attenuation networks which are matched to 50  $\Omega$ . The opamp used was the AD8001. The gain of the opamps was set to two by making  $R_1 = R_2 = R_3 = R_4 = 1$  k $\Omega$ . The resistor  $\pi$  attenuation networks were included to meet the input power limits of the mixer  $M_2$ . A 23 dB attenuation was achieved by setting  $R_6 = R_7 = R_9 = R_{10} = 58\Omega$  and  $R_5 = R_8 = 352\Omega$ . The mixer  $M_2$  was implemented using a SBL-1 mixer.

The multiplier board consisted of the multiplier  $M_1$ , the amplifier  $A_4$ , the resistors  $R_{11}$  through  $R_{15}$ , the capacitor  $C_4$  and two schmitt triggers. The schmitt triggers formed the input stage to the signed edge detector. They reduced the ripple on the maximal-length sequence waveform. The capacitor  $C_4$  was set equal to 10 nF and is used to remove the D.C. value from the input waveform so as not to multiply by zero. The amplifier  $A_4$  is used as a line driver. It was implemented by an AD8001. Its gain was set to two by making  $R_{11} = R_{12} = 1$  k $\Omega$ . The resistor  $\pi$  attenuation network attenuates by 23 dB.  $R_{14} = R_{15} = 58\Omega$  and  $R_{13} = 352\Omega$ . The mixer  $M_1$  was implemented using a SBL-1 mixer.

The adjustable clock board consisted of a passive Butterworth filter, a phase-locked loop and a comparator. The Butterworth filter is a second order bandpass filter. The centre frequency was set at 6 Megahertz.  $L_1 = 4.3\mu\text{H}$ ,  $L_2 = 144\mu\text{H}$ ,  $C_2 = 3.9$  pF and  $C_3 = 0.15$  nF. The inductors were tunable. The phase-locked loop was built using the XR-215 chip. The free-running frequency was set at 6 Megahertz by a 39 pF capacitor. For a detailed description of the component value see Chapter 6. The output of the phase-locked loop was hard-limited using an LM360 high speed comparator.

---

## REFERENCES

- [1] R. Adler. A study of locking phenomena in oscillators. *Proceedings of the IRE*, 34:351 – 357, June 1946.
- [2] R.M. Braun. Injection-lock bandwidth extension by control modulation and current 'hump' slope detection. *Electronics Letters*, 21(13):564–565, June 85.
- [3] A.M. Davidson. *Synchronous Oscillator*. University of Cape Town Undergraduate Thesis, November 1993. Supervisor: J.G. van de Groenendaal.
- [4] A.M. Davidson, J.G. van de Groenendaal, and R.M. Braun. The injection locked oscillator as a synchronisation network. *Submitted to IEE Journal on Electronics and Communications*, December 1994.
- [5] A.M. Davidson, J.G. van de Groenendaal, and R.M. Braun. A novel code phase synchronisation network for direct sequence spread spectrum. *IEEE SA Symposium on Communications and Signal Processing 94*, pages 95 – 98, October 1994.
- [6] R.C. Dixon. *Spread Spectrum Systems with Commercial Applications*. John Wiley and Sons, 1994.
- [7] F.M. Gardner. Comments on synchronised oscillators. *Proceedings of the IEEE*, 75(8):1121–1123, August 1987.
- [8] I. Hickman. Synchronous oscillators: Alternative to pll? *Electronics World and Wireless World*, pages 490–494, June 1994.
- [9] J.J. Spilker and D.T. Magill. The delay-lock discriminator - an optimum tracking device. *Proceedings of the IRE*, 49:1403–1416, September 1961.
- [10] F.G. Stremler. *Introduction to Communication Systems (Third Edition)*. Addison-Wesley Publishing Company, 1990.
- [11] V. Uzunoglu and M.H. White. Synchronous and the coherent phase-locked synchronous oscillators : New techniques in synchronisation and tracking. *IEEE Transactions on Circuits and Systems*, 36(7):997–1003, July 1989.
- [12] V. Uzunoglu and M.H. White. The synchronous oscillator: A synchronisation and tracking network. *IEEE Journal on Solid-State Circuits*, 20(6):1214–1226, December 85.
- [13] J.G. van de Groenendaal. Internal report on spread spectrum synchronisation network. *Internal Report*, January 1994.
- [14] J.G. van de Groenendaal. Signed edge detection of square pulses. *Paper for Submission*, April 1994.
- [15] J.G. van de Groenendaal. *Novel Low Cost Synchronisation Network for Spread Spectrum Systems*. PhD thesis, University of Cape Town Graduate School Report: Elec. Eng., 1995.
- [16] J.G. van de Groenendaal and P. van Rooyen. Spread spectrum synchronous oscillator tracking loop used to synchronize pn sequences. *Provisional Patent pending*, January 1993.
- [17] R.E. Ziemer and R.L. Peterson. *Digital Communications and Spread Spectrum Systems*. Macmillan Publishing Company, January 1985.